

# **Mechanics of Solid Polymers**



PLASTICS DESIGN LIBRARY (PDL)

PDL HANDBOOK SERIES

Series Editor: Sina Ebnesajjad, PhD ([sina@FluoroConsultants.com](mailto:sina@FluoroConsultants.com))

President, FluoroConsultants Group, LLC

Chadds Ford, PA, USA

[www.FluoroConsultants.com](http://www.FluoroConsultants.com)

The **PDL Handbook Series** is aimed at a wide range of engineers and other professionals working in the plastics industry, and related sectors using plastics and adhesives.

PDL is a series of data books, reference works and practical guides covering plastics engineering, applications, processing, and manufacturing, and applied aspects of polymer science, elastomers and adhesives.

**Recent titles in the series**

Biopolymers: Processing and Products, Michael Niaounakis (ISBN: 9780323266987)

Biopolymers: Reuse, Recycling, and Disposal, Michael Niaounakis (ISBN: 9781455731459)

Carbon Nanotube Reinforced Composites, Marcio Loos (ISBN: 9781455731954)

Extrusion, 2e, John Wagner & Eldridge Mount (ISBN: 9781437734812)

Fluoroplastics, Volume 1, 2e, Sina Ebnesajjad (ISBN: 9781455731992)

Handbook of Biopolymers and Biodegradable Plastics, Sina Ebnesajjad (ISBN: 9781455728343)

Handbook of Molded Part Shrinkage and Warpage, Jerry Fischer (ISBN: 9781455725977)

Handbook of Polymer Applications in Medicine and Medical Devices, Kayvon Modjarrad & Sina Ebnesajjad (ISBN: 9780323228053)

Handbook of Thermoplastic Elastomers, Jiri G Drobny (ISBN: 9780323221368)

Handbook of Thermoset Plastics, 2e, Hanna Dodiuk & Sidney Goodman (ISBN: 9781455731077)

High Performance Polymers, 2e, Johannes Karl Fink (ISBN: 9780323312226)

Introduction to Fluoropolymers, Sina Ebnesajjad (ISBN: 9781455774425)

Ionizing Radiation and Polymers, Jiri G Drobny (ISBN: 9781455778812)

Manufacturing Flexible Packaging, Thomas Dunn (ISBN: 9780323264365)

Plastic Films in Food Packaging, Sina Ebnesajjad (ISBN: 9781455731121)

Plastics in Medical Devices, 2e, Vinny Sastri (ISBN: 9781455732012)

Polylactic Acid, Rahmat et. al. (ISBN: 9781437744590)

Polyvinyl Fluoride, Sina Ebnesajjad (ISBN: 9781455778850)

Reactive Polymers, 2e, Johannes Karl Fink (ISBN: 9781455731497)

The Effect of Creep and Other Time Related Factors on Plastics and Elastomers, 3e, Laurence McKeen (ISBN: 9780323353137)

The Effect of Long Term Thermal Exposure on Plastics and Elastomers, Laurence McKeen (ISBN: 9780323221085)

The Effect of Sterilization on Plastics and Elastomers, 3e, Laurence McKeen (ISBN: 9781455725984)

The Effect of Temperature and Other Factors on Plastics and Elastomers, 3e, Laurence McKeen (ISBN: 9780323310161)

The Effect of UV Light and Weather on Plastics and Elastomers, 3e, Laurence McKeen (ISBN: 9781455728510)

Thermoforming of Single and Multilayer Laminates, Ali Ashter (ISBN: 9781455731725)

Thermoplastics and Thermoplastic Composites, 2e, Michel Biron (ISBN: 9781455778980)

Thermosets and Composites, 2e, Michel Biron (ISBN: 9781455731244)

To submit a new book proposal for the series, or place an order, please contact David Jackson, Acquisitions Editor

[david.jackson@elsevier.com](mailto:david.jackson@elsevier.com)

# **Mechanics of Solid Polymers**

## **Theory and Computational Modeling**

**Jörgen Bergström**



**ELSEVIER**

AMSTERDAM • BOSTON • HEIDELBERG • LONDON  
NEW YORK • OXFORD • PARIS • SAN DIEGO  
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

William Andrew is an imprint of Elsevier



William Andrew is an imprint of Elsevier  
525 B Street, Suite 1800, San Diego, CA 92101-4495, USA  
32 Jamestown Road, London NW1 7BY, UK  
225 Wyman Street, Waltham, MA 02451, USA

First edition 2015

Copyright © 2015 Elsevier Inc. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: [www.elsevier.com/permissions](http://www.elsevier.com/permissions).

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

#### Notices

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

#### British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library

#### Library of Congress Cataloging-in-Publication Data

A catalog record for this book is available from the Library of Congress

ISBN: 978-0-323-31150-2

For information on all William Andrew publications  
visit our website at [store.elsevier.com](http://store.elsevier.com)

*Publisher:* Matthew Deans

*Acquisition Editor:* David Jackson

*Editorial Project Manager:* Peter Gane

*Production Project Manager:* Melissa Read

*Designer:* Greg Harris

Printed and bound in the United States of America

15 16 17 18 19 10 9 8 7 6 5 4 3 2 1



Working together  
to grow libraries in  
developing countries

[www.elsevier.com](http://www.elsevier.com) • [www.bookaid.org](http://www.bookaid.org)

# Preface

---

The intention of this book is to create a modern review of polymer mechanics theory. This includes explaining how experimental characterization and material modeling are interconnected and can be used to guide accurate finite element predictions of all types of polymers (elastomers, thermoplastics, thermosets, biomaterials, etc.). After finishing my Ph.D. research at MIT in the area of modeling the large strain time-dependent response of elastomers, I have spent the last 16 years as a technical consultant helping companies better understand their polymer products and become more competitive. As part of this effort, I realized early on that there is a huge need for better training and computational tools when it comes to understanding the commonly observed non-linearities of polymer behaviors. To help fill this gap I created the popular website <https://PolymerFEM.com>, which is a free forum for advanced testing and modeling of polymers. I have also given a large number of short courses for professional designers, engineers, and material scientists in various topics related to computational polymer mechanics. This book is the result of combining all of these different areas into a text that is suitable for students, researchers, and industrial engineers.

The topics that are covered in this book include experimental testing, simple material models (e.g. hyperelasticity, linear viscoelasticity, plasticity), and advanced non-linear viscoplastic models. For each topic, details of the theory are presented, and many examples of when and how the models can be applied to solve real problems are shown. As sometimes the theory can seem abstract, I have included a large number of code examples (mainly in the scripting language Python) illustrating in a concrete way the essence of the equations. All of the code is provided on the website for the book: <http://PolymerMechanics.com>.

The overall goal for this book is to provide essential information about how different polymers behave and how their mechanical response can be represented in a finite element simulation.

This field of research is now sufficiently mature that virtually all polymers can be accurately simulated, if you use an appropriate material model. Finally, as I say at the end of all my classes, I work in this field because I really enjoy it. I encourage you to reach out to me if you have any questions or comments.

Jörgen Bergström, Ph.D.

# 1 Introduction and Overview

---

## Chapter Outline

1.1 Introduction	1
1.2 What Is a Polymer?	2
1.3 Types of Polymers	4
1.4 History of Polymers	7
1.5 Polymer Manufacturing and Processing	11
1.6 Polymer Mechanics	11
1.7 Exercises	15
References	16

## 1.1 Introduction

Polymers are a broad class of materials that include traditional engineering materials such as elastomers (rubbers), thermoplastics, and most types of adhesives. In addition to these man-made materials, many natural and biological substances are also polymers, for example, DNA, protein, skin tissue, hair, and many more. Although these materials clearly behave differently from each other—based on our every-day experience—they have many important features in common with respect to their mechanical response. The goal of this text is to outline these characteristic features, and specify different ways that the mechanical response can be predicted using analytical or computational tools.

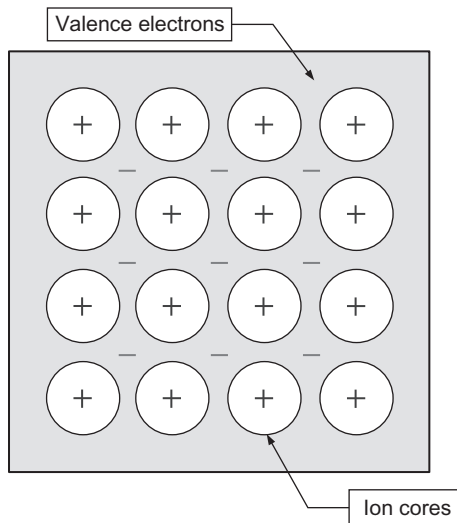
Before embarking on a detailed discussion of these topics it is useful to have a basic understanding of the history of polymeric materials, and knowledge about the fundamentals of polymer processing and polymer mechanics. This chapter lays the foundation for the analysis that will follow and presents definitions and terminology that are needed for the following discussions.

## 1.2 What Is a Polymer?

To answer this question it is enlightening to first ask the complimentary question: what is not a polymer? It turns out that all solid materials can be classified into one of three basic types: metals, ceramics, or polymers. In addition to these basic material types, there are also two groups that can be considered subsets or combinations of these types: composite materials and semi-conductors.

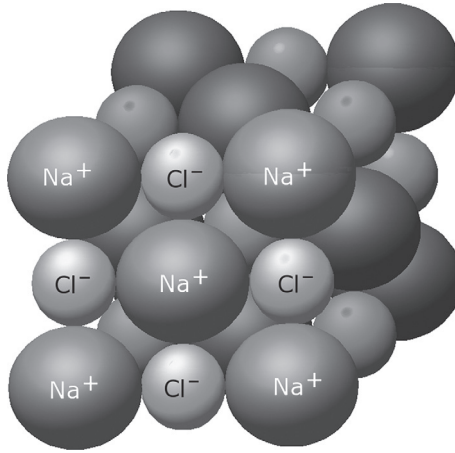
Let us start by considering metals: a metal is a material in which the atoms are held together by metallic bonds (Figure 1.1). It is the delocalized electrons and the strong interaction forces between the positive atom nuclei that give the characteristic response of metals, such as good thermal and electrical conduction. The metallic bonds allow for relative sliding of large groups of atoms, enabling plastic deformation and ductility [1].

Similarly, a ceramic material can be defined as a material in which the atoms are held together by ionic bonds created by positive and negatively charged ions (Figure 1.2). Many ceramic



**Figure 1.1** The atoms in a metal are held together by metallic bonds. The valence electrons are released from the atom nuclei and form an electron cloud.



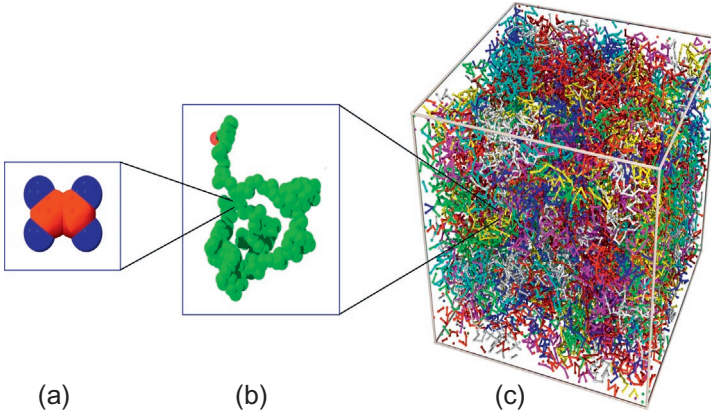


**Figure 1.2** In a ceramic material, the atoms are held together by ionic bonds created by charged ions.

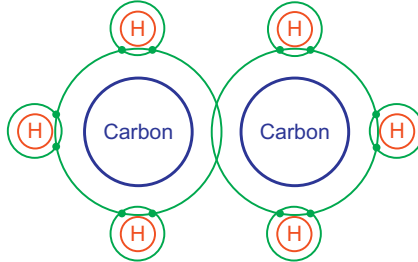
materials have excellent stiffness and compressive strength properties, particularly at high temperatures [2]. In ceramic materials, the electrons are tightly held giving poor conduction of heat and electricity.

The third major material type is polymers. As an informal definition, a polymer is a material with many different length scales (Figure 1.3).

On the most local scale, the atoms are arranged into monomer units and bonded together using covalent bonds. The monomer units are then connected into long chain-like structures. The different macromolecules (sometimes called chain molecules) can be arranged into a network structure by crosslinks or entanglements, and they interact by weak van der Waals forces. The atoms of the polymer backbone are held together by covalent bonds that share electrons between atoms resulting in very strong bonds with very little electron mobility (Figure 1.4). Polymers are therefore typically poor conductors of heat and electricity. The weak bonds between the molecules create very interesting mechanical properties characterized by low stiffness and high ductility. The details of these characteristic behaviors and how they can be modeled are given in the following chapters of this book.



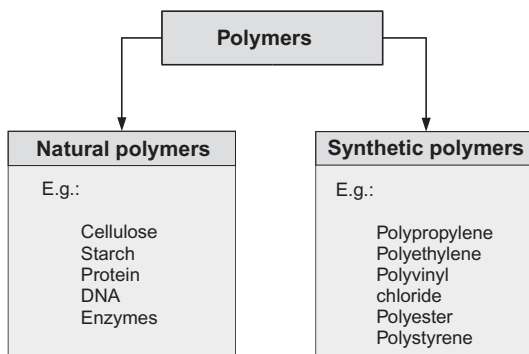
**Figure 1.3** The microstructure of a polymer contains many different length scales. On the most local level, the atoms are arranged into monomer units (a). The monomers are linked into chain molecules (b), which form a macromolecular network (c).



**Figure 1.4** The atoms in a polymer molecule are held together by covalent bonds.

### 1.3 Types of Polymers

Due to the wide variety of polymers and polymer behaviors it is often useful to categorize polymers into different groups. One approach is to distinguish between natural polymers and synthetic polymers (Figure 1.5). Natural polymers, also called biopolymers, include a vast selection of materials. For example, all plants and animals are largely made from biopolymers. Plants are typically made from cellulose (e.g., cotton and wood) or starch (e.g., potatoes and carrots) both of which are polysaccharides. Another common natural polymer is protein, which is formed

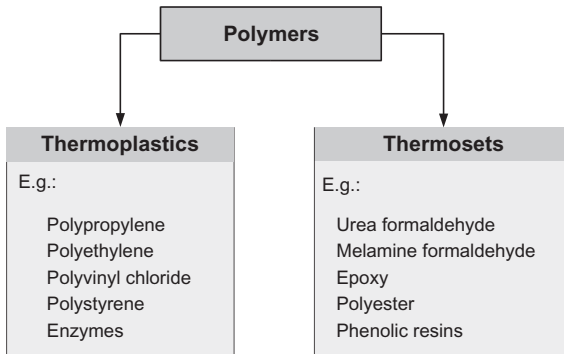


**Figure 1.5** Examples of natural and synthetic polymers.

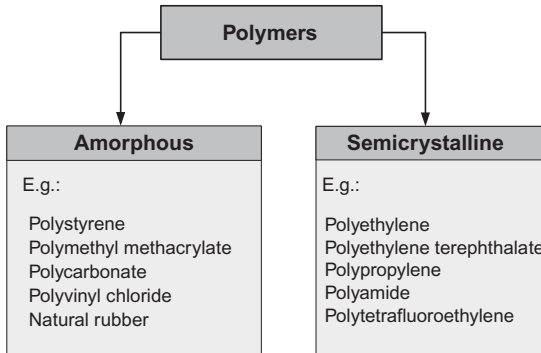
from amino acids. Other examples of biopolymers are: DNA, RNA, peptides, enzymes, skin, hair, silk, and chitin. Man-made polymers, also called synthetic polymers, include most traditional engineering polymers such as polypropylene (PP), polyethylene (PE), and nitrile rubber.

Another useful approach to categorize polymers is to distinguish between thermoplastics and thermosets (Figure 1.6). A thermoplastic is a polymer that is not permanently crosslinked and that softens and can be reshaped when heated. Thermoplastics can generally be exposed to repeated temperature cycles without undergoing significant degradation, making them suitable for recycling. A thermoset is a polymer that is crosslinked (cured) through the addition of energy, typically in the form of heat or irradiation. During the curing process the macromolecules are crosslinked and permanently included in a molecular network structure. Thermosets are generally stiffer and stronger than thermoplastics, but cannot be reshaped or melted.

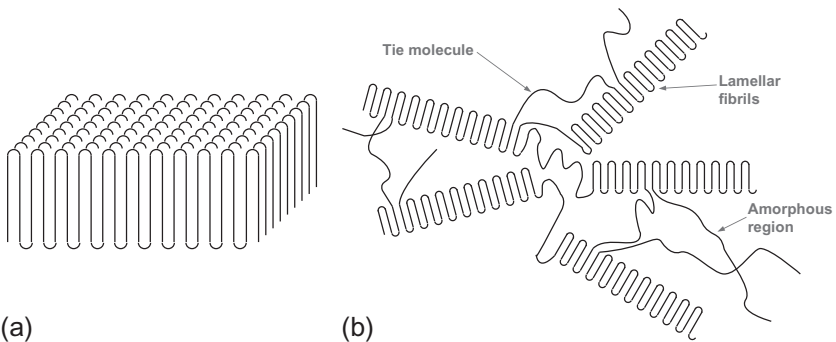
A third way to distinguish polymers is to separate amorphous and semicrystalline polymers (Figure 1.7). The polymer molecules in an amorphous polymer form an entangled network that is characterized by randomness and lack of long-range structure. In a semicrystalline polymer, parts of the molecular structure are crystalline and other parts are amorphous. The crystalline structure is typically considered to consist of layered lamellar crystals (Figure 1.8). On a larger scale, the amorphous and crystalline phases often aggregate to form supramolecular



**Figure 1.6** Examples of thermoplastics and thermosets.



**Figure 1.7** Examples of amorphous and semicrystalline polymers.



**Figure 1.8** (a) Crystallizing polymers from layered structures called lamellae; (b) spherulines are supramolecular structures containing both lamellae and amorphous tie regions.

spherulites. When the crystallization occurs in the absence of flow or mechanical deformation, spherulites is the most common form of the crystal structure. The kinetics of polymer crystallization is very complex and still an active area of research. Amorphous and semicrystalline polymers often exhibit different mechanical behavior. Semicrystalline polymers have a true melting temperature ( $T_m$ ) at which the crystalline domains break up and become disordered. Amorphous polymers do not have a melting temperature but soften significantly above their glass transition temperature ( $T_g$ ). At temperatures above  $T_g$ , large segmental motions are activated and the polymer starts to behave liquid-like.

During the last few years it has started to become more important for the polymer industry to provide products that are sustainable and bio-friendly [3]. This trend is driven by a global drive and emphasis on sustainable technology. There are currently many different polymer products available that are based on renewable resources and that are easily recyclable and renewable. Polylactic acid, PLA, is one example. This polymer is synthesized from corn, and it biodegrades after use if composted. In many applications, PLA is an alternative to PET and PVC, and it is currently used, for example, in candy wrap, optical films, and shrink labels. The main disadvantages of PLA include slightly higher density and a higher price than some traditional engineering polymers.

## 1.4 History of Polymers

Since all biological tissues are polymers (e.g., skin tissue, cartilage, tendons, etc.), the history of polymers is as old as the history of these materials. Estimates indicate that the age of these materials are millions of years old [4]. Natural evolution for this time period has led to very specialized biopolymers with truly amazing mechanical properties that often even today are difficult to match or replace by synthetic polymers (Table 1.1).

Natural polymers are not only occurring as biological tissue, but also as cotton, starch, and cellulose. For example, it has been known for about 500 years that *Hevea brasiliensis* produces a

**Table 1.1 Brief Summary of the History of Polymers [5–7]**

<b>Date</b>	<b>Polymer Development</b>
1500s	British explorers find Mayan civilizations that use natural rubber from rubber trees
1839	Charles Goodyear discovers vulcanization of natural rubber by adding sulfur and heating to 130 °C (270 °F)
1862	Alexander Parks invents the first man-made polymer, named Parkesine
1907	Leo Bakelite invents bakelite, the first completely synthetic plastic. This material is a stiff thermoset polymer with good heat resistance, useful for electrical insulation
1925	X-ray crystallography shows, for the first time, that polymers are made of long macromolecules
1927	Large-scale production of PVC begins
1930	Polystyrene (PS) is invented
1933	Polyethylene (PE) is invented
1938	Polyamide and fluoropolymers are invented at DuPont
1971	Kevlar is invented
1976	Polymers surpass steel as the most used engineering material by volume
Late 1970s	Conductive polymers are invented
Early 2000s	Use of synthetic polymers in biomedical applications

substance with quite useful properties. This material, also known as natural rubber, was known by the Mayan civilization to be a useful material for making tools and games.

It was not until 1839, when Charles Goodyear discovered vulcanization, that natural rubber became an important engineering

material. Vulcanized natural rubber has very good elastic and resilient properties, and it still remains one of the most useful and important rubbers for industrial applications. For example, car tires typically contain a large volume fraction of natural rubber. The first man-made polymer was developed by Alexander Parks in 1862. He developed an organic material, referred to as Parkesine, which could be reshaped when heated and remained stiff when cooled. The material, however, was not a big hit due to its high cost. The first completely synthetic polymer was Bakelite, developed by Leo Bakelite in 1907. This material provided significant advantages over many other contemporary materials due to its good electrical insulating properties and ability to be molded into complicated shapes. It is interesting to note that the macromolecular microstructure of polymers was not understood until 1925. Before that time, different theories had been presented related to what caused the interesting behavior of Bakelite, but the microstructure was not known. After this discovery, and the realization that it is the macromolecular structure that creates the unique behavior of polymers, a large number of common traditional engineering polymers were developed during the following 10 years. For example, PVC, PS, PE, Nylon, and PTFE were discovered during the late 1920s and during the 1930s. Since that time the use of polymers has continued to grow, and the number of different engineering polymers has continued to increase. Some of the new materials have focused on improved performance. For example, polyethyleneimine, PEI, and polyether ether ketone (PEEK) are polymers with high thermal stability, strength, and stiffness.

One interesting milestone was passed in 1976 when the produced volume of polymers for the first time exceeded the produced volume of steel. Since then, the polymer use and technology has continued to grow. For example, the first conductive polymer was discovered in the late 1970s. Conductive polymers are important because they can potentially be used to create lightweight batteries, solar cells, etc. During the last few years synthetic polymers have started to be used in biomedical applications. Artificial skin and bone, drug delivery devices, and scaffolds for growing transplant implants are just a few examples.

Also, recently a significant effort has been directed toward improving the sustainability of polymer materials to make them bio-friendly. For example, there are thermoplastic elastomers that do not require aggressive chemistry during crosslinking. Another recent development of high importance is the discovery of conducting polymers. By combining the traditional strengths of polymers: low cost and easy manufacturing; with conductive properties it is possible to create organic light-emitting diodes (LEDs), lasers, and many other unique products.

The bulk of the produced polymers today are traditional engineering polymers (Table 1.2). The polymer with the highest production volume is polypropylene (PP). The polymer with the second highest production volume is HDPE. However, if the different types of PE (i.e., HDPE, LLDPE, and LDPE) are combined, then they constitute about half of the total production volume. Today polymers are ubiquitous, and one of the most important materials in our everyday life.

**Table 1.2 Production Volume of Different Polymers in North America During 2003 [5, 6]**

<b>Resin</b>	<b>Production Volume (kg)</b>
Polypropylene (PP)	$8.0 \times 10^9$
High density PE (HDPE)	$7.1 \times 10^9$
Polyvinyl chloride (PVC)	$6.7 \times 10^9$
Linear low density PE (LLDPE)	$5.0 \times 10^9$
Natural rubber (NR)	$4.9 \times 10^9$
Styrene-butadiene rubber (SBR)	$3.8 \times 10^9$
Thermosets	$3.7 \times 10^9$
Low density PE (LDPE)	$3.5 \times 10^9$
Thermoplastic polyester	$3.4 \times 10^9$
Polystyrene (PS)	$2.9 \times 10^9$
Acrylonitrile butadiene styrene (ABS)	$0.5 \times 10^9$
Polyamide (Nylon)	$0.5 \times 10^9$



## 1.5 Polymer Manufacturing and Processing

Many different manufacturing procedures can be used to create man-made polymer materials and products. The macromolecular structure of the raw material is created by chemically joining together monomers either by addition polymerization or condensation polymerization.

In *addition polymerization* molecules with double or triple chemical bonds are linked together into macromolecules. The unsaturated monomers are linked up with other monomers to create the molecular chains. This type of polymerization is typically used to create polyethylene, propylene, and polyvinylchloride. The reactions are carried out in the presence of a catalyst that also can influence the structure and organization of the material.

*Condensation polymerization* is based on a different mechanism in which monomers bond together by condensation reactions. The chemical reactions are typically achieved through alcohol, amine, or carboxylic acid functional groups. In each step of this process a simple compound, often water, is formed. This process is used when amino acids link up to form proteins. It is also the way in which Kevlar is formed.

The required processing to create final products of the polymeric raw materials is an important field that has been extensively studied [8–15]. Some of the more common processing operations are: blow molding, extrusion, fiber spinning, filament winding, film blowing, injection molding, pultrusion, reaction injection molding, spin coating, and transfer molding. Further information on these processing operations can be found in numerous textbooks geared toward the specific subject matter.

## 1.6 Polymer Mechanics

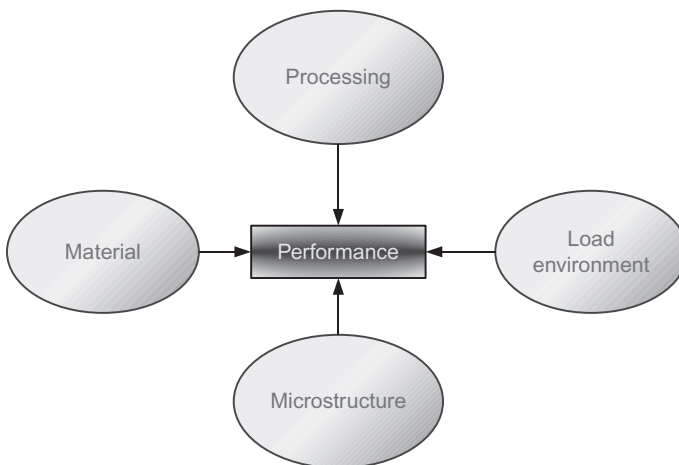
The previous few sections have presented different aspects of polymer history, chemistry, and manufacturing. This section gives an overview to the main topic of this book: polymer mechanics. By definition, polymer mechanics involve the mechanical behavior of solid polymers. This field of study can be divided into

many different subtopics. The following are examples of common questions that a polymer mechanist is often interested in:

- What are the different ways a particular polymer can behave in a certain load environment?
- What is the best way to experimentally characterize the mechanical response of a polymer?
- How well will a certain polymer product behave? How can the performance be improved?
- Which polymer is most appropriate for a given application?

In order to address these questions, it is important to understand the connection between polymer material, processing conditions, material microstructure, and load environment; and how all of these factors influence the performance of the polymer product (Figure 1.9).

One topic of polymer mechanics involves the tools and techniques that are used for experimental characterization of polymeric materials. Another and complimentary topic is related to theoretical predictions, either by traditional analysis or computer

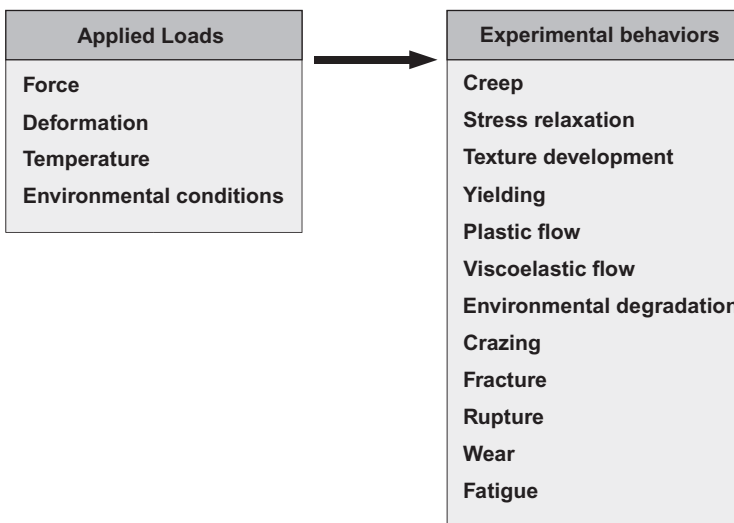


**Figure 1.9** The performance of a polymer product is governed by the material type, the processing conditions, the material microstructure, and the applied load environment.

simulations. Due to the complexity of the polymer microstructure, the use of computer simulations is becoming increasingly important when studying the behavior of solid polymers. This approach to study the material response is often called *computer experiments* due to the possibility to replace time-consuming and costly physical experiments with computer simulations.

The goal of polymer mechanics is to develop an understanding of the mechanical behavior of polymers and to develop tools for predicting the observed mechanical response of polymers in different load environments. In this context, polymer mechanics is not directly involved with the chemistry behind how polymers are created, or the processing steps that are needed to create a polymer component. The main reasons for this distinction are that the chemistry and rheology involved in the manufacturing process are significantly different topics than the study of the mechanics of solid polymers. Polymer production, manufacturing, and processing are discussed in depth in many textbooks [8–16].

Polymers, due to their molecular structure, exhibit many different types of phenomena when exposed to normal loading environments. Some of the more common behaviors are listed in [Figure 1.10](#).



**Figure 1.10** The response of polymers when subjected to external loads can be of many different characteristic types.

As illustrated in this figure, the response of polymers can take many different forms when exposed to an external load environment. Some of the more common and important behaviors are: creep, stress relaxation, anisotropy, texture development, yielding, plastic flow, viscoelastic flow, environmental degradation, crazing, fracture, rupture, wear, and fatigue. The goal of this book is to address these issues. Specific emphasis will be placed on why these phenomena occur, and how they can be predicted and computationally modeled.

There are two types of modeling approaches that aim at capturing these phenomena. The first approach, and the classical approach, is to use a phenomenological experience-based model. As an example consider an attempt to predict the tensile failure of polycarbonate under monotonic uniaxial loading. By performing a sufficient number of experimental tests it might be possible to prove a significant correlation between the Mises stress and the observed onset of failure. A polymer mechanist might then take this as an indication that Mises stress is a good failure predictor for polycarbonate, using the hypothesis that failure occurs when the Mises stress in a specimen (obtained either by Finite Element Modeling (FEM) or direct closed-form calculations) exceeds the critical Mises stress, where the critical Mises stress is a material parameter. The astute reader may realize that this may not be a good model. For example, can this failure criterion be used if the polymer component was deformed 100 times faster? What if the temperature is different? What if the component is deformed in simple shear to failure? Is the Mises stress criterion still valid under these conditions? All of these questions are serious and need to be carefully addressed before the proposed failure criterion can be considered a general failure criterion. This highlights the main limitation of phenomenological models:

*A phenomenological model is strictly only applicable for the exact loading conditions for which it was validated.*

This limitation is often somewhat neglected in industrial settings where the need to reach an answer in a short amount of time and with a limited budget is often driving the analysis.

The second modeling approach for capturing polymer mechanics phenomena is to use a micromechanical model. Micromechanical models, by definition, are models that use information and knowledge about the microstructure of the material as the basis for the model. Imagine for a moment that we have a model that relates the applied deformation to the average molecular chain stretch on a molecular level. That model would then be more reliable for predicting the response in general loading modes and temperatures. Micromechanical models are almost always preferred over phenomenological models, but due to the complexity of the deformation characteristics of the molecular microstructure it is often very difficult to develop models that are purely micromechanical. The idea of the micromechanical models is to bridge different length scales: if we understand the mechanisms driving the deformation behavior on a micro (or nano) scale, then how can we use that information to predict the behavior on a macro or continuum scale? Due to the challenge of translating information from different length scales, a number of models have been developed that combine elements from the micromechanics of deformation and phenomenological approaches. These models are often called *micromechanism inspired models* and are currently for some of the phenomena presented in [Figure 1.10](#) the most accurate approach that is currently available.

The remaining chapters of this book address these different aspects of polymer mechanics, demonstrating when the different phenomena occur and how they can be predicted.

## 1.7 Exercises

1. What are the three major classes of solid materials?
2. How do you define a polymer and what makes a polymer unique?
3. Give a few examples of different polymers.
4. When was the first man-made polymer created?
5. What are the two major ways in which polymer materials can be created?

6. If a polymer product is exposed to an external load environment, what are the different ways in which the polymer might behave?
7. What are the differences between a phenomenological and micromechanical model?

## References

- [1] M.F. Ashby, D.R. Jones, *Engineering Materials: An Introduction to Their Properties and Applications*, Pergamon, Oxford, 1980.
- [2] N.E. Dowling, *Mechanical Behavior of Materials: Engineering Methods for Deformation, Fracture, and Fatigue*, Prentice Hall, Upper Saddle River, NJ, 1998.
- [3] E. Chiellini, H. Gil, J. Burchert, P. Gateholm, M. van der Zee, *Biorelated Polymers—Sustainable Polymer Science and Technology*, Kluwer Academic Publishers, New York, 2001.
- [4] *Encyclopedia of Life Sciences*, Nature Publishing Group, 2004 (online edition).
- [5] American Plastics Council. <http://AmericanPlasticsCouncil.org>. 2015 (accessed April 2015).
- [6] *Worldwide Rubber Statistics*, International Institute of Synthetic Rubber Producers, Houston, TX, 2004.
- [7] *Encyclopedia of Polymer Science and Technology*, John Wiley & Sons, 2004 (Online edition ISBN: 9780471440260).
- [8] D.V. Rosato, *Filament Winding: Its Development, Manufacture, Applications, and Design*, Interscience Publishers, New York, 1964.
- [9] C. Rauwendaal, *Polymer Extrusion*, Hanser Publisher, Munich, 1990.
- [10] A. Ziabicki, *Fundamentals of Fiber Formation*, Wiley, New York, 1976.
- [11] J.F. Agassant, P. Avenas, J.Ph. Sergent, *Polymer Processing Principles and Modeling*, Hanser, 1986.
- [12] G. Potech, *Injection Molding: An Introduction*, Hanser Publishers, Munich, 1995.
- [13] I. Manas-Zloczower, Z. Tadmor, *Mixing and Compounding of Polymers*, Hanser, New York, 1994.
- [14] C.W. Macosko, *RIM, Fundamentals of Reaction Injection Molding*, Hanser, New York, 1989.

- [15] S. Middleman, A.K. Hochberg, *Process Engineering Analysis in Semiconductor Device Fabrication*, McGraw-Hill, New York, NY, 1993.
- [16] R.B. Bird, C.F. Curtiss, R.C. Armstrong, O. Hassager, *Dynamics of Polymeric Liquids*, Wiley-Interscience, New York, 1987.

# 2 Experimental Characterization Techniques

---

## Chapter Outline

2.1	Introduction	20
2.2	Mechanical Testing for Material Model Calibration	22
2.2.1	Uniaxial Compression Testing	24
2.2.2	Uniaxial Tension Testing	29
2.2.3	Plane Strain Tension	33
2.2.4	Simple Shear Testing	37
2.2.5	Impact Testing	40
2.2.6	Dynamic Mechanical Analysis	43
2.2.7	Hardness and Indentation Testing	47
	Rockwell Hardness Testing	47
	Shore (Durometer) Testing	48
	Barcol Hardness Testing	50
	Nanoindentation	51
2.2.8	Split-Hopkinson Pressure Bar Testing	53
2.2.9	Bulk Modulus Testing	64
2.2.10	Other Common Mechanical Testing Modes	71
2.2.11	Testing for Failure Model Calibration	73
2.3	Mechanical Testing for Material Model Validation	73
2.3.1	Material Model Verification and Validation	75
2.3.2	Small Punch Testing	77
2.3.3	V-Notch Shear Testing	80
2.4	Surface Characterization Techniques	80
2.4.1	Optical Microscopy	81
2.4.2	Scanning Electron Microscopy	84
2.4.3	Atomic Force Microscopy	87
2.5	Volume Characterization Techniques	89
2.5.1	Differential Scanning Calorimetry	89
2.5.2	Transmission Electron Microscopy	90
2.5.3	X-Ray Diffraction	92



Wide-Angle X-Ray Diffraction	93
Small-Angle X-Ray Diffraction	95
2.5.4 Birefringence	95
2.5.5 Swell Testing	97
2.6 Chemical Characterization Techniques	99
2.6.1 Fourier Transform Infrared Spectroscopy	100
2.6.2 Energy Dispersive Spectroscopy	101
2.6.3 Size-Exclusion Chromatography	103
2.6.4 Thermogravimetric Analysis	107
2.6.5 Raman Spectroscopy	109
2.7 Exercises	110
References	112

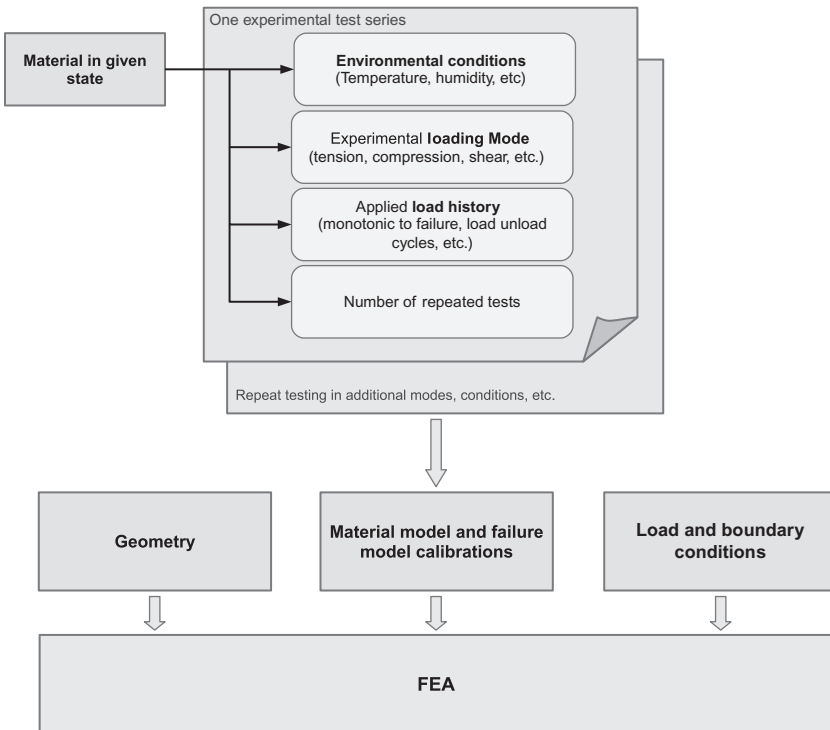
## 2.1 Introduction

One of the key components of polymer mechanics is experimental material characterization. Polymers, due to their macromolecular structure, exhibit not only a wide range of different behaviors but also rapid changes in behavior for limited variations in load environment and temperature. At low temperatures (relative to the glass transition temperature), polymers behave as stiff solids that if deformed enough start to undergo viscoplastic deformation. At temperatures in the vicinity of the glass transition temperature, polymers typically exhibit a leathery or rubbery response. At high temperatures, above both the glass transition temperature and the melting temperature, the response is viscous and liquid-like if the polymer is uncrosslinked, rubbery if the material is weakly crosslinked, and stiff viscoelastic if the material is highly crosslinked. It is important to understand and be able to experimentally determine these characteristic behaviors, and this is the topic of [Section 2.2](#) of this chapter.

Experimental testing can be performed for different reasons. One common reason is to obtain enough information to calibrate one or more material models. Another reason is to obtain information related to when and how a material fails; that is, to obtain information for failure model calibration. Sometimes testing is also performed for quality control purposes, or to characterize

certain aspects of the material response or structure. The focus of this chapter is on experimental testing for material model and failure model calibration, and also on validating already calibrated models. There is an almost unlimited amount of different testing techniques that are available for this purpose, and this chapter is simply focused on presenting some of the more commonly used and some of the most useful methods.

An experimental test program for material model calibration typically consists of multiple individual experiments, each of which is performed in a specific loading mode, with a specific applied stress/strain load history, in a given temperature environment. [Figure 2.1](#) shows a schematic representation of an experimental test program and how it connects to finite element (FE)



**Figure 2.1** An experimental testing program consists of multiple individual experiments each of which needs the specification of the loading mode, the applied load history, the environmental conditions, and the number of repeated tests.

investigations. Much of the materials covered in this chapter are targeted to how to properly design an experimental test program for the purpose of material model calibration and validation.

Another aspect of polymer characterization is related to the *morphology* of the material, either for a surface or for the bulk volume. Experimental techniques designed for this purpose are discussed in [Sections 2.4](#) and [2.5](#), respectively. The goal of these experimental characterization techniques is to probe or investigate different aspects of the microstructure of the material. For example, the presence of microcracks, the degree of crystallinity, and the type of crystal structure can all be examined using these techniques.

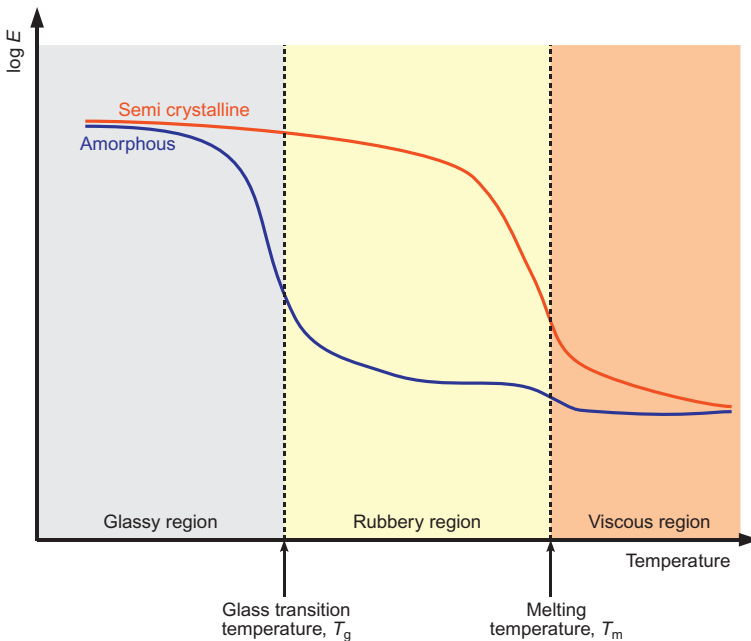
The last section of this chapter ([Section 2.6](#)) discusses chemical characterization techniques that are useful for polymers. The results from these techniques are not useful for phenomenological polymer mechanics modeling, but can be very useful and informative in experimental studies of for example fatigue, and material degradation. These chemical characterization techniques are also immensely important in polymer failure analysis. When faced with mechanical failure or fracture of a polymer product, one of the first investigations that is often done is to make sure the polymer that was used in the product was of the right type, and that the molecular structure of the polymer was what it was supposed to be. The chemical characterization techniques can answer these questions. The information obtained through chemical characterization can also be used to better understand the mechanical behavior of polymers on a microlevel, and as will be discussed in Chapter 8, is useful for developing micromechanical models.

## 2.2 Mechanical Testing for Material Model Calibration

The most common method to experimentally characterize the mechanical response of a polymer involve mechanically loading a polymer component or test specimen, and measuring the resulting force and displacement response. There are many different ways

this can be performed. This section summarizes some of the more common experimental characterization techniques.

One of the most basic mechanical tests—perhaps even the most important polymer mechanics test—is a test designed to determine the stress-strain response of a given polymer. The stress-strain response will dictate the deformation characteristics of the polymer in its intended application, and is therefore a key component for quantifying not only the deformation behavior but also fatigue and fracture resistance. As discussed above, one complication for polymers is that the stress response during an imposed deformation history is strongly dependent on the material, the applied rate of deformation, and the temperature. These factors all play an important role when designing or choosing a mechanical test for polymers. The influence of temperature, in particular, has utmost influence on all aspects of the mechanical response of polymers. One way this is often presented is in terms of the dependence of the material stiffness (Young's modulus) on the temperature, see [Figure 2.2](#).



**Figure 2.2** Dependence of the Young's modulus on the temperature for different classes of polymers.

For example, the glass transition temperature for PET is about 70 °C (343 K) and the melting temperature is about 250 °C (523 K), so at room temperature this polymer will already be at 85% of its glass transition temperature, and 56% of its melting temperature. The thermal activation barrier to viscoelastic and viscoplastic flow is therefore small in many normal applications, resulting in a complicated macroscopic response when exposed to external loads.

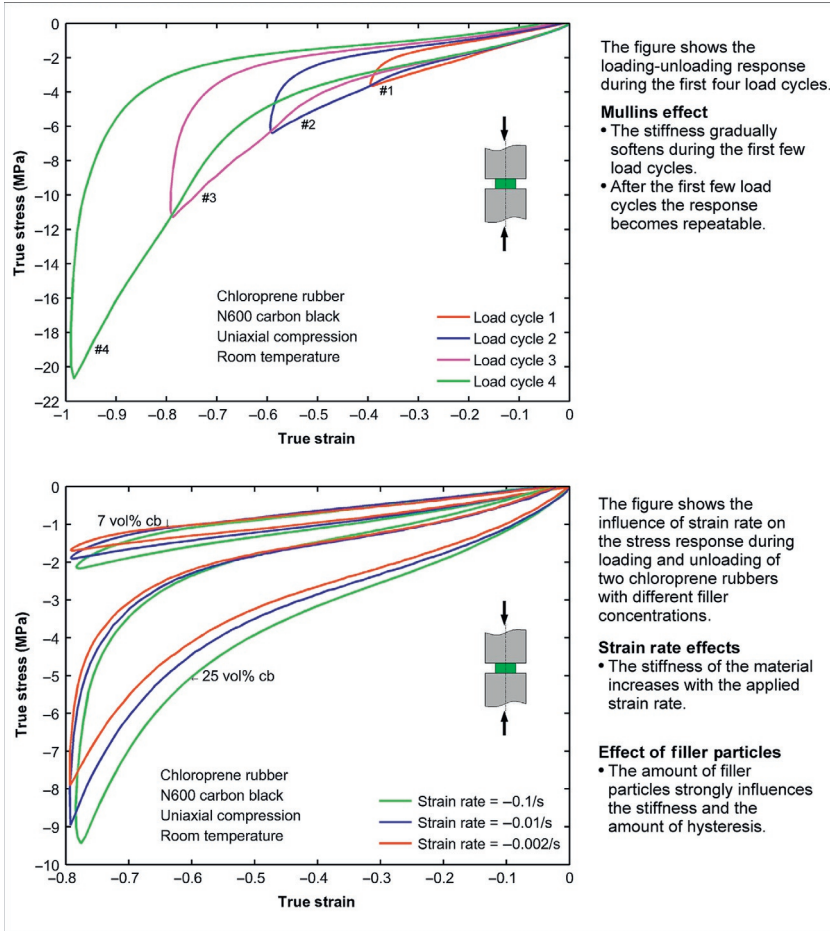
In a broad sense, the stress-strain response of polymers can be divided into three different categories, each with its own distinct features: (1) deformation of elastomers and thermoplastic elastomers (TPE); (2) deformation of thermoplastics below the glass transition temperature; and (3) deformation of highly crosslinked thermosets. Characteristic experimental features of these different categories are illustrated in [Figures 2.3, 2.4, and 2.5](#), respectively.

### 2.2.1 Uniaxial Compression Testing

The most common method to experimentally determine the stress-strain response of a material is to perform uniaxial tension or compression tests. In these tests, the test specimen is loaded in a designated test machine using grips for tension tests, and compression platens for compression tests. The specimen deformation is typically measured using an extensometer or strain gauge, either contacting the specimen or by using optical techniques.

Both uniaxial tension and uniaxial compression tests have strengths and limitations. For the uniaxial compression tests, the interface friction between the test specimen and the loading platens can cause a nonhomogeneous deformation state, often indicated by barreling of the test specimen, see [Figure 2.6](#).

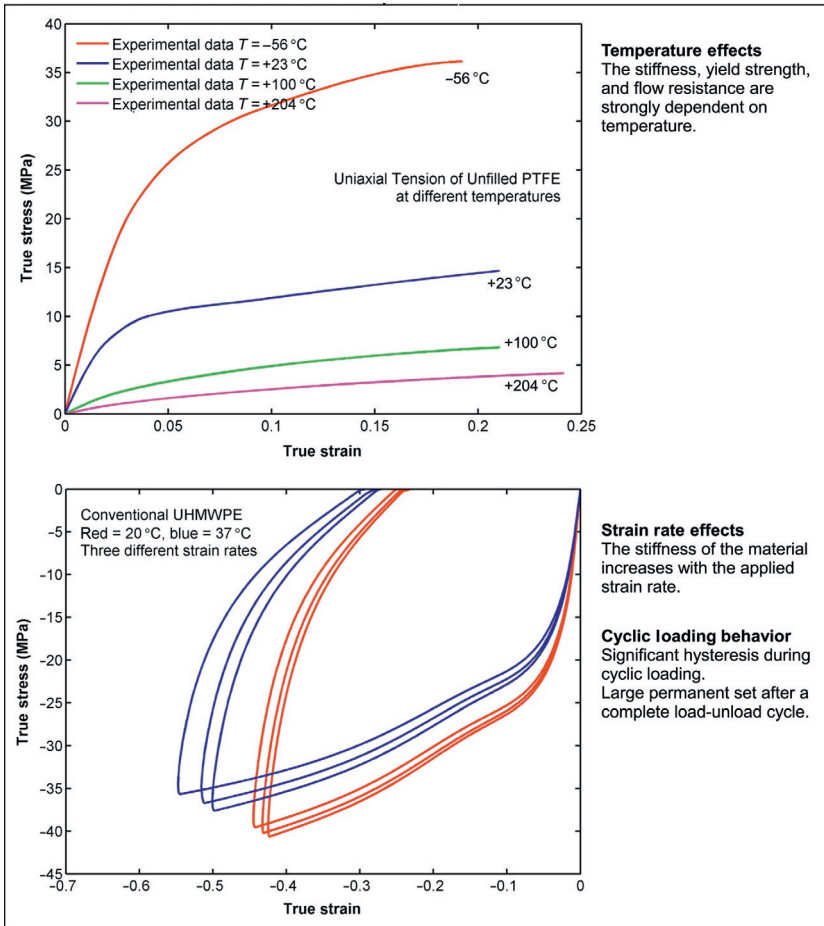
This figure shows FE predictions of both the deformed shape and the change in the measured engineering stress as a function of the friction coefficient and the applied strain. The initial diameter of the test specimen that was used in the virtual experiments was 28.6 mm and the initial height was 12.5 mm. These values follow the recommendations in ASTM D575 [1]. A Neo-Hookean



**Figure 2.3** Characteristic stress-strain response of elastomers loaded above the glass transition temperature.

material model with a shear modulus of 2 MPa and a bulk modulus of 200 MPa was used to represent the polymer. The results clearly show that significant barreling can occur even when the friction coefficient is as low as 0.1, and the error in the stress is not trivial for finite strains.

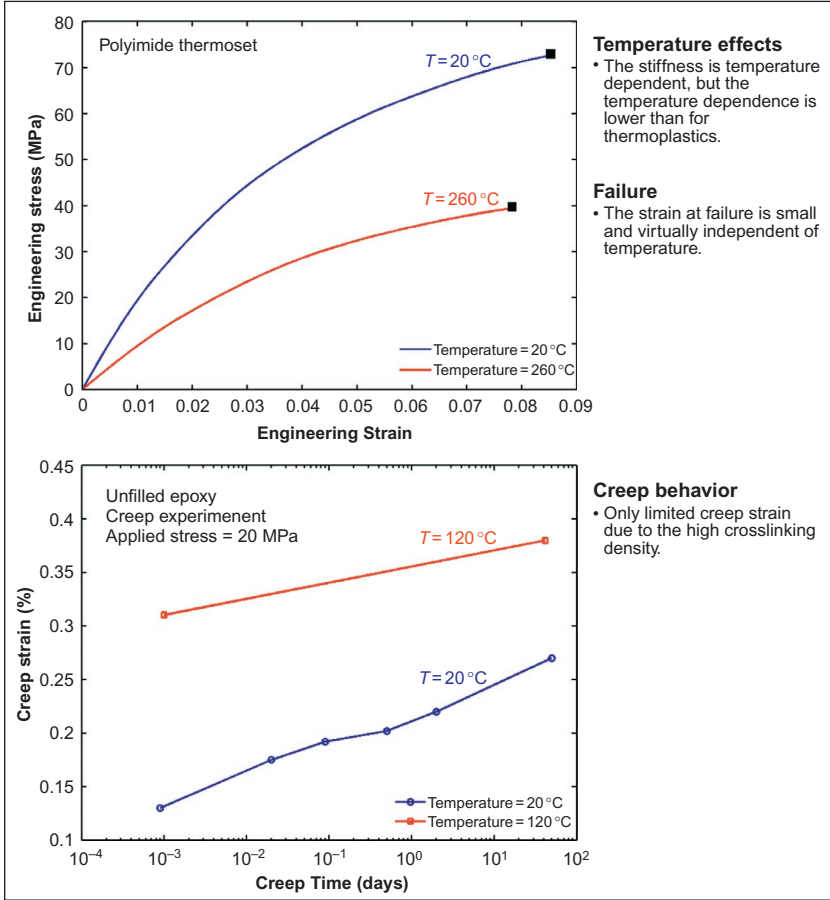
The error in the stress predictions can be obtained by comparing the results from the case with friction to the case without friction. The relative error in the predictions from this test case is plotted in Figure 2.7 showing that the error is almost linearly



**Figure 2.4** Characteristic stress-strain response of thermoplastics below the glass transition temperature.

depend on the friction coefficient and weakly dependent on the applied strain.

The stress and strain inside the test specimen becomes inhomogeneous when there is a nonzero friction coefficient. As an example, [Figure 2.8](#) shows a cut through a specimen with a diameter of 28.6 mm and a height of 12.5 mm (Following ASTM D575 [1]) that was compressed to an engineering strain of  $-0.3$  with a friction coefficient of 0.1. In this case, the max Mises stress is about 4.5 times higher than the lowest Mises stress, which is quite substantial.

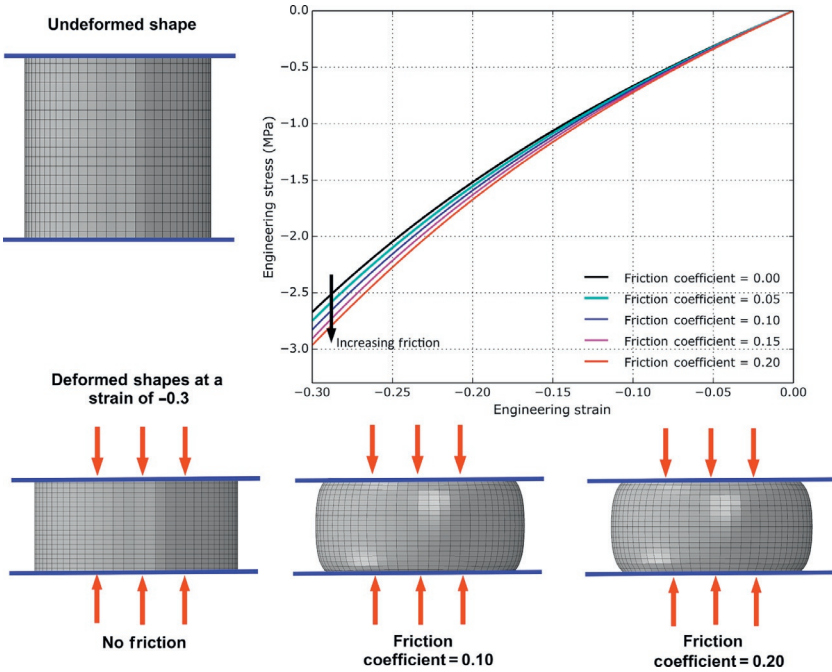


**Figure 2.5** Characteristic stress-strain behavior of thermosets.

One way to reduce the influence of the interface friction is to make the specimen height/diameter ratio larger. Figure 2.9 shows the results from virtual experiments of specimens with different heights but the same diameter. In this case, the relative error in the stress measurement goes down from 11.5% to 3.5%, at an applied strain of  $-0.2$ , if the specimen height to diameter ratio increases from 0.44 to 1.31. It is important, however, not to make the test specimen too tall since that can cause a buckling instability during the compression testing.

Interface friction in a compression experiment influences not only the stiffness during monotonic loading but also the stress

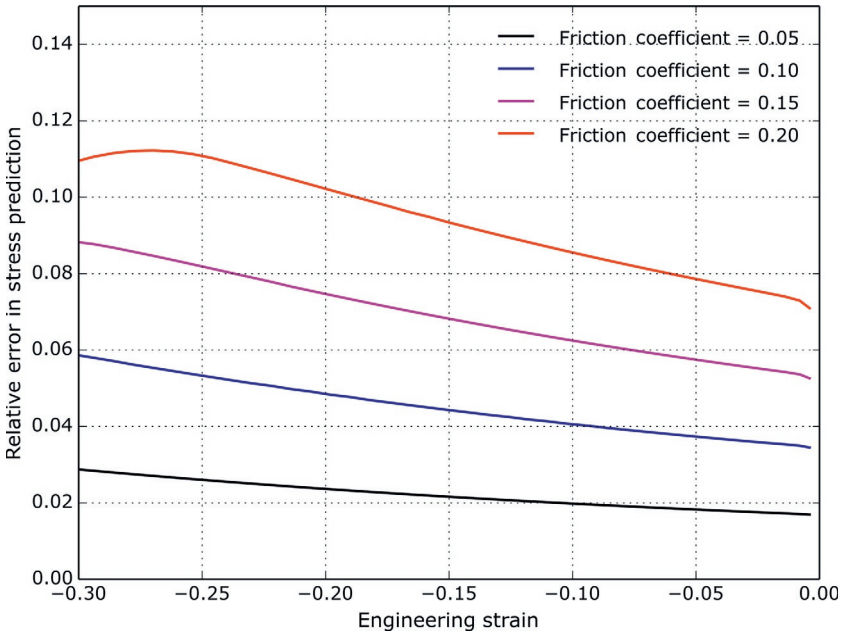




**Figure 2.6** 3D FE results showing the influence of interface friction on the deformed specimen shape and the stress-strain response. The polymer was modeled using a Neo-Hookean material model with a shear modulus of 2.0 MPa and a bulk modulus of 20.0 MPa.

response during unloading. This effect especially changes the response during the initial unloading part of an experiment. This behavior is examined using FE-based virtual experiments summarized in [Figure 2.10](#). As shown in the figure, a simple hyperelastic Neo-Hookean material dissipates energy during cyclic loading when there is interface friction, and a nonlinear viscoelastic material, here represented using the Bergstrom-Boyce (BB) model, changes its behavior during the loading phase and during the initial part of the unloading. The frictional effects specifically increase the slope of the stress-strain curve right after unloading. This behavior is easy to misinterpret as an indication of a Mullins (or other) damage mechanism, but can simply be a consequence of the interface friction.

To reduce the amount of barreling it is common to lubricate the interface between the specimen and the loading platens,



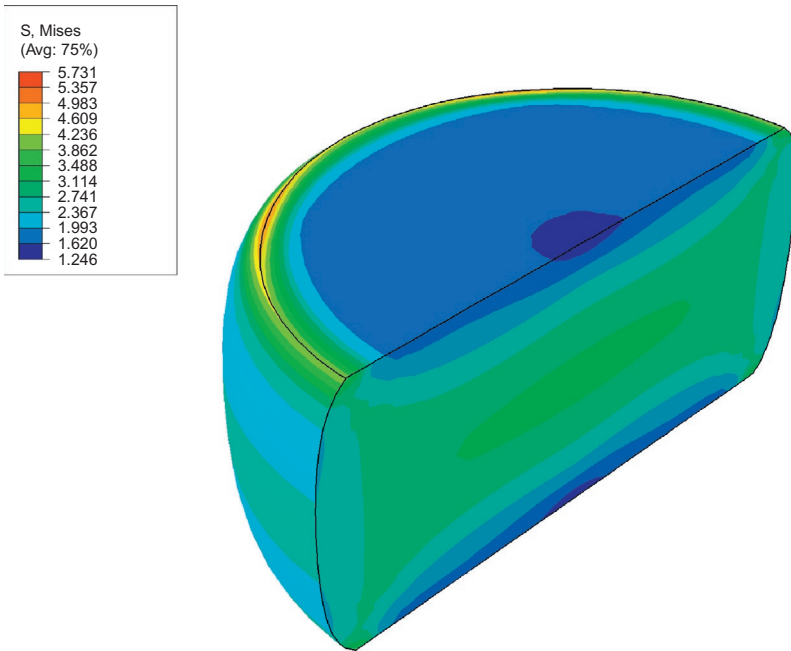
**Figure 2.7** FE results showing the influence of interface friction on the predicted error in the measurement results. The polymer was modeled using a Neo-Hookean material model with a shear modulus of 2.0 MPa and a bulk modulus of 20.0 MPa.

for example by a nonaggressive oil or liquid soap [2]. Another alternative is to use PTFE sheets at the interfaces [2, 3].

If the purpose of the experimental testing is to obtain suitable experimental data for material model calibration, then it is still possible to use compression data as long as the friction coefficient is known or can be estimated. In this case, the material model calibration is best performed using FE simulations of the actual experiments that were formed. This topic is discussed in more detail in Chapter 9.

## 2.2.2 Uniaxial Tension Testing

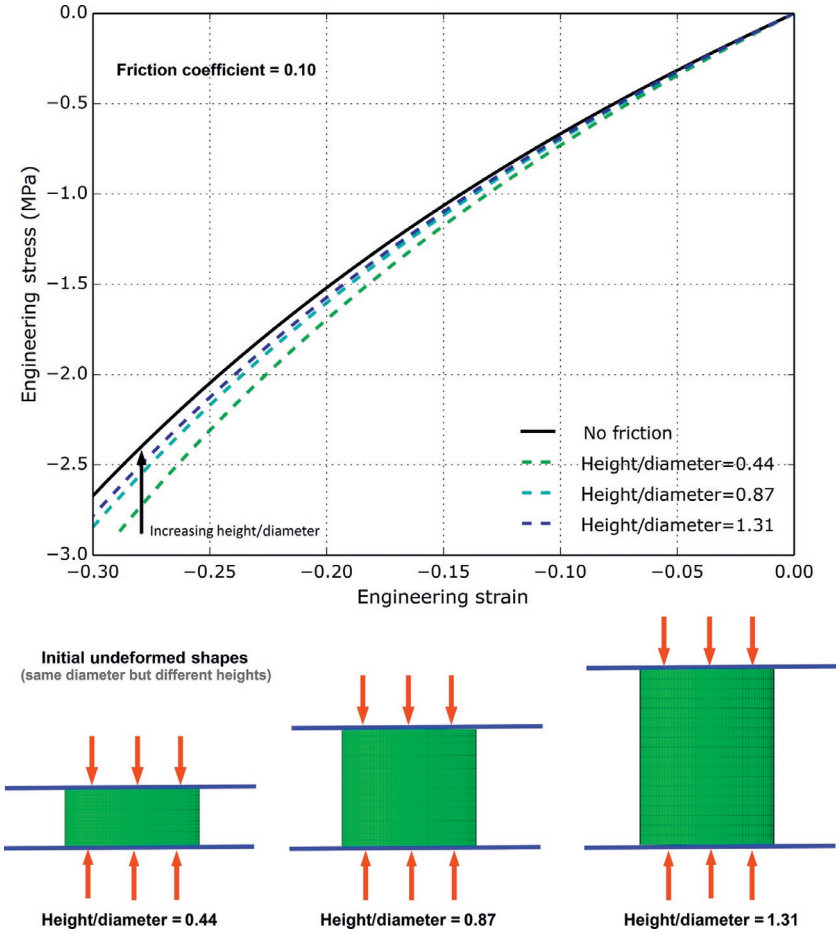
In uniaxial tension testing, the loading grips will always introduce stress concentrations in the part of the specimen that is close to the gripped regions. To reduce the influence of these stresses the test specimens are typically given a dog-bone shape.



**Figure 2.8** Predicted contours of Mises stress. The polymer was modeled using a Neo-Hookean material model with a shear modulus of 2.0 MPa and a bulk modulus of 20.0 MPa. Applied engineering strain =  $-0.3$ , friction coefficient = 0.1.

This specimen design will create an almost uniform uniaxial deformation and stress state in the central gauge region of the specimen. Details of recommended test specimen geometries are given in different standards [4–6]. Figure 2.11 shows the distribution of Mises stress in an ASTM D638 Type IV specimen pulled in tension to a true strain of approximately 15%. In this case, the test specimen was represented using a Neo-Hookean material model with a shear modulus of 2 MPa and a bulk modulus of 200 MPa. The green region in the figure has a stress that is varying less than 1%.

One complication that can occur with dog-bone shaped specimens during tensile loading is *necking*. This is a phenomenon that is important to be aware of since it creates an inhomogeneous deformation state in the specimen, and hence makes it difficult to extract the actual stress-strain response unless the localized deformation state is directly measured using, for example, digital

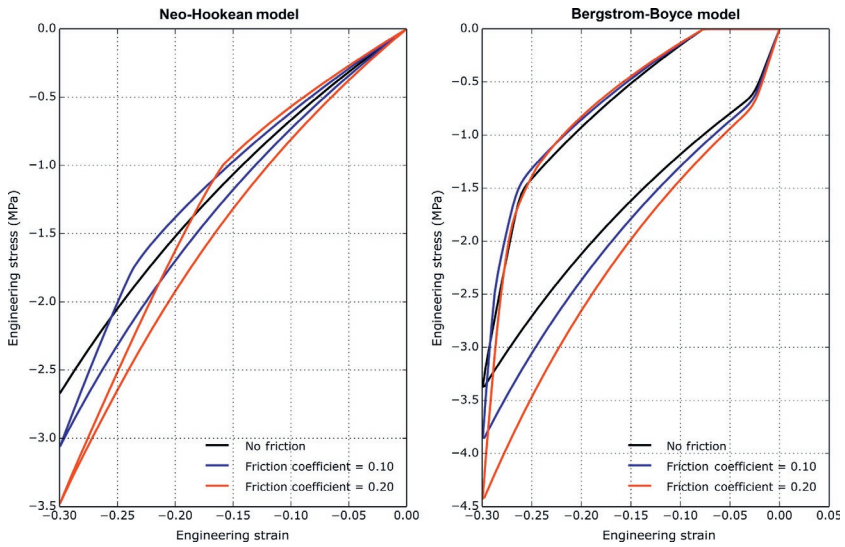


**Figure 2.9** FE results showing the influence of specimen height to diameter ratio on the measurement results. The polymer was modeled using a Neo-Hookean material model with a shear modulus of 2.0 MPa and a bulk modulus of 20.0 MPa.

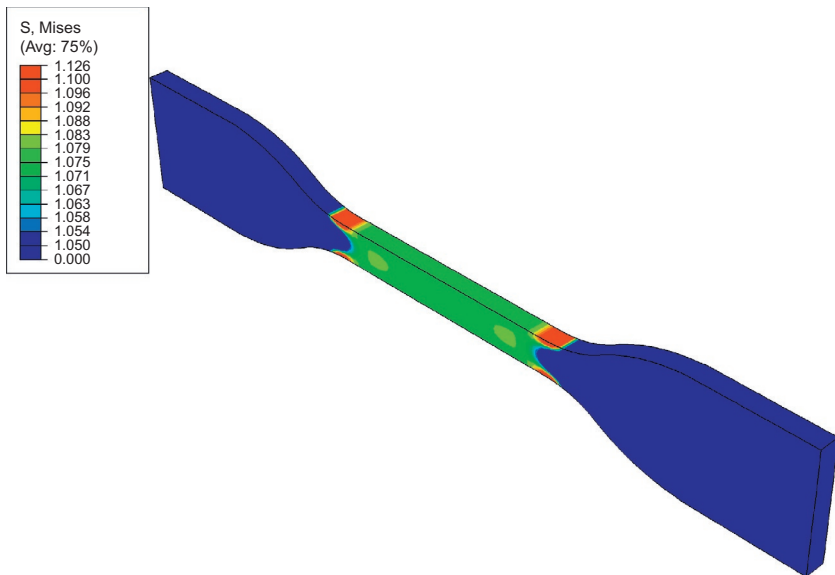
image correlation. The mathematical details of necking are given in Figure 2.12. As is shown in this figure the traditional condition for necking is the following:

$$\frac{d\sigma^{eng}}{d\varepsilon^{eng}} = 0. \tag{2.1}$$

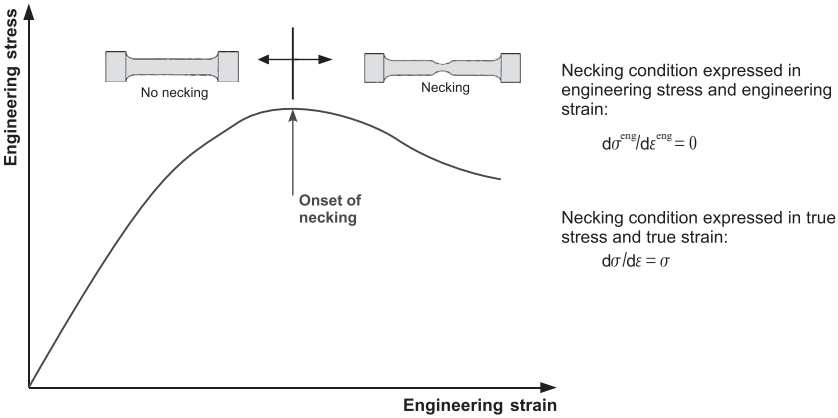
That is, if there is a peak in the engineering stress versus engineering strain curve then the specimen may neck. In practice,



**Figure 2.10** FE results showing the influence of interface friction on the predicted stress-strain response during both loading and unloading. The frictional forces influence both the tension and the compression responses.



**Figure 2.11** Stress distribution inside an ASTM D638 Type IV specimen pulled in tension to a true strain of about 15%. The material was represented using a Neo-Hookean material model with a shear modulus of 2.0 MPa and a bulk modulus of 200.0 MPa.



**Figure 2.12** Necking of test specimens in uniaxial tension.

however, a small drop in the engineering stress may not lead to a visible neck in tension specimen.

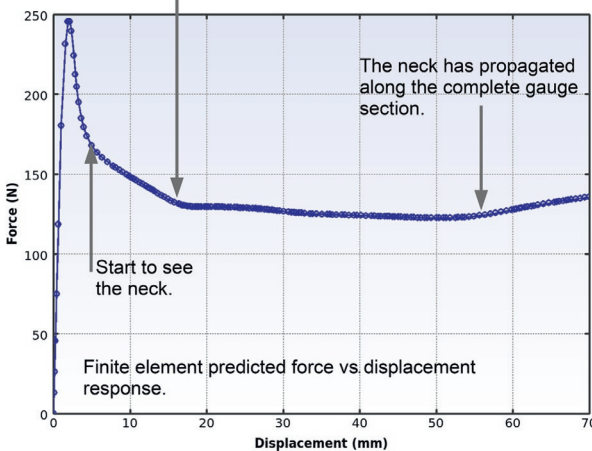
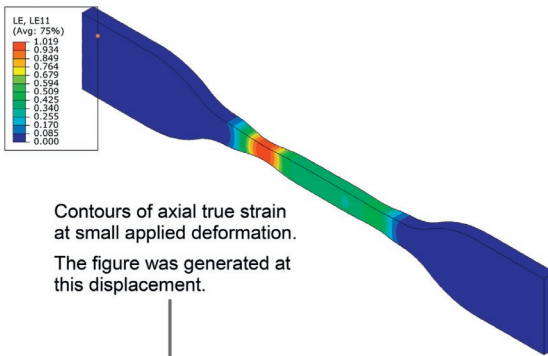
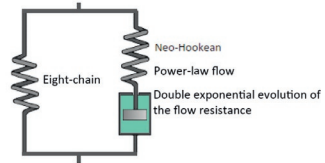
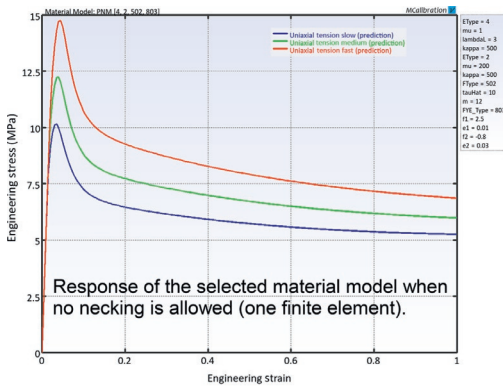
Figure 2.13 shows the progression of the necking from a virtual experiment. In this case, a material model with strong stress softening after yielding was selected using the Parallel Network Model from the PolyUMod library (see Section 8.7). The ASTM D638 Type IV specimen was then simulated with that material model. As shown in the figure, the necking starts relatively early and a stable neck is then propagated along the gauge section of the specimen until it has reached a maximum length. At that time the total force starts to increase again.

Note that only some polymer neck when loaded in tension. For example, elastomers do not neck, LDPE and HDPE undergo necking, but UHMWPE does not neck.

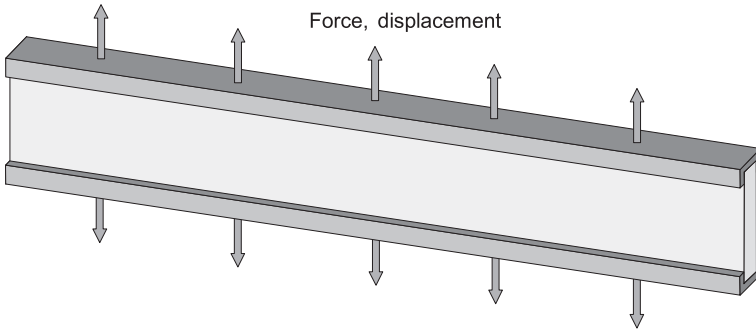
If the purpose of the experimental testing is to obtain suitable experimental data for material model calibration, then it is still possible to use tension data with necking as long as experimental setup is simulated as part of the calibration procedure. This topic is discussed in more detail in Chapter 9.

### 2.2.3 Plane Strain Tension

One relatively common loading mode for elastomers is *plane strain tension*. In this test, a wide sheet of material is pulled in



**Figure 2.13** Necking of an ASTM D638 IV test specimen pulled in uniaxial tension. The material was modeled using the specified Parallel Network Model.



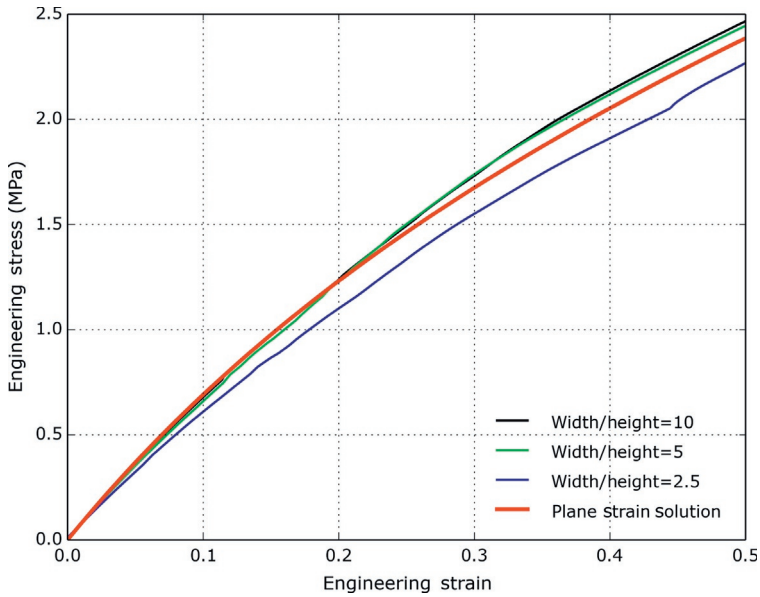
**Figure 2.14** Schematic figure showing a plane strain tension specimen.

tension, see [Figure 2.14](#). The general guideline is for the specimen width to be at least 10 times wider than its height. By having a wide specimen, the in-plane displacements will be mainly along the loading direction (and through the specimen thickness). This will ensure that the deformation state is close to plane strain.

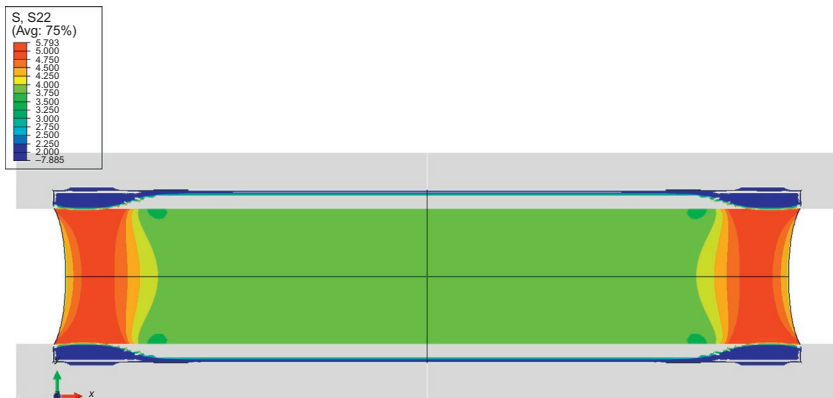
In plane strain loading, since there is no contraction of the specimen its width direction, the thickness strain will be similar in magnitude to the applied axial strain (as long as the material is almost incompressible). As an example, if the applied axial (true) strain is 50% then the strain through the thickness will be approximately  $-50\%$ . Even if the specimen is gripped with platens that apply a constant pressure, the specimen will thin down at the interface line with the platen edge causing an effective specimen length that increases with the applied strain. Due to this effect it is important to measure the effective axial strain in the center of the specimen. This can be achieved by using a laser extensometer or a digital image correlation strain measurement system.

The influence of the specimen width on how close the deformation state is to the desired plane strain deformation state is examined in [Figures 2.15](#) and [2.16](#). These figures show the results from an FE study that is using one-eighth symmetry, and a Neo-Hookean material model with a shear modulus of 2 MPa and a bulk modulus of 200 MPa. In the simulation, the specimen was gripped using an analytical rigid surface that was pushed against





**Figure 2.15** Influence of specimen width on the stress-strain response.



**Figure 2.16** Contours of vertical stress in a plane strain test specimen that is 10 times wider than its initial height. The applied engineering strain (at the crosshead) is 0.8.

the specimen with a force per grip length of 10 N/mm. The friction coefficient between the specimen and the simulated loading platen was 1.0.

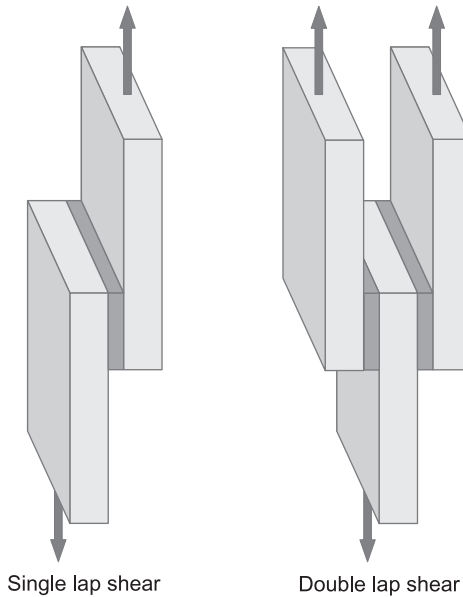
The thick red line in this figure shows the true plane strain response, the blue line shows the response if the specimen is 2.5 times wider than its initial height, the green and the black lines show the response for 5 and 10 times wider than the initial height. The FE results show that having a specimen width of only 2.5 times the initial height is not sufficient for introducing a mainly plane stress-strain state, and that if the width is 5 or 10 times the initial gauge section length then the determined stress will have an error that is less than 9% at an applied effective strain of 50%.

One other experimental complication of the plane strain tension test is that it is often necessary to grip the specimen relatively tight in order to avoid specimen slippage at the grips. This can cause a stress concentration close to the grips, and since the specimen is not dog-bone shaped, may influence the stress and strain state inside the specimen.

#### 2.2.4 Simple Shear Testing

As will be discussed in Chapter 5, most finite strain material models for polymers divide the mechanical response into distinct shear and volumetric behaviors. Based on this decomposition of the deformation it is sometimes desirable to directly measure the shear response of a material. This can be achieved, for example, by performing torsion experiments. But since torsional test machines are still relatively uncommon, it is valuable to be able to determine the shear response using a more common uniaxial loading machine. A frequently used technique to measure the shear response is to use a single-lap or double-lap shear setup, as shown in [Figure 2.17](#).

The results from single- and double-lap shear experiments can be quite different. The single-lap shear test is not self-centering and the top and bottom regions need to be tightly gripped in order to avoid specimen rotation which will result in a deformation



**Figure 2.17** Schematic figures of single-lap and double-lap shear specimens. The dark blue regions are the test specimens.

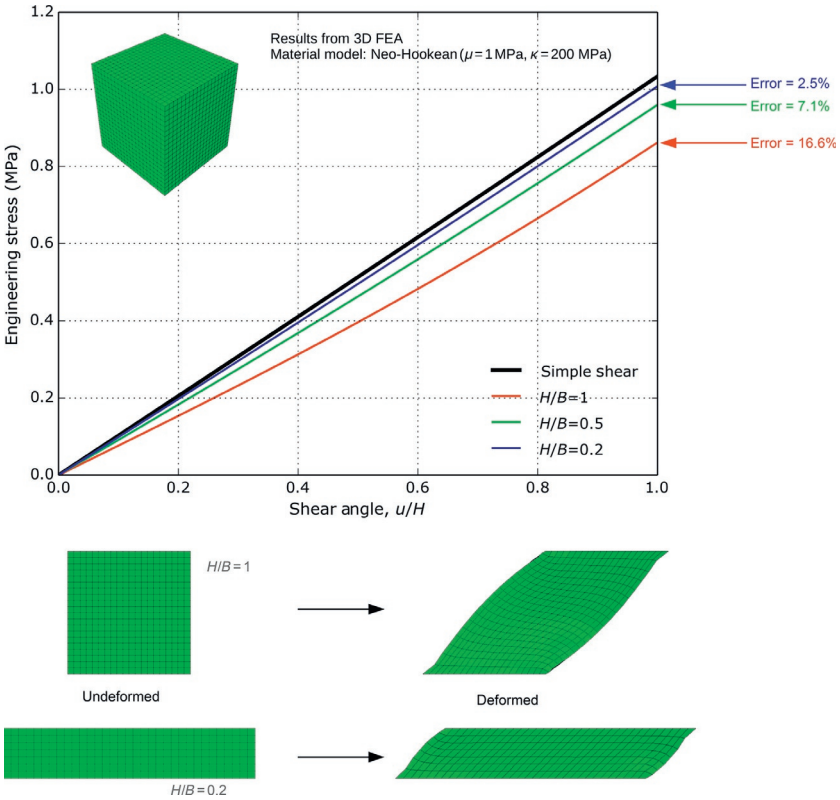
state that is not consistent with simple shear. The symmetry of the double-lap shear test removes most issues related to specimen rotations, and because of that it is usually easier to both grip and run a double-lap shear test.

To get a deformation state that is mostly simple shear, and that is defined by the deformation gradient

$$\mathbf{F} = \begin{bmatrix} 1 & \gamma & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.2)$$

where  $\gamma$  is the applied displacement divided by the specimen thickness, it is important that the thickness of the specimen is small relative to the two other in-plane dimensions.

Figure 2.18 shows the results from an FE study aimed at examining the error that can be expected in a simple lap or double lap shear experiment. In this case, the specimen is assumed to have a base area of 10 mm  $\times$  10 mm, and different



**Figure 2.18** FE results showing the influence of specimen dimensions on the simple shear stress. Neo-Hookean material model with a shear modulus of 2 MPa and a bulk modulus of 200 MPa.

specimen heights (thicknesses) were examined. The material was represented using a Neo-Hookean hyperelastic model with a shear modulus of 2 MPa and a bulk modulus of 200 MPa. The specimen was assumed to not rotate during the loading, and the specimen was assumed to be perfectly bonded to the loading platens.

The results from the study show that the accuracy is increased (and the loading mode more similar to simple shear) for specimens with a small height to base width. As an example, if the specimen width (and depth) is five times larger than the specimen height, then the error in the predicted shear stress is about 2.5%.

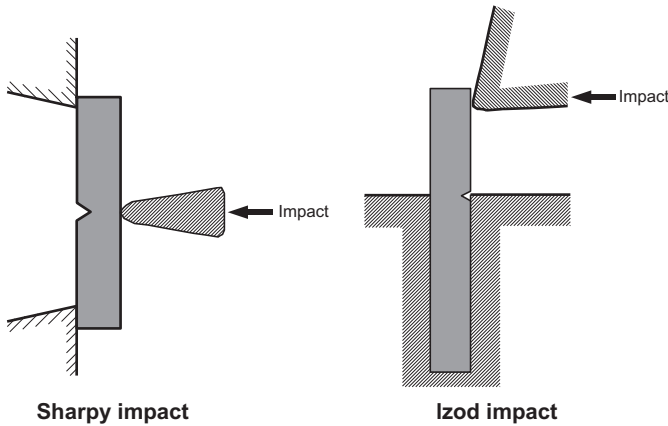
### 2.2.5 Impact Testing

In the traditional experimental stress-strain tests, discussed in Sections 2.2.1–2.2.4, the test specimens are typically loaded at a quasi-static rate. In many real applications, however, polymer components are often exposed to rapid impact loads: for example, impact after free falling, or direct blows or collisions. The purpose of performing mechanical impact tests is to simulate these conditions more directly and to examine the mechanical behavior of polymers under impact conditions.

Two commonly used impact testing methods for polymers are the *Izod* and *Charpy* tests. Both the *Izod* and the *Charpy* tests are performed using a pendulum impact machine, see Figures 2.19 and 2.20.



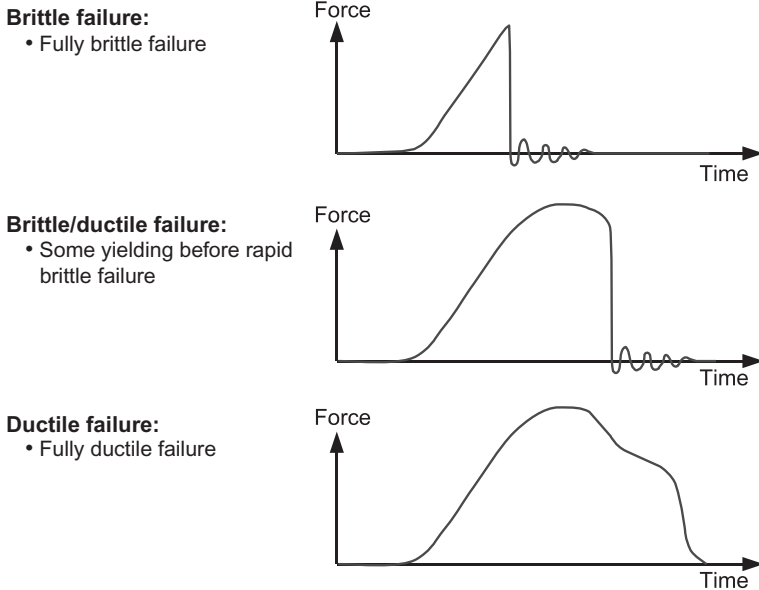
**Figure 2.19** Typical analog (noninstrumented) impact testing machine. (Reproduced with permission from Qualitest, [www.WorldofTest.com](http://www.WorldofTest.com).)



**Figure 2.20** Test specimen geometry used in Sharpy and Izod impact testing.

In the tests, the specimen is positioned in a vice, and the pendulum hammer is released from a predefined height. The residual kinetic energy in the hammer after the specimen has been broken carries the hammer upward to a final location. The energy required to break the specimen can be directly obtained from the initial drop height and the final drop height. The impact tests are routinely performed over a wide temperature interval to examine ductile-brittle transitions. The tests are also often performed with different specimen geometries, and with specimens with or without notches.

There are two basic ways in which the impact tests can be performed: noninstrumented or instrumented. A noninstrumented impact test only measures the energy required to break the specimen, whereas an instrumented impact test also measures the force history during the impact, see [Figure 2.21](#). In the instrumented impact tests, the pendulum's hammer is equipped with a load cell, and the impact data are stored using a high-speed data acquisition system enabling extraction of not only the energy required to cause failure but also force and velocity as a function of time. This measured force can be used to distinguish between brittle and ductile fracture, and can be used to extract information about the required energies for crack initiation and



**Figure 2.21** Schematic test results from an instrumented Izod or Charpy impact test.

crack propagation. Note that the most commonly used impact-test standards do not call for instrumentation [7, 8].

Due to the complicated evolution in multiaxial stress and strain during the impact event, the results from these impact tests are typically not useful for determining true material properties or for calibrating constitutive models. What the tests are useful for is to provide relative ranking of impact strength between different materials. As a further complication, the impact properties can be very sensitive to test specimen thickness and molecular orientation. The differences in specimen thickness as used in the ASTM and the ISO standard methods may affect impact properties strongly [7–10].

Both the Izod and the Charpy impact tests can present experimental challenges. One of the main weaknesses of the Izod impact test is that it tends to measure notch sensitivity instead of inherent impact toughness. To overcome this weakness is sometimes unnotched Izod tests performed. Another weakness of the Izod test is that the force used to clamp the sample can vary between

tests, and the force creates an initial stress state in the specimen. These complications often increase the statistical variations in the impact test results. A final weakness of both Izod and Charpy tests is that incorrect or inconsistent notching of test specimens can cause significant variability.

The in-plane tensile impact resistance of polymer films is tested using a different experimental approach. These impact tests are typically performed using the Gardner falling weight method for rigid materials [11] or dart drop for films [12]. In these tests, a weight is placed at the end of a nub or dart that is raised to a specific height and dropped on the secured specimen. The drop tests are typically incremental, requiring a relatively large number of test specimens to determine the failure energy. The energy absorbed during the impact is calculated based on the contact area, weight, and drop height of the impact object. Due to the strain-rate dependence of polymers, a 2 kg weight dropped from 1 m impart the same amount of energy as a 20 kg weight dropped from 0.1 m, but the two cases can result in different behaviors due to different impact velocities. These tests typically provide pass/fail data: they give the average impact energy that breaks the sample 50% of the time. If the falling dart is instrumented then these tests can also be used for material model calibration by using FE simulations of the experimental setup.

The high strain-rate response of many materials that are only available in challenging specimen geometries can also be evaluated using impact tests. As an example, instrumented ball impact testing can very accurately determine the through thickness response of thin films and pressure sensitive adhesives [13].

## 2.2.6 Dynamic Mechanical Analysis

Dynamic mechanical analysis (DMA) is a useful technique for experimental characterization of the small-strain viscoelastic properties of polymers [14–17]. DMA measures the stiffness and viscoelastic damping properties under dynamic vibrational loading at different temperatures. The technique is applicable to virtually all polymers, including elastomers, thermoplastics,

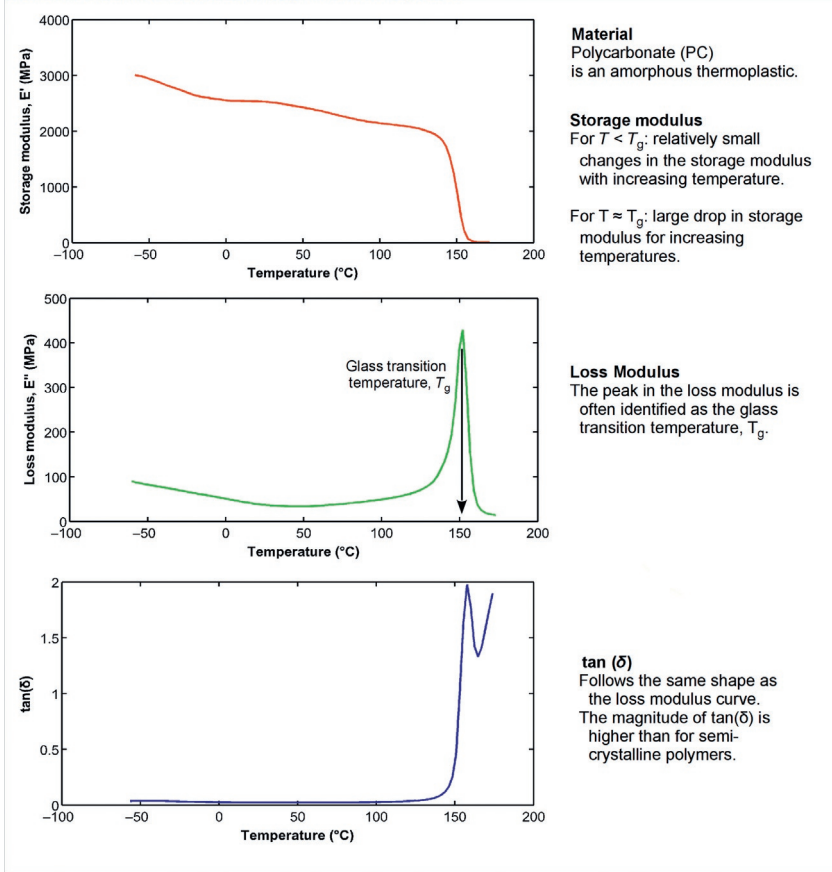


thermosets, and films and fibers of these materials. DMA is an interesting technique because of its ease of use and its ability to extract large amounts of experimental data from a few experimental tests. It is also one of the most sensitive analysis techniques for determining, for example, the glass transition temperature  $T_g$ .

In DMA experiments, a specialized testing machine applies a sinusoidal force or displacement to a test specimen and the resulting response is measured. For viscoelastic materials the measured response lags behind the input stress wave with respect to its phase angle and this lag is known as the phase angle,  $\delta$ . As is discussed in more detail in Chapter 6, an effective modulus, called the complex modulus,  $E^*$  can be obtained by dividing the stress amplitude with the strain amplitude. The complex modulus can be further decomposed into a storage modulus  $E'$ , which is in-phase with the applied loading, and a loss modulus  $E''$ , which is out of phase with the applied loading. The storage modulus is directly related to the energy storage capabilities of the material, and the loss modulus is related to the dissipated heat (hysteresis). Another commonly used quantity is  $\tan(\delta)$ , which is the ratio of the loss modulus to the storage modulus, that is  $\tan(\delta) = E''/E'$ . The quantities  $E'$ ,  $E''$ , and  $\tan(\delta)$  are strong functions of temperature and loading frequency. The goal of DMA experiments is to determine how these viscoelastic quantities depend on temperature and loading frequency.

The experimental procedure to perform DMA is described both in ASTM [16] and ISO standards [17]. The DMA experiments can be performed in many different loading modes, for example, uniaxial tension, compression, shear, bending, or torsion. The experimental setup is driving the deformation either at resonance frequency or using forced vibration. Which loading mode and approach that is used in the experiment is determined by the particular DMA test machine that is used. For solid polymers the most common experimental approach is to select a constant frequency and amplitude of the vibration, and then measure the material response as a function of temperature. Following standard guidelines [16, 17], the temperature is swept either in a stepwise manner with constant temperature increments that are held for a

Characteristic DMA Results for Amorphous Thermoplastics

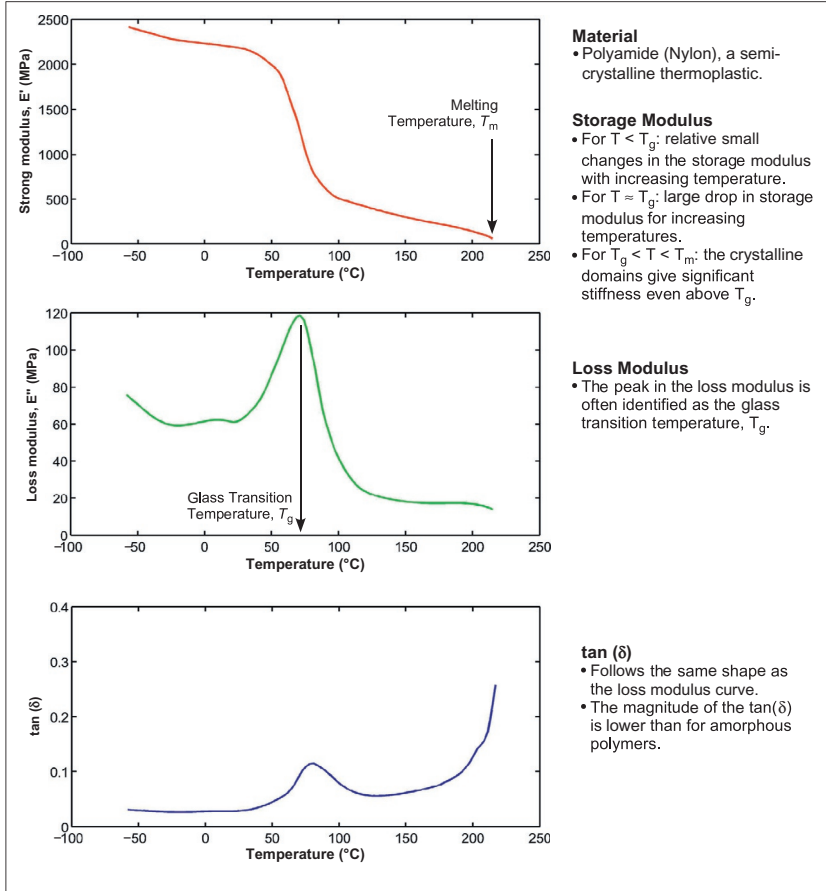


**Figure 2.22** Exemplar DMA test results for polycarbonate which is an amorphous thermoplastic polymer.

certain time before the dynamic measurements are made, or by applying a constant heating rate, often about 2 °C/min.

Exemplar DMA result for amorphous and semi-crystalline thermoplastics are shown in Figures 2.22 and 2.23. Figure 2.22 shows the storage modulus ( $E'$ ), loss modulus ( $E''$ ), and  $\tan(\delta)$  for polycarbonate, an amorphous thermoplastic. The figure shows that both the storage modulus and the loss modulus are only weakly dependent on temperature for temperatures  $T < T_g - 30\text{ }^\circ\text{C}$ . In the temperature range from  $T_g - 30\text{ }^\circ\text{C}$  to  $T_g + 30\text{ }^\circ\text{C}$ , the storage modulus goes from an initially high value down to

## Characteristic DMA Results for Semi-Crystalline Thermoplastics



**Figure 2.23** Exemplar DMA test results for nylon which is a semi-crystalline thermoplastic polymer.

virtually zero stiffness. In the same temperature interval, the loss modulus exhibits a peak due to the enhanced mobility of the amorphous microstructure from thermal energy. The temperature at which the loss modulus reaches its peak value is often taken as the glass transition temperature,  $T_g$ . Since  $\tan(\delta)$  value is given by  $E''/E'$ , the overall shape of  $\tan(\delta)$  is similar to the shape of  $E''$ .

Figure 2.23 shows the DMA data for polyamide (Nylon), a semi-crystalline thermoplastic. The storage modulus is at low

temperatures only weakly dependent on the temperature, but undergoes a drastic reduction around  $T_g$ . Due to the crystalline domains of the microstructure, the storage modulus stays finite at temperatures that are above  $T_g$  but below the melting temperature of the crystalline domains. For this reason, semi-crystalline polymers can often be used as structural components even at as high temperatures as  $T_g + 100^\circ\text{C}$ . The loss modulus and the  $\tan(\delta)$  function follow a similar trend as for amorphous thermoplastics. Due to the integrity of the crystalline domains,  $\tan(\delta)$  is typically much smaller for semi-crystalline thermoplastics compared with amorphous thermoplastics.

### 2.2.7 Hardness and Indentation Testing

The hardness of a polymer is an empirical quantity related to the inherent indentation resistance. The two most common methods for measuring polymer hardness are the Rockwell hardness test [18] and the Shore (durometer) hardness test [19].

These tests provide a fast and easy method to characterize the hardness of a polymer. They are useful for ranking the hardness of different polymers, but do not necessarily provide results that correlate with other fundamental material properties.

#### ***Rockwell Hardness Testing***

The Rockwell hardness test is typically chosen for stiff thermoplastics, such as polycarbonate and polystyrene. For these materials the resistance to viscoelastic flow below the glass transition temperature is almost time-independent and the extracted hardness values are therefore also almost time-independent.

Rockwell hardness tests are typically performed following ASTM D785 [18]. In this test specification, a specimen with a thickness of at least 6 mm is indented by a steel ball. A small load is applied on the indenter, the indentation displacement is recorded, and then a larger load is applied and removed. After a

short time, with the preload still applied, the remaining indentation is determined and used to calculate the hardness value.

Different Rockwell hardness scales utilize different-size steel balls (between 3 mm and 13 mm) and different loads (between 60 kg and 150 kg). The three most common scales used for plastics are Rockwell E, Rockwell M, and Rockwell R. The correlation between the Rockwell scales is weak; attempts at conversion between the scales are therefore discouraged.

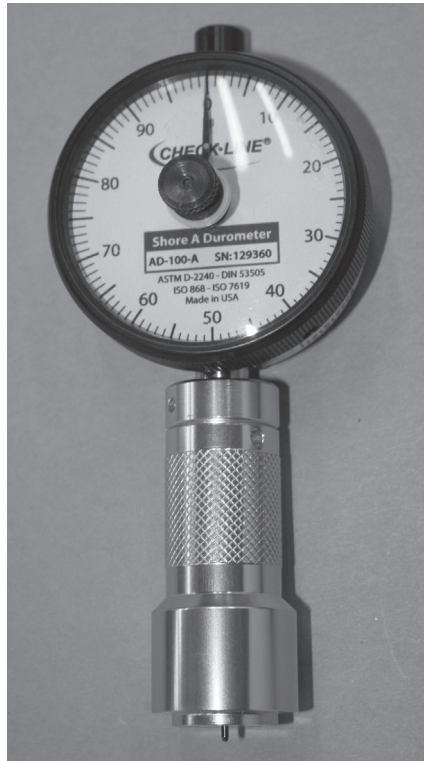
The results obtained from this test provide a useful measure of relative resistance to indentation of various types of plastics. However, the Rockwell hardness test does not serve well as a predictor of other properties such as strength or resistance to scratches, abrasion, or wear, and should not be used alone for product design specifications.

### **Shore (Durometer) Testing**

The Shore (Durometer) test is the preferred hardness testing method for elastomers and is also commonly used for compliant plastics such as polyolefins, fluoropolymers, and vinyl polymers. Shore hardness testing is typically performed following ASTM [19] standards. The two most common test procedures are the Shore A and the Shore D scales. The Shore A scale is used for “softer” rubbers while the Shore D scale is used for “harder” ones. Other Shore scales, such as Shore O and Shore H hardness, are used less often.

The Shore hardness is measured with an apparatus known as a Durometer (see [Figure 2.24](#)) and the determined hardness values are therefore referred to as *Durometer hardness*. The hardness value is determined by the indentation distance into the sample. The type of indenter and applied load is determined by the durometer hardness scale. Due to the viscoelastic behavior of rubbers and plastics, the indentation reading may change over time—so the indentation time is sometimes reported along with the hardness number.

The results obtained from these tests provide a useful measure of relative resistance to indentation of various grades of polymers.



**Figure 2.24** Photograph of an exemplar analog durometer. (Reproduced with permission from Veryst Engineering, [www.veryst.com](http://www.veryst.com).)

As for the Rockwell hardness, the Shore Durometer hardness value does not provide direct information of other properties such as strength or resistance to scratches, abrasion, or wear, and should not be used alone for product design specifications.

The correlation between the different Shore Durometer hardness scales is weak, and attempts at converting between the scales are discouraged. Similarly, conversion between Shore hardness and Rockwell hardness is discouraged. Despite these limitations it can be useful to show a rough indication of how the different hardness scale are correlated. [Figure 2.25](#) shows a schematic of the different hardness scales.

Polymer	Hardness scales		
	Durometer A	Durometer D	Rockwell R
Phenolics →			—150— —140—
Acrylics →			—130—
Polycarbonate →			—120— —110—
Nylon →		— 80 —	—100—
Polystyrene →		— 70 —	— 90 —
Polypropylene →		— 60 —	— 80 —
	— 90 —	— 50 —	— 70 —
	— 80 —	— 40 —	— 60 —
Filled rubber →	— 70 —	— 30 —	
	— 60 —		
	— 50 —		
	— 40 —		
Unfilled rubber →	— 30 —		
	— 20 —		

**Figure 2.25** Hardness scales used for different classes of polymers. Durometer A and D are primarily used for elastomers and soft thermoplastics. The Rockwell hardness test is often used for stiff thermoplastics.

### **Barcol Hardness Testing**

Barcol hardness is a third hardness test that is used to determine the hardness of both reinforced and nonreinforced thermoset polymers using a barcol impressor. The test procedure is described in ASTM D2583 [20]. The test procedure is portable and therefore suitable for both field testing and quality control testing. The indenter is a truncated steel cone that is loaded onto the test specimen.

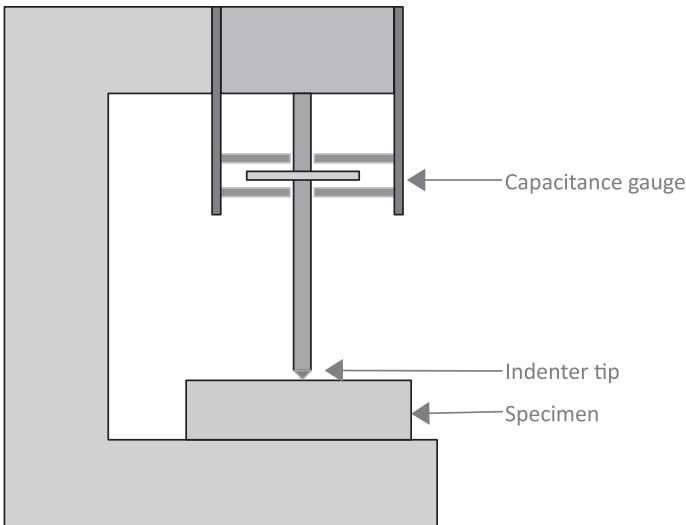
As for the other hardness tests, the barcol hardness test does not provide direct information of other mechanical properties, and should only be used as a relative ranking of hardness.

### ***Nanoindentation***

Instrumented microindentation and nanoindentation can be quite useful for evaluating the nonlinear viscoplastic response of many different polymers. It is particularly suitable for cases where it would be difficult to extract large enough or otherwise suitable test specimens for the more traditional testing techniques.

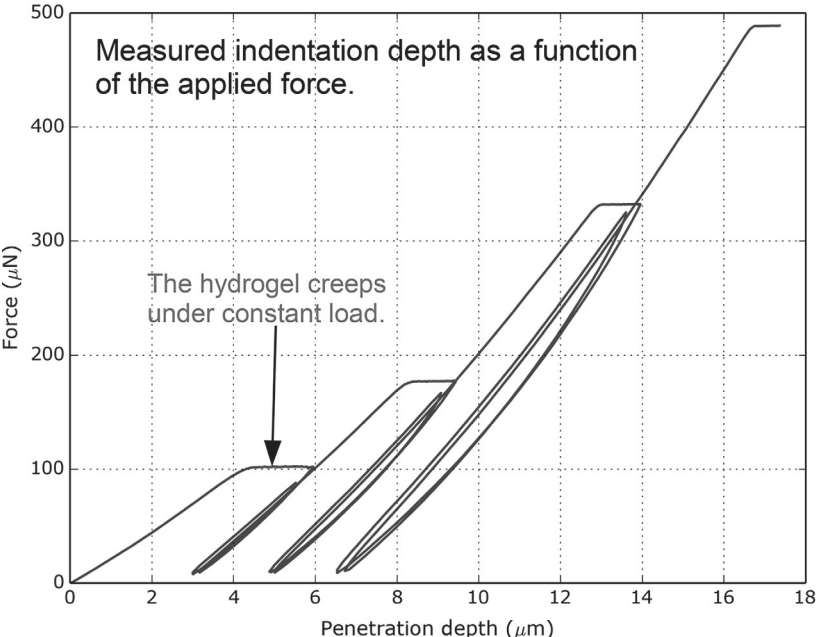
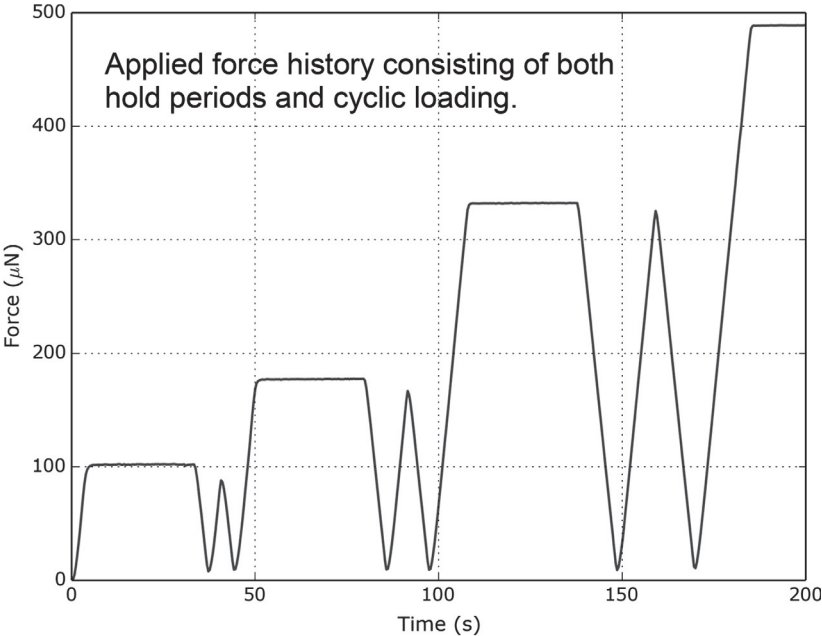
Typically a nanoindentation test system is operated by applying a known force on the indenter rod, which is then loading the test specimens through a specified indenter tip. The applied displacement is determined from an accurate capacitance gauge, see [Figure 2.26](#).

One example of the results that can be obtained from a nanoindentation test is shown in [Figure 2.27](#). In this case, a hydrogel material was tested using a nanoindenter run in load-control mode. The specimen was loaded using multiple load-unload segments with inserted creep segments. The results from this test illustrate that the hydrogel undergoes significant creep under the tested conditions. A suitable material model can be calibrated to this data set by inverse calibration using FE simulations of the specific loading geometry and history.



**Figure 2.26** Schematic of a nanoindentation setup.





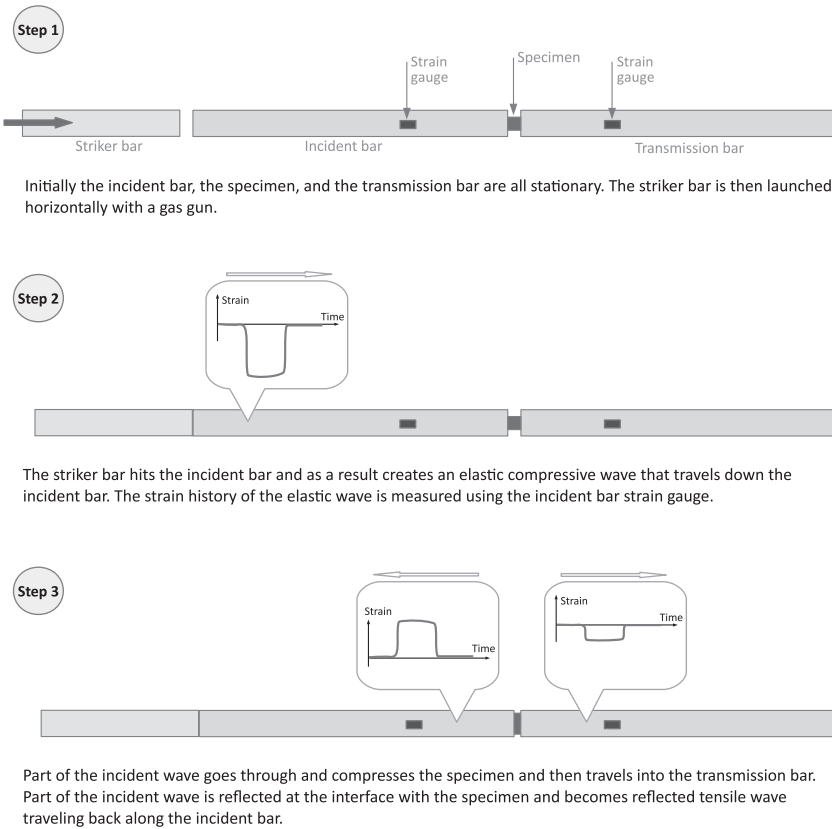
**Figure 2.27** Exemplar nanoindentation testing of a hydrogel material. The material exhibits a nonlinear viscoelastic response.

### 2.2.8 Split-Hopkinson Pressure Bar Testing

Measuring the high strain-rate response of polymers can be very important for many industrial applications where a component is rapidly loaded, for example in drop tests or impact situations. The mechanical response of different adhesives, elastomers, and thermoplastics can be strongly dependent on the loading rate. For this reason, it is important to be able to quantify the material response also at very high strain rates. The most common way to achieve this is to use a *Split-Hopkinson Pressure Bar* (SHPB) test system, which is also called a *Kolsky bar*. The SHPB approach can be applied in different loading modes, including tension, compression, torsion, etc. This section focuses on an SHPB system operating in a compressive mode, additional information for other loading modes is given in the book by Chen and Song [21].

A traditional compressive SHPB system consists of a striker bar, an incident bar, a test specimen, and a transmission bar (see [Figure 2.28](#)). The striker bar, the incident bar, and the transmission bar are typically made from aluminum or steel, but sometimes a polymer (e.g., poly(methyl methacrylate) or polycarbonate) is also used. At the start of an SHPB test, all components are stationary and the specimen is positioned between the incident bar and the transmission bar. Then the striker bar is launched horizontally from a gas gun. The striker bar hits the incident bar and as a result creates an elastic compressive stress wave that travels down the incident bar. Since the incident bar and the transmission bars are designed to be long and slender the stress waves will be essentially one-dimensional. The time history of the stress wave in the incident bar is measured using a strain gauge located a known distance from the specimen. Once the stress wave in the incident bar reaches the specimen interface, part of the wave propagates into the specimen and part of the wave is reflected as a tensile stress wave back in the incident bar. The strength of the reflective wave is then measured again using the same incident bar strain gauge.

The part of the stress wave that goes through the test specimen will create a compressive deformation field inside the specimen.

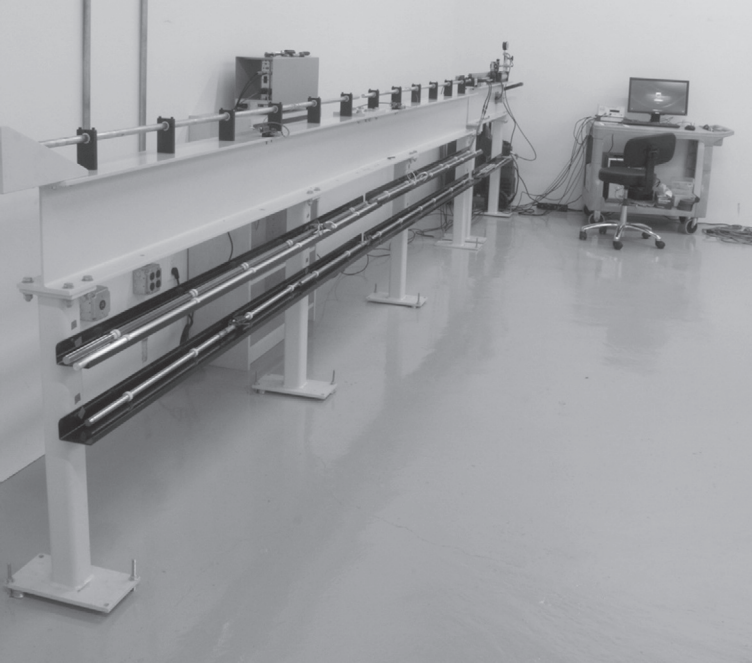


**Figure 2.28** Schematic picture of a compressive SHPB system.

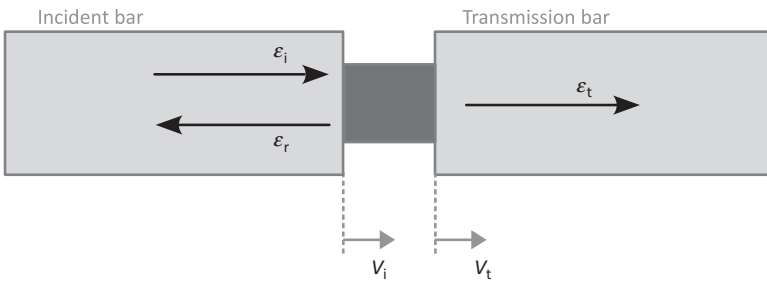
That stress wave will then propagate through the transmission bar. The strength of the transmitted stress wave is measured by a second strain gauge located at a known location along the transmission bar.

A schematic picture of an SHPB system is shown in [Figure 2.28](#), and a photograph of an exemplar system is shown in [Figure 2.29](#).

The elastic wave speed in the bars is given by:  $C_b = \sqrt{E_b/\rho_b}$ , where  $E_b$  is the Young's modulus of the bar and  $\rho_b$  is the density of the bar. As an example, if aluminum bars are used, then the Young's modulus is given by:  $E_b = 72.4 \text{ MPa}$ , the density is  $\rho_b = 2700 \text{ kg/m}^3$ , giving a wave speed of  $C_b = 5178 \text{ m/s}$ . So



**Figure 2.29** Photograph of an SHPB system. (Reproduced with permission from Veryst Engineering, [www.veryst.com](http://www.veryst.com).)



**Figure 2.30** Definition of incident, transmitted, and reflected strains at the specimen interfaces in an SHPB test.

if the incident bar is  $L_s = 2$  m long, then it will take  $t = L_s/C_b = 0.4$  ms for the wave to reach the specimen.

The stress and strain in the test specimen can be calculated from the signals from the incident bar strain gauge and the transmitted bar strain gauge, see [Figure 2.30](#).

The effective engineering strain rate in the specimen is given by:

$$\dot{\varepsilon}_s = \frac{v_i - v_t}{L_s} = \frac{C_b}{L_s} (\varepsilon_i - \varepsilon_r - \varepsilon_t), \quad (2.3)$$

where  $v_i$  is the speed of the incident side of the specimen,  $v_t$  is the speed of the transmission bar side of the specimen, and  $L_s$  is the specimen length. The engineering stress on the incident side of the specimen is:

$$\sigma_i = \frac{A_b}{A_s} E_b (\varepsilon_i + \varepsilon_r), \quad (2.4)$$

where  $A_b$  is the cross-sectional area of the bar and  $A_s$  is the cross-sectional area of the specimen. The engineering stress on the transmitted side of the specimen is:

$$\sigma_t = \frac{A_b}{A_s} E_b \varepsilon_t. \quad (2.5)$$

If the specimen is in stress equilibrium then  $\sigma_i = \sigma_t$ , which gives  $\varepsilon_i + \varepsilon_r = \varepsilon_t$ . Since the three strains  $[\varepsilon_i, \varepsilon_r, \varepsilon_t]$  are measured, the experimental strain values can be used to assess if the specimen was in stress equilibrium during the experiment.

Based on these signals, the engineering stress history in the specimen can be calculated from:

$$\sigma_s(t) = \frac{A_b}{A_s} E_b \varepsilon_t(t). \quad (2.6)$$

The engineering strain rate in the specimen is given by Equation (2.3) together with the stress equilibrium condition:

$$\dot{\varepsilon}_s = -\frac{2C_b}{L_s} \varepsilon_r(t). \quad (2.7)$$

The applied engineering strain history in the specimen can be obtained by integrating the strain rate in Equation (2.7):

$$\varepsilon_s(t) = -\frac{2C_b}{L_s} \int_0^t \varepsilon_r(t) dt. \quad (2.8)$$

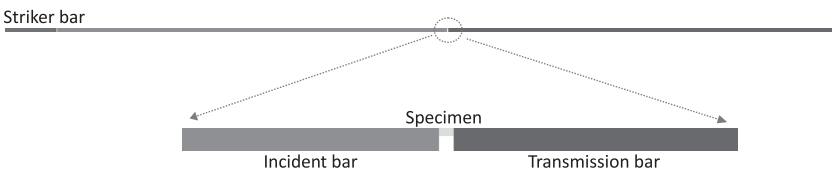
The details of how an SHPB experiment works can be demonstrated using an FE-based virtual experiment. In this virtual

experiment, the striker bar has a length of 0.3 m and a diameter of 18 mm, the incident bar has a length of 2.3 m and a diameter of 18 mm, and the transmission bar has a length of 2.3 m and a diameter of 18 mm. All of these bars are made of aluminum. The specimen has a length of 6.4 mm and a diameter of 6.4 mm. The initial striker bar velocity is 12 m/s.

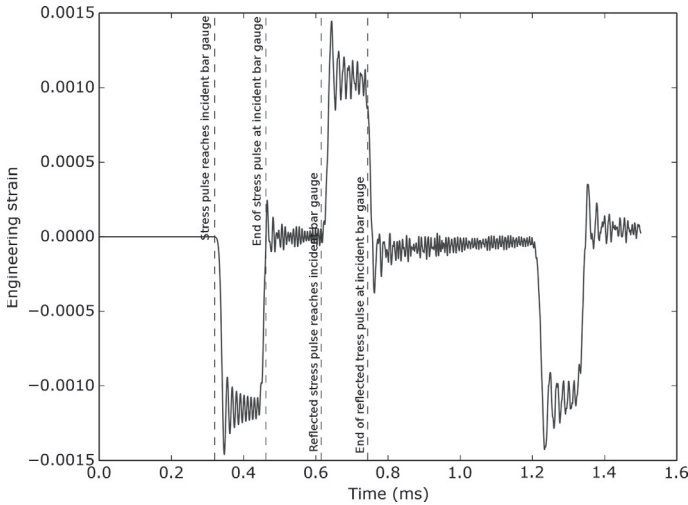
In the virtual experiment, the specimen is represented using the Three Network (TN) model (see Section 8.6) with material parameters suitable for UHMWPE. An FE model of this system was created and the stress and strain values at different locations in the model was extracted. The FE results from the virtual strain gauge locations were then inserted into the equations derived above in order to calculate the stress and strain response, just like what is done for real experimental data.

The value of using an FE-based virtual experiment is that it can be used to probe the response of the test system components at any time, and the extracted stress-strain results can also be directly compared to the known behavior of the simulated material (Figure 2.31).

Figure 2.32 shows the measured strain from the FE simulation at the incident bar gauge location. The strain at the incident bar strain gauge is initially zero, and stays zero until the stress wave has traveled from the striker bar impact location to the gauge location. At that time there is a large negative stress at the gauge location until the whole stress wave pulse has propagated to the right. The stress wave then reaches the end of the incident bar where some of the stress wave is propagated into the test specimen, and some of the stress wave is reflected as a tension wave back along the incident bar. This reflected tension wave then reaches the incident bar again.



**Figure 2.31** Schematic figure of the FE model used in the virtual experiment.



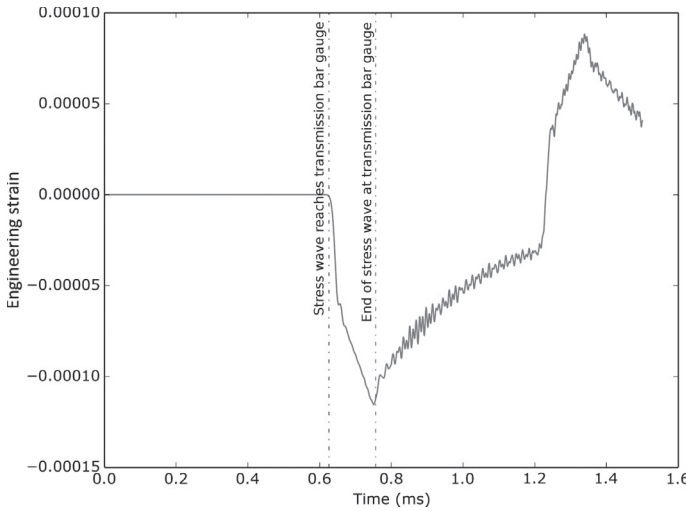
**Figure 2.32** Axial strain at the strain gauge location in the incident bar. The strain history is predicted from the FE virtual experiment.

In this example, at a time of 1.2 ms the stress wave in the incident bar has propagated back and forth in the bar and has reached the incident bar strain gauge a second time. As discussed above, this second stress wave is not used in the data analysis.

The FE predicted strain at the transmission bar strain gauge as a function of time is plotted in Figure 2.33. The transmission bar strain remains zero until slightly more than 0.6 s, at which time a compressive stress wave propagates through the strain gauge location. The time when this occurs is very similar to the time at which the reflected stress wave reaches the incident bar strain gauge (as shown in Figure 2.32).

The wave speed in the polymer specimen is lower than the wave speed in the aluminum bars, but since the specimen length is so small, the time it takes for the stress wave to propagate through the specimen is significantly smaller than the time it takes for the stress wave to propagate from the strain gauge to the end of the bar.

The engineering strain rate in the test specimen (in an average sense) can be calculated from Equation (2.7). This equation shows that the engineering strain rate is directly given by the reflected



**Figure 2.33** Axial strain at the strain gauge location in the transmission bar. The strain history is predicted from the FE virtual experiment.

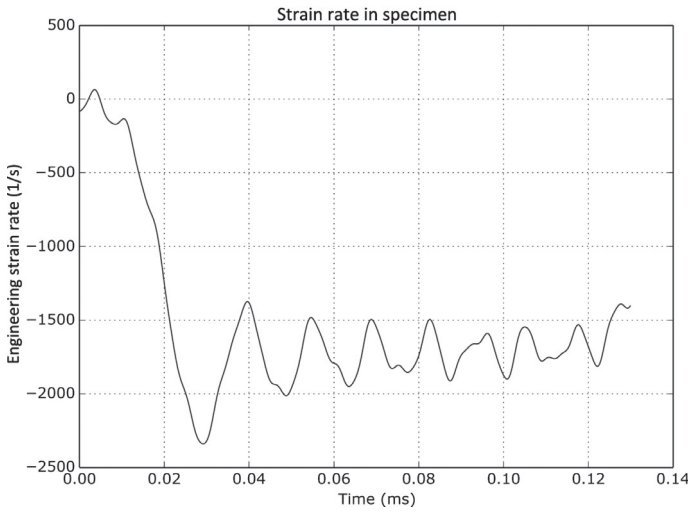
strain at the incident bar strain gauge, the wave speed in the incident bar, and the test specimen length.

In this case, the engineering strain rate goes very quickly from 0 to about  $-2000$  per second, and then stays at about  $-1800$  per second for the duration of the first stress wave. Note that the strain rate is not constant during the compression of the specimen. This is not a problem since the experimental data from the experiment can be used for material model calibration even if the strain rate is not constant, as long as the time-history of the strain rate is known (Figure 2.34).

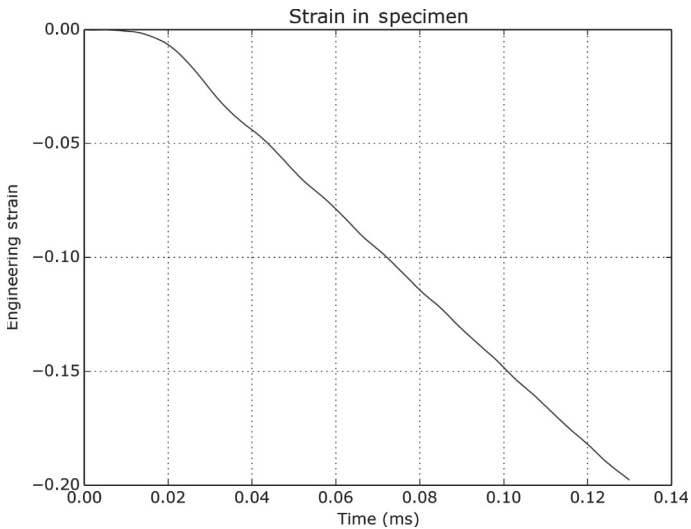
As shown in Equation (2.8), the strain history can be obtained from integrating the strain rate history from the previous figure. This integration can be done quickly using a numerical approach and the results are plotted in Figure 2.35.

The stress in the test specimen can be calculated from Equation (2.6). As shown in this equation, the specimens stress can be obtained from the transmission bar strain gauge data, as shown in Figure 2.36. This figure also shows the strain history of one element in the middle of the test specimen. The stress in this interior element was extracted from the FE postprocessing, and

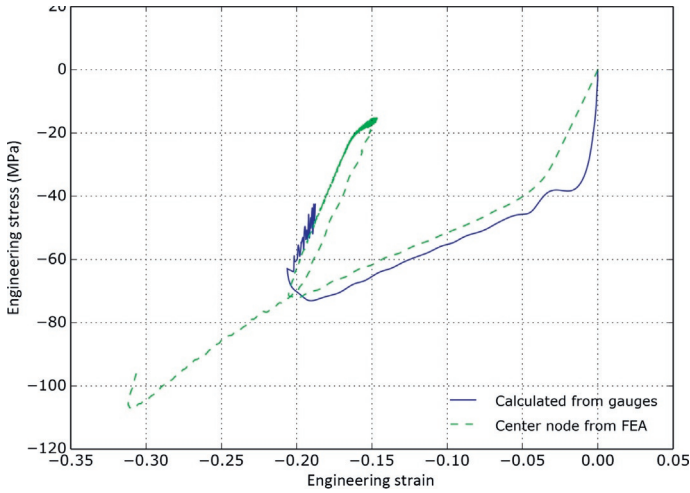




**Figure 2.34** Strain rate in the test specimen as a function of time. The strain rate is calculated from the FE virtual experiment data.



**Figure 2.35** Strain history in the test specimen as a function of time. The strain is calculated from the FE virtual experiment data.

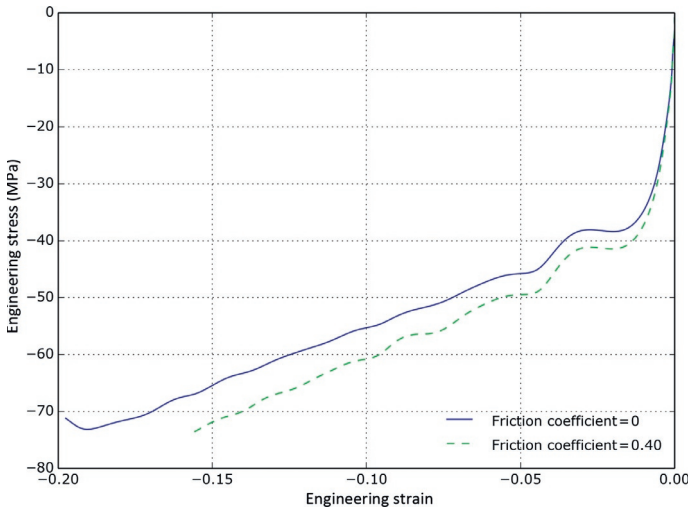


**Figure 2.36** Predicted stress in the test specimen in the FE virtual experiment.

since it was not calculated in an approximate way from the strain gauge signals but directly from the selected constitutive model, it represents the true response of the material.

It is interesting to note that the stress-strain response of the interior element undergoes loading, followed by unloading, followed by further loading. This is caused by the displacement field history in the incident bar due to the propagating stress waves. The calculated stress from the strains at the two gauge locations capture the first compression phase and a small amount of the initial unloading. This is typical of SHPB data that is obtained from an experiment on a polymer. The figure also shows that the SHPB technique does not accurately measure the initial stress-strain response at small deformations. The stress-strain response at larger deformations can often be obtained accurately.

The SHPB experiment suffers from the same issue with interface friction as in uniaxial compression. [Figure 2.37](#) shows the stress-strain response when there is no friction and when there is a friction coefficient of 0.4. The only difference between the two curves shown in the figure is the friction coefficient in the interface between the test specimen and the bars.



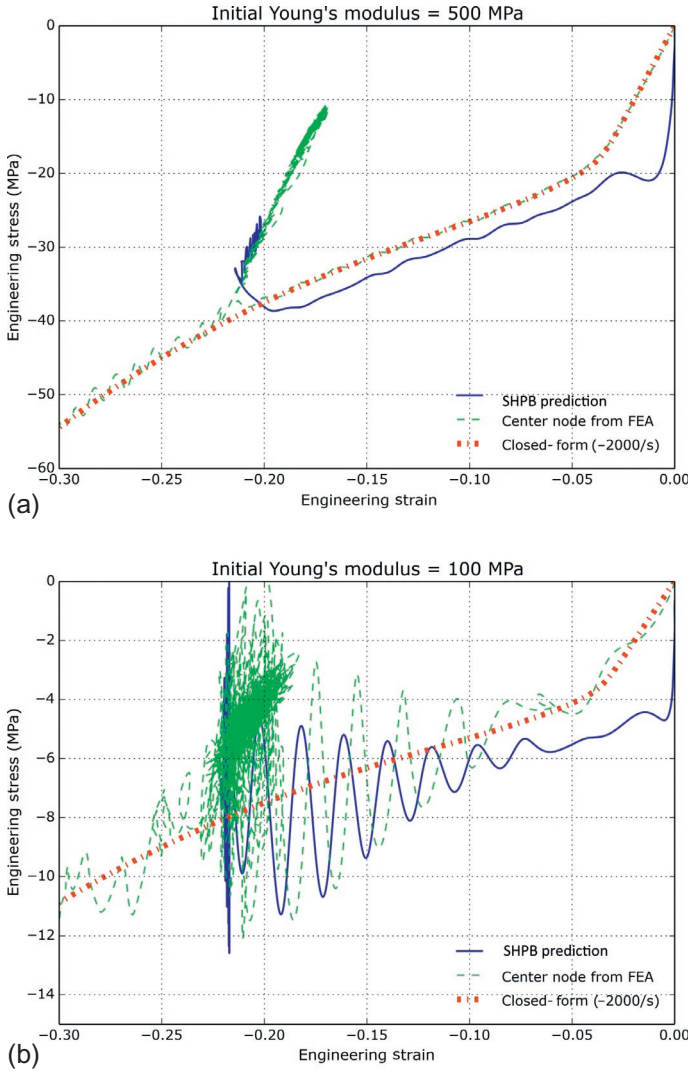
**Figure 2.37** Dependence of the predicted stress on the friction coefficient in an FE virtual experiment of an SHPB test.

As expected, the influence of friction is also important for the SHPB test, and the magnitude of how much the stress changes with friction coefficient is similar to what is observed in slow rate uniaxial compression experiments.

One of the limitations of the SHPB test is that it cannot be used to measure the high strain rate response of very soft polymers. If a material is very soft then the wave speed in the material becomes quite low (the wave speed is given by  $\sqrt{E/\rho}$ ) and the specimen will be deformed in an inhomogeneous way during the loading. Also, and equally important, the stress wave that is transmitted from the specimen to the transmission bar strain gauge becomes very low making the calculation of the stress difficult.

One example of this phenomenon is shown in [Figure 2.38](#). The graph in part (a) shows the SHPB prediction, the stress at a center node, and the exact stress-strain response from a TN material model with an initial Young's modulus of 500 MPa.

[Figure 2.38\(b\)](#) shows the same results as in [Figure 2.38\(a\)](#), but for a material with an initial Young's modulus of 100 MPa. This material model was created from the 500 MPa material model by scaling all material parameters with units of stress by a factor of one-fifth. These figures show that the SHPB prediction



**Figure 2.38** Predicted stress in the test specimen in the FE virtual experiment: (a) stress-strain response for a material with an initial Young’s modulus of 500 MPa and (b) same results for a material with an initial Young’s modulus of 100 MPa.

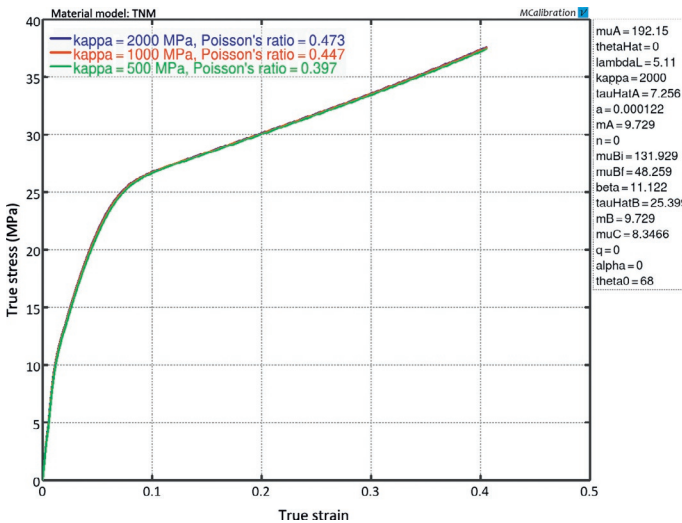
of the 500 MPa material is accurate and reliable, but that the prediction of the 100 MPa material is not useful. In this case, the 100 MPa material is too soft for accurate characterization using an SHPB test.

## 2.2.9 Bulk Modulus Testing

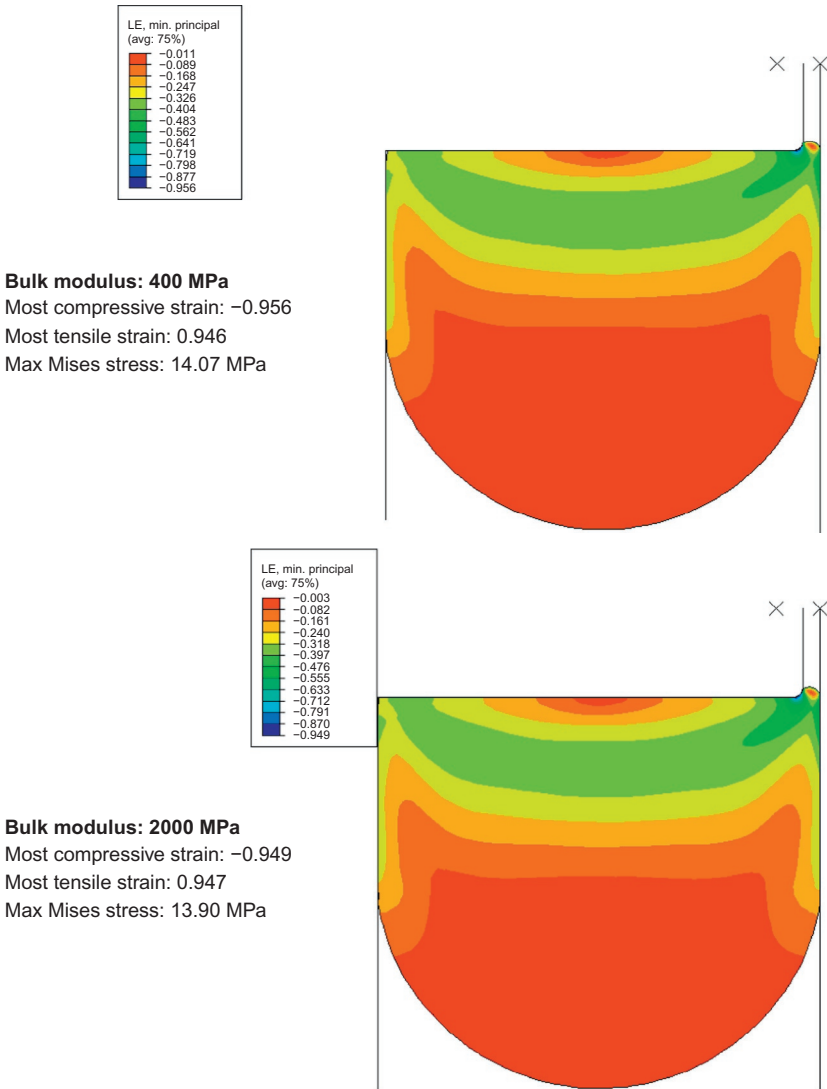
It is not necessary to accurately know the bulk modulus (or the Poisson's ratio) of a polymeric material in order to perform accurate FE simulations of most industrial applications. The reason for this is that the bulk modulus has only very limited influence on the mechanical response under most loading modes.

As an example, [Figure 2.39](#) shows the predicted stress-strain response in uniaxial tension for the TN model (see Chapter 8, Section 8.6) with different values for the bulk modulus. In this case, the true stress is reduced by less than 0.5% when the bulk modulus is reduced from 2000 MPa to 500 MPa, corresponding to a change in the small strain Poisson's ratio from 0.47 to 0.40.

In some specific applications, the bulk modulus can influence the component response. One example is an o-ring that is constrained in a groove. [Figure 2.40](#) shows the results from an example in which an o-ring is pressurized in a confined geometry with a small groove. The pressure below the o-ring was 15 MPa. The o-ring material was represented using the BB model



**Figure 2.39** Influence of the bulk modulus and initial Poisson's ratio on the predicted stress-strain response in uniaxial tension. The predictions were obtained from the TN model.



**Figure 2.40** Influence of the bulk modulus (and Poisson’s ratio) on the predicted response of an o-ring in a confined geometry with a small groove. The predictions were obtained from the BB model.

(see Section 8.2) with material parameters corresponding to a soft rubber. The FE simulation was first run with a bulk modulus of 400 MPa (corresponding to a Poisson’s ratio of 0.495), and then with a bulk modulus of 2000 MPa (corresponding to a Poisson’s ratio of 0.499).

The results from the FE simulations are shown in [Figure 2.40](#). This figure shows that the o-ring gets severely deformed due to the applied pressure and starts to extrude into the small gap. Note the difference in the predicted max stress and max strain values between the two cases with different bulk moduli is only about 1%.

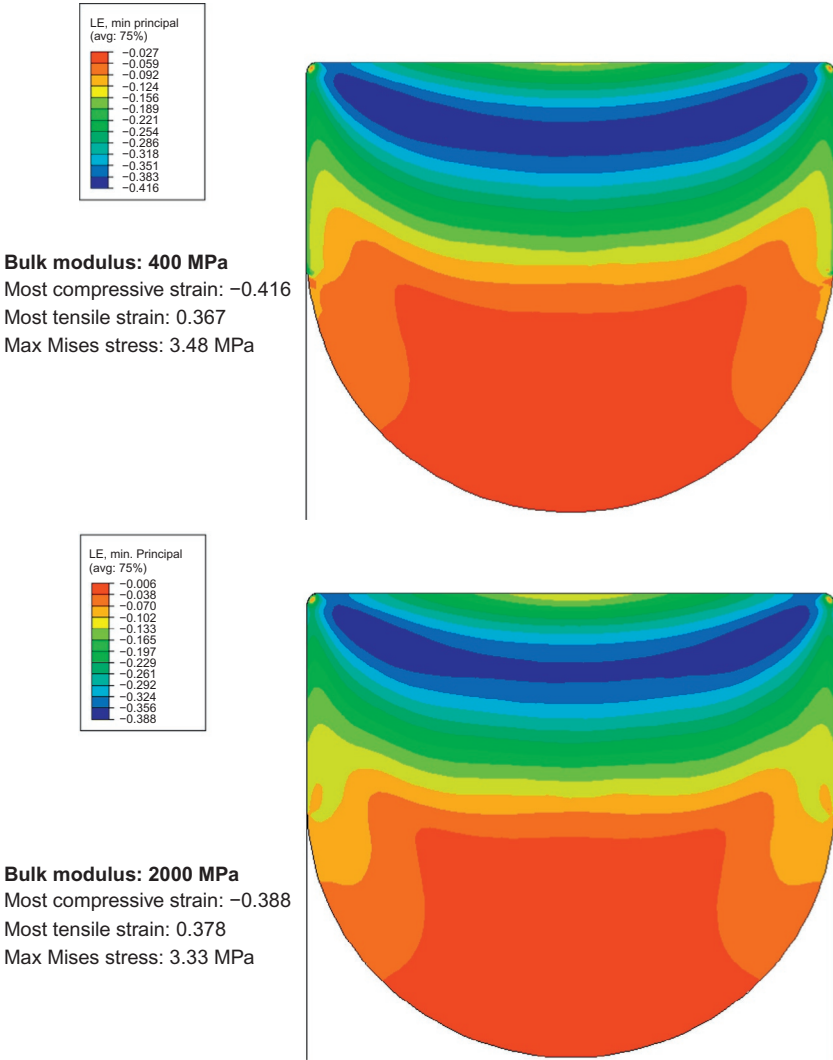
A second example with a constrained o-ring without a groove is shown in [Figure 2.41](#). This example is exactly the same as the previous example except that there is no gap for the o-ring to extrude into, and the applied pressure was 40 MPa. In this case, the predicted max stress and max strain values between the two cases with different bulk moduli is about 7%.

The bulk modulus of a polymer can be measured using different experimental techniques. One approach is to use a Digital Image Correlation (DIC) strain measurement system that can directly measure both the axial and the transverse strains during a uniaxial tension or compression test. These data can then be used to calculate the Poisson's ratio, and from that the bulk modulus can be calculated using Table 5.1. (see Chapter 5).

Another approach is to use pressure-volume-temperature (PVT) testing [22]. In this technique, the specific volume of a material is measured as a function of the applied pressure and temperature. From that experimental data the bulk modulus can be calculated.

A third method is to use a confined compression test. In this technique, a cylindrical specimen is inserted into a thick-walled cylinder, and then compressed with a cylindrical rod with the same diameter as the diameter of the whole in the thick-walled cylinder. [Figure 2.42](#) shows a schematic of the experimental setup.

In this type of experiment, the force-displacement response of the cylindrical push rod, and the strain on the outside surface of the thick-walled cylinder are measured. From these signals the pressure-volume response of the polymer can be determined, from which the bulk modulus is calculated from the slope of the calculated pressure-volume response. One benefit of this approach is that it can be performed at a wide temperature and pressure range.

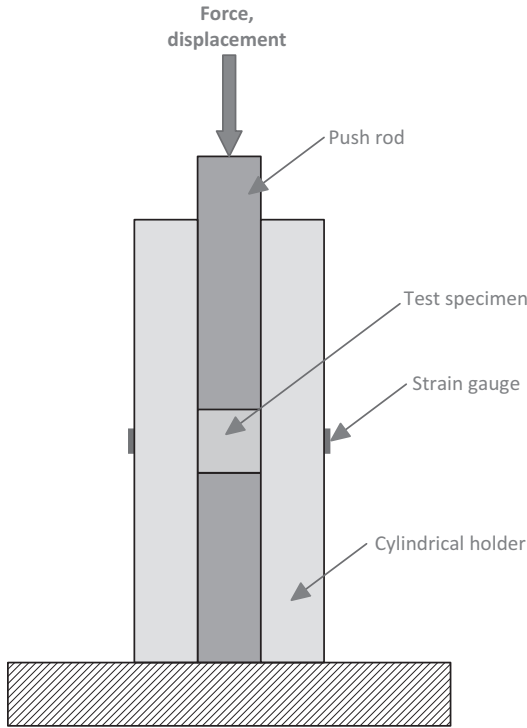


**Figure 2.41** Influence of the bulk modulus (and Poisson’s ratio) on the predicted response of an o-ring in a confined geometry without a small groove. The predictions were obtained from the BB model.

In small strain volumetric loading, the bulk modulus can be obtained from Hooke’s law (see Chapter 5, Section 5.2.1):

$$\kappa = -\frac{P}{\varepsilon_v} = -\frac{\Delta P}{\Delta \varepsilon_v}, \tag{2.9}$$





**Figure 2.42** Schematic figure of a bulk compression fixture.

where  $P = -(\sigma_r + \sigma_\theta + \sigma_z)/3$  is the pressure, and  $\varepsilon_v = \varepsilon_r + \varepsilon_\theta + \varepsilon_z$  is the volumetric strain.

In this case,  $\varepsilon_r = \varepsilon_\theta = 0$ , giving  $\varepsilon_v = \varepsilon_z$ , and  $\kappa = -P/\varepsilon_z$ . The axial strain  $\varepsilon_z$  is given by the applied deformation that is directly measured in the experiment. The experiment also measures the axial stress  $\sigma_z = F/A$ , but to calculate the bulk modulus it is necessary to know the *pressure*, not only the axial stress.

From Hooke's law with  $\varepsilon_r = \varepsilon_\theta = 0$  it directly follows that  $\sigma_r = \sigma_\theta$ . The radial stress in the specimen can be obtained from the known closed-form solution of a pressurized thick-walled cylinder [23]:

$$\sigma_r = \sigma_\theta^{\text{gauge}} \cdot \frac{b^2 - a^2}{2a^2}, \quad (2.10)$$

where  $\sigma_\theta^{\text{gauge}}$  is the circumferential stress from the strain gauge on the outside of the cylindrical holder,  $b$  is the outside radius of the cylindrical holder, and  $a$  is the radius of the test specimen.

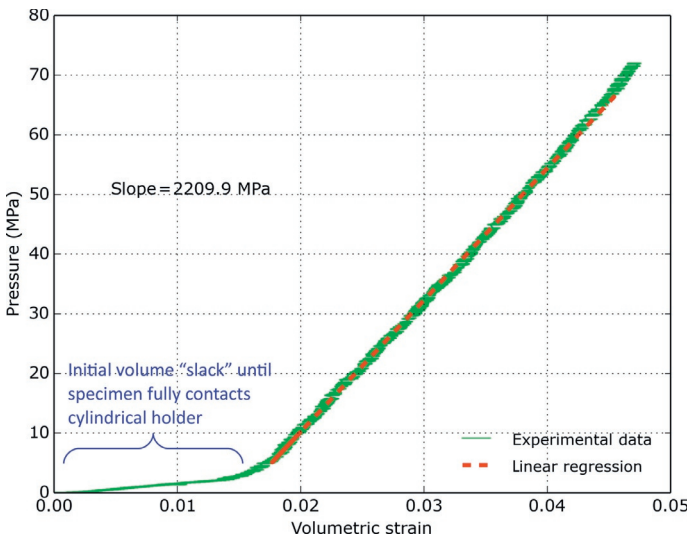
From these equations the pressure can be calculated from:

$$P = -\frac{1}{3} \left[ \frac{F}{A_0} + \frac{b^2 - a^2}{a^2} \varepsilon_{\theta}^{\text{gauge}} E_{\text{holder}} \right], \quad (2.11)$$

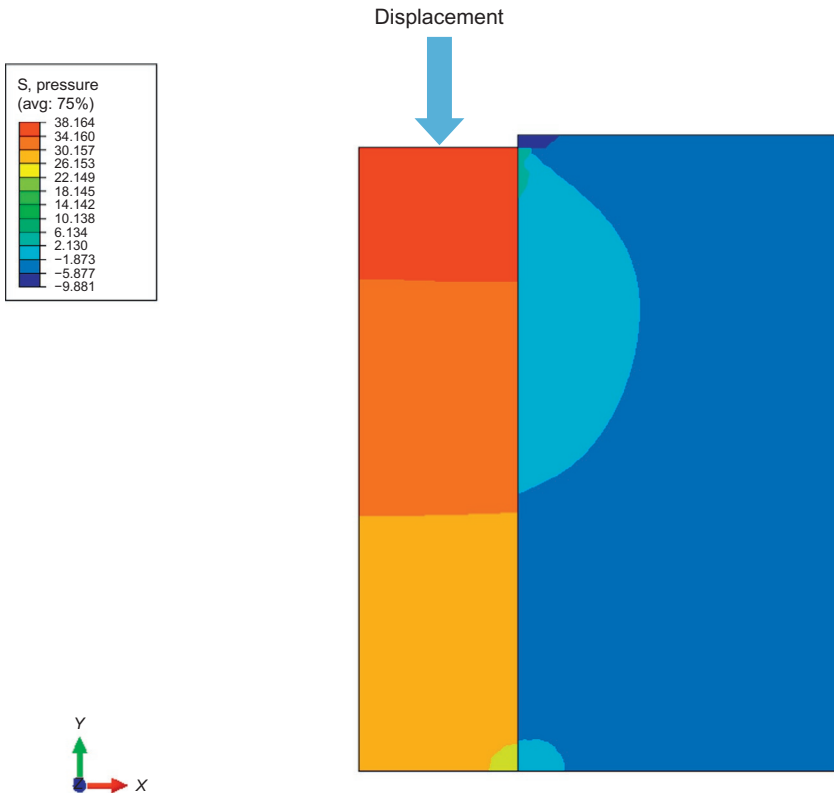
where  $\varepsilon_{\theta}^{\text{gauge}}$  is the circumferential strain from the strain gauge and  $E_{\text{holder}}$  is the Young's modulus of the holder.

One example showing experimental bulk compression data is shown in [Figure 2.43](#). In this case, a cylindrical PTFE specimen was compressed in the confined compression holder. The results from the test show that the initial response consists of a region of low modulus, corresponding to the initial specimen compression until there is full contact with the walls of the fixture. At larger strain, the slope of the pressure-volumetric strain curve gives the real bulk modulus of the material. For this PTFE material the bulk modulus was 2210 MPa.

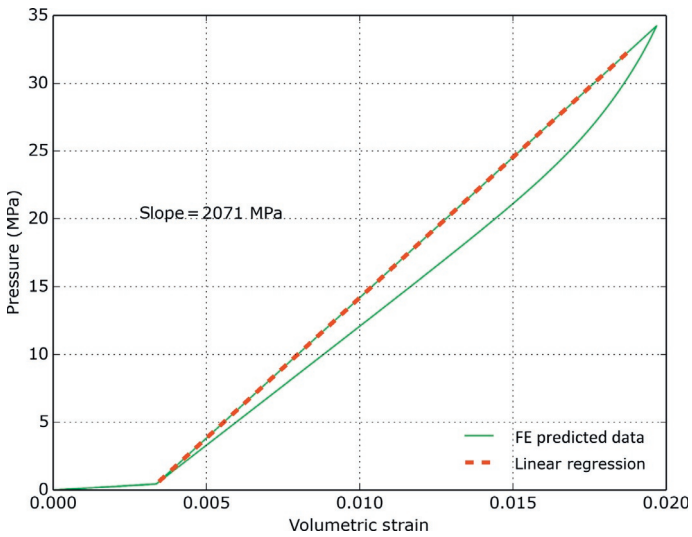
Like all experimental tests, there are complications also with the bulk compression test. One common way to examine the accuracy of an experimental technique is to perform an FE study of the experimental setup. One example of this approach is shown in [Figures 2.44](#) and [2.45](#).



**Figure 2.43** Experimental volumetric data for a PTFE material. The slope of the pressure-volumetric strain data is the bulk modulus.



**Figure 2.44** FEA predicted distribution of pressure in a volumetrically compressed test specimen. Axisymmetric model of the test.



**Figure 2.45** FEA predicted pressure-volume response.

In this example, a cylindrical specimen with a diameter of 6 mm and a height of 25 mm was volumetrically compressed in a cylindrical holder with an interior diameter that was just slightly larger than the diameter of the specimen. The specimen was represented using the TN model (see Section 8.6) with a bulk modulus of 2000 MPa. The friction coefficient between the specimen and the cylindrical holder was taken to be 0.05.

Figure 2.44 shows that the pressure distribution becomes inhomogeneous due to the large initial specimen height. Clearly, the lower the friction the more accurate the experiments become.

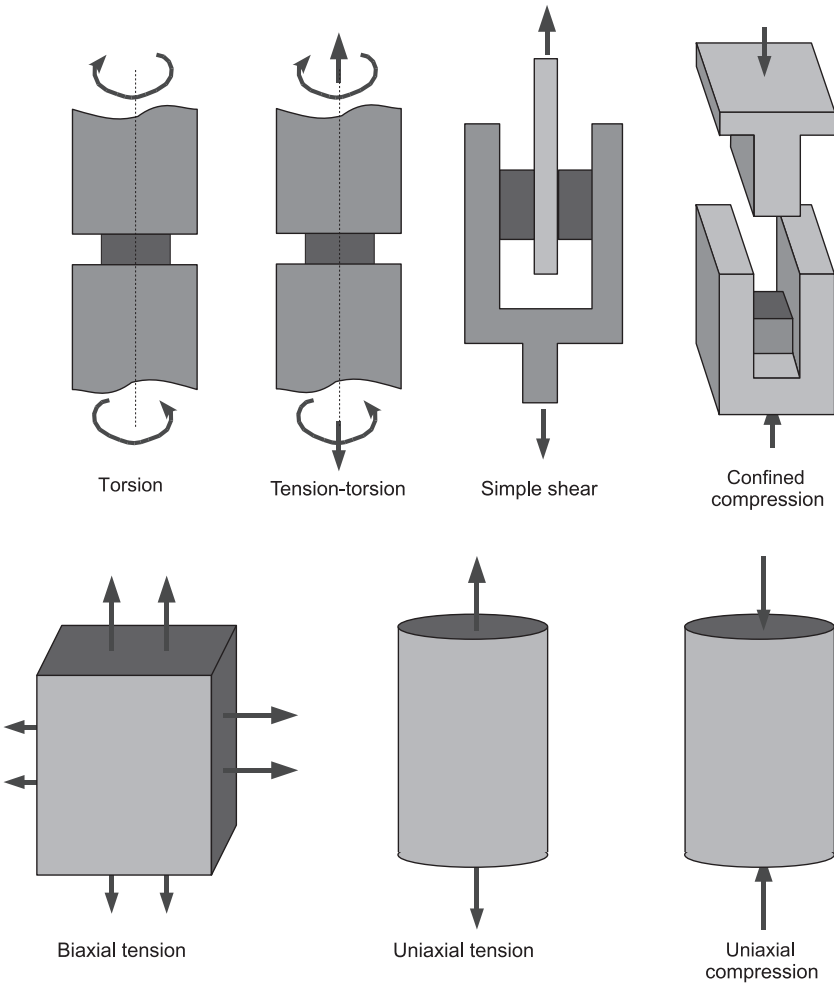
From the FE simulation the axial force, the applied displacement of the top surface of the specimen, and the strain on the outside surface of specimen holder can be extracted. From this data the pressure and volumetric strain can be calculated from Equation (2.11), see the results plotted in Figure 2.45. The figure shows that the slope of the extracted pressure-volumetric strain data is about 2071 MPa, which is 3.5% higher than the real bulk modulus. It is also interesting to see that there is hysteresis in the pressure-volumetric strain response during cyclic loading. This hysteresis can also be seen in real experimental data.

Further FE analysis shows that the accuracy of this experimental setup can be further improved by using specimens that are less tall, and therefore reduce the influence of frictional forces.

### 2.2.10 Other Common Mechanical Testing Modes

Other loading modes that are commonly used to determine the stress-strain response of polymers are shown in Figure 2.46. These loading modes include torsion, combined tension-torsion, biaxial tension, 3-pt and 4-pt bending, and confined compression. It is not necessary to use all of these different experimental tests in order to characterize the behavior of a polymeric material.

Depending on what material model is used, it is sometimes recommended to perform tests in uniaxial loading and at least one of the other loading modes. For polymer foams, due to their compressibility, it is important to also run confined compression or triaxial compression in order to separate the shear response



**Figure 2.46** Different loading modes that are commonly used for determining the stress-strain response of polymers.

from the volumetric response of the material. The number of different tests and types of loading modes that are required for material model calibration is dependent on the required accuracy of the model predictions and the material model that is chosen. Interestingly, the more advanced material models that are micromechanically motivated (see Chapter 8) typically require fewer loading modes for characterization compared to the purely phenomenological models that often need many different loading

modes for accurate and reliable calibration. The advanced micromechanically motivated models, on the other hand, often need experimental data also for different loading rates and temperatures for calibration.

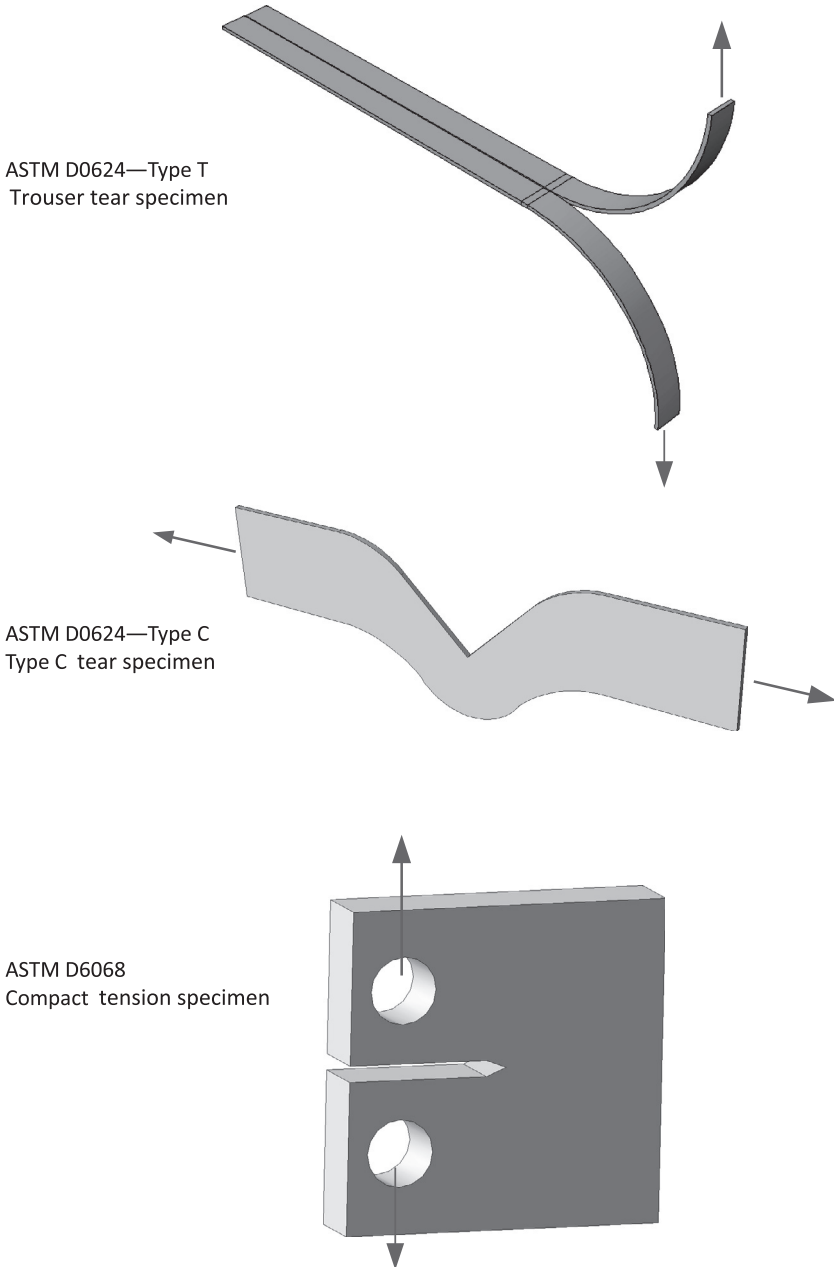
### 2.2.11 Testing for Failure Model Calibration

Experimental testing is also critical for calibrating failure models for all types of polymers. This includes fracture mechanics testing, fatigue testing, and damage mechanics testing. In general, the failure testing can be divided into testing of continuum-level failure properties and fracture-mechanics testing. The continuum-level testing is based on the exact testing techniques that have been discussed in the previous sections of this chapter. These tests are either applied in monotonic or cyclic loading to failure depending on the desired type of failure model.

The fracture mechanics testing is performed differently for elastomers and thermoplastics. For elastomers, a number of test geometries have been developed that are particularly useful for determining the critical tearing energy of an elastomer or rubber. [Figure 2.47](#) shows a few common test geometries for this purpose. The Trouser Tear specimen and the Type C Tear specimen are specific specimen geometries for determining the tear strength of rubbers [24], and the compact tension specimen has been developed for measuring the fracture toughness of plastic materials [25].

## 2.3 Mechanical Testing for Material Model Validation

The previous section introduced a large number of experimental tests that can be used to quantify the mechanical response of a polymer, but the results from those tests can also be used as the source for the selection and calibration of different materials model. Practical aspects of material model calibration is discussed in more detail in Chapter 9.



**Figure 2.47** Different loading modes that are commonly used for determining the failure response of polymers.

Many times, specifically in industrial settings, the material model calibration is the final step in the material modeling work after which the calibrated material models are inserted into FE input files. This approach is tempting since the end goal is typically the FE analysis of a product or design, or to guide a design optimization. The accuracy of the FE solver is usually excellent as long as the FE mesh is sufficiently fine, but the accuracy of product-level FE results are *not* guaranteed unless suitable and accurate material models have been used. For that reason, it is in general recommended to validate the calibrated material models by comparing FE predictions, that are based on the selected and calibrated material model, to a set of experimental data that was not used for the material model calibration. This new set of experimental data are the *validation test results*. It is typically recommended that the validation tests be performed in a similar loading mode as the intended application, or in a general multiaxial loading mode.

Sometimes the results from an experiment can be used for both material model calibration and material model validation, even if the deformation state in the test specimen is inhomogeneous. If the deformation state is inhomogeneous, then *inverse calibrations* can be used to calibrate the selected material model. Inverse material model calibration is a technique in which an FE simulation is setup of the actual test and the results from that test is then used iteratively as part of the model calibration.

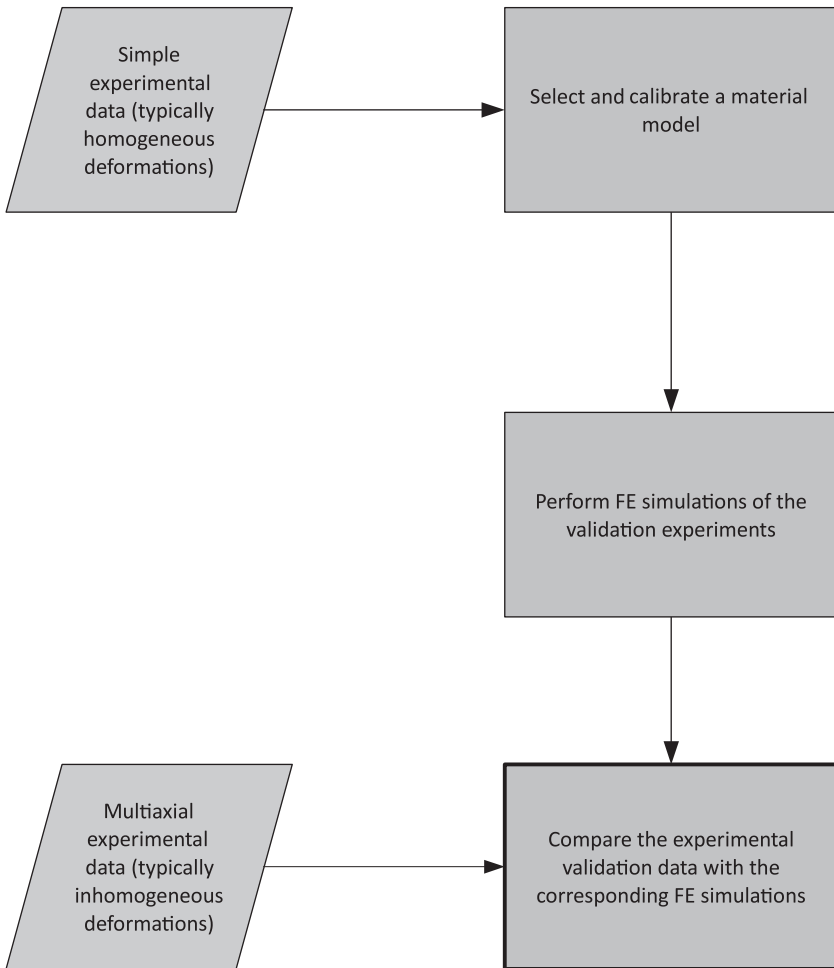
The typical steps that are performed as part of material model validation are shown in [Figure 2.48](#).

### 2.3.1 Material Model Verification and Validation

Computer simulation models are often used as an important step in product design, problem solving, and used as a tool to provide insight into processes. For this reason it is important to make sure that the simulation models are *verified* and *validated*, in order to make sure the simulation results are as accurate as expected.

Material model *verification* is a process that is used to ensure that the model has been implemented properly with respect to





**Figure 2.48** Material model validation consists for multiple steps: calibration, multiaxial testing, and validation simulations.

the constitutive equations. This typically include code review, and examining the model predictions from different solution schemes and implementations. Most often this step has been performed by the FE solver provider, or the material model provider. If you are implementing your own material models then you should perform this step very carefully.

Material model *validation* is a process that is used to ensure that the model sufficiently accurately captures the response of a real material. Often validation is limited to a certain range of temperatures and strain-rates.

**Material Model Validation:**

“Are you using the right material model?”

**Material Model Verification:**

“Has the material model been implemented right?”

The following sections provide some common experimental tests that can be used for material model validation.

### 2.3.2 Small Punch Testing

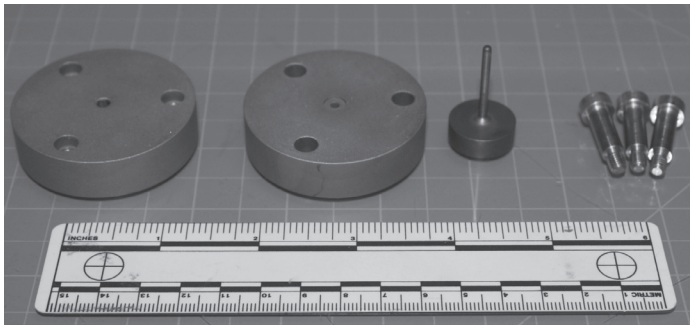
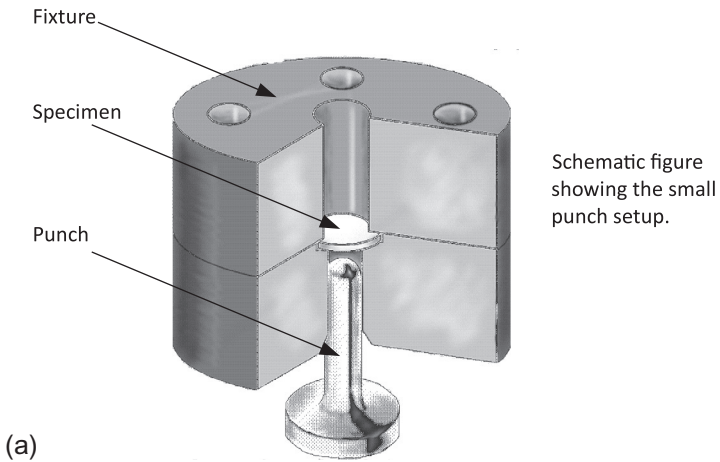
Small punch testing is an experimental technique that can be used to examine the biaxial bending response of thermoplastic and elastomeric materials. The details of the experimental approach is shown in [Figure 2.49](#) and is also described in the ASTM standard F2183 [26]. In this test, a thin cylindrical specimen with a diameter of 6.4 mm and a thickness of 0.5 mm is placed into a steel fixture and then loaded using a hemispherical punch head that drives the specimen to deform in a loading mode similar to biaxial drawing, see [Figure 2.50](#). During the test the force-displacement response of the punch head is measured.

The main benefit of this test is that can be performed on small specimens that can be extracted from, for example, retrieved medical devices [27–29]. Since the experimental test is easy to perform and since it introduces a multiaxial stress and strain state in the material, it is also a useful material model validation test. Validation testing is discussed in more detail in [Section 2.3](#).

The small punch test is easy to analyze using FE simulations. [Figure 2.50](#) shows an exemplar axisymmetric FE mesh of a small punch specimen.

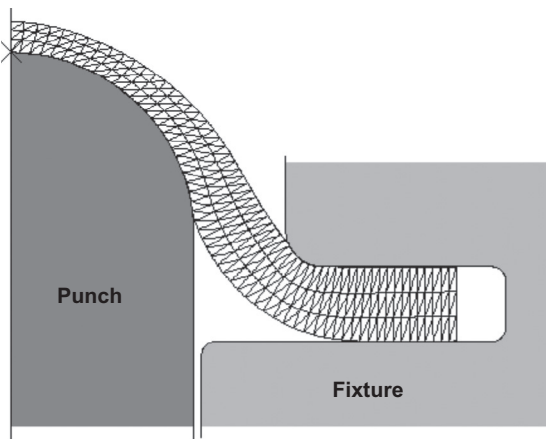
The main challenge with the small punch test is the strong influence of friction. [Figure 2.51](#) shows the results from a case study in which three friction coefficients were examined (0, 0.1, and 0.2). The figure shows that at large deformations the force-displacement response can be rather strongly dependent on the friction coefficient.

The table below shows how the friction coefficient influence the max stress, max strain, and the minimum specimen thickness.

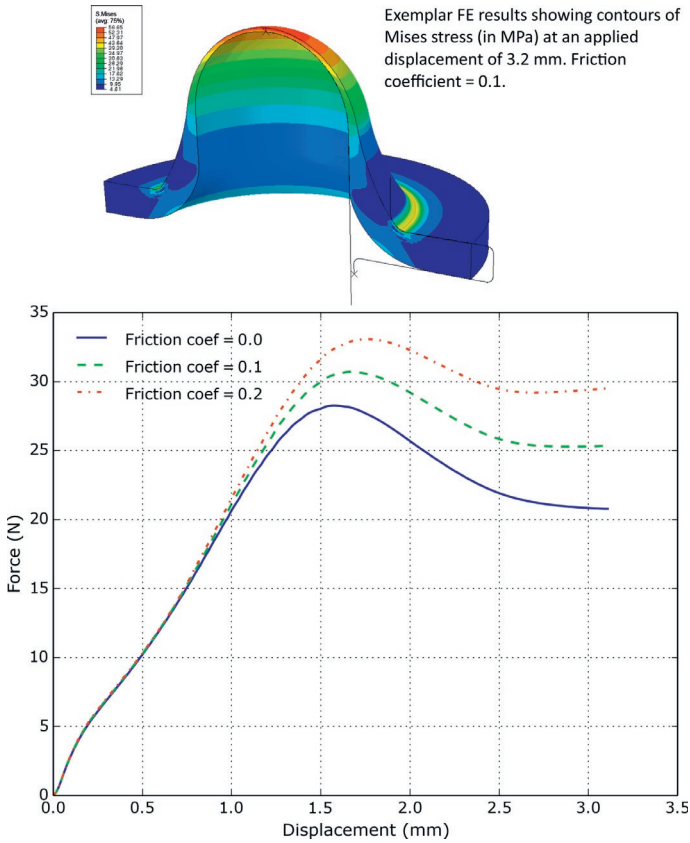


(b) Photograph of a disassembled small punch fixture .

**Figure 2.49** Schematic figure of a small punch fixture. (Reproduced with permission from Veryst Engineering, [www.veryst.com](http://www.veryst.com).)



**Figure 2.50** Schematic FE mesh of a small punch specimen being loaded by a hemispherical punch head.



**Figure 2.51** Results from an FE study of the influence of friction on the small punch results.

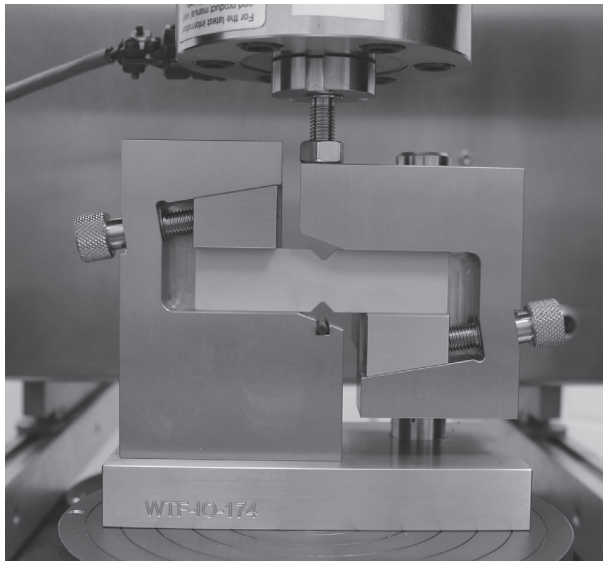
These results clearly show that at large punch displacements all stress and strain fields in the specimen are strongly dependent on the friction coefficient. Hence, in order to effectively use this multiaxial test it is important to have a good understanding of the friction coefficient between the test specimen and the steel material in the fixture.

Friction Coefficient	Max Mises Stress (MPa)	Max Principal Strain	Min Thickness (mm)
0	76.5	1.41	0.47
0.1	56.7	1.22	0.45
0.2	45.3	1.09	0.43

### 2.3.3 V-Notch Shear Testing

V-notch shear is another common experimental method for examining the multiaxial stress and strain response of a polymer. The V-notch shear is an attractive method since it mainly probes the shear behavior, and it does not suffer from uncertainties due to friction. This experimental method is also called *Iosipescu Shear* and described in ASTM D5379 [30]. A photograph of the shear fixture is shown in Figure 2.52.

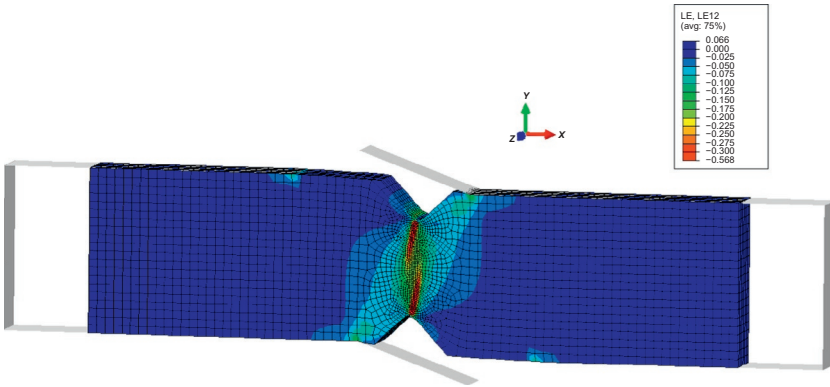
The stress field in the V-notch specimen can be examined using an FE study. Figure 2.53 shows one example where a specimen modeled using the TN model with material parameters suitable for UHMWPE.



**Figure 2.52** Photograph of a V-notch shear fixture. (Reproduced with permission from Veryst Engineering, [www.veryst.com](http://www.veryst.com).)

## 2.4 Surface Characterization Techniques

There are numerous experimental techniques that can be used for characterizing surface topology, structure, and chemistry of



**Figure 2.53** Predicted distribution of shear strain in the V-notch shear test.

polymers. The most important and commonly used techniques are reviewed in the following sections.

### 2.4.1 Optical Microscopy

Optical microscopes are designed to create magnified visual or photographic images of small objects. To accomplish this, the microscope is designed to perform three tasks: create a magnified image of the sample, distinguish different details of the image, and make the final image visible to the human eye or camera. This class of characterization instruments includes everything from a simple magnifying glass to advanced multi-lens microscopes.

There are two basic ways in which optical microscopes can be operated depending on how the light is projected on the sample. The first approach, which is widely used to study biological polymers, is to prepare very thin specimens and have the light transmit through the sample. The second approach is used for thicker-section samples and nontransparent materials. Here, the light pass through the objective and then reflects from the surface of the sample and into the microscope objective. These two techniques are referred to as *transmitted* and *reflected light microscopy*.

The resolution limit of optical microscopes is controlled by diffraction, which in turn is controlled by the numerical aperture

( $A_N$ ) of the optical system and the wavelength of the light used ( $\lambda$ ). Assuming that optical aberrations are negligible, the resolution ( $d$ ) is given by:

$$d = \frac{\lambda}{A_N}. \quad (2.12)$$

By assuming  $\lambda = 550$  nm (corresponding to green light), and  $A_N = 1.5$  (corresponding to oil as a medium), the maximum resolution becomes about  $0.3 \mu\text{m}$ . One of the main drawbacks of optical microscopy is this relatively large resolution limit.

Another limitation of optical microscopy is the poor contrast produced when light is passed through very thin specimens or reflected from surfaces with a high degree of reflectivity. Different optical techniques have been developed to improve the contrast of optical microscopy. Some of the more important and commonly used techniques are: polarized light, phase contrast imaging, differential interference contrast, fluorescence illumination, dark-field illumination, Rheinberg illumination, Hoffman modulation contrast, and the use of various gelatin optical filters. Some of these techniques are briefly discussed in the following. A more detailed discussion of these and other optical microscopy techniques are given in [31, 32].

- **Polarized Light Microscopy**

This technique uses optical anisotropy to reveal the structure of the sample. The microscope is equipped with a polarizer and an analyzer. The enhanced contrast results from interaction of the plane-polarized light with a birefringent specimen (doubly refracting) to create two perpendicular optical waves. Due to the interaction with the specimen, the velocities of the two waves are different, causing the two waves to be out of phase. The light components are combined with interference as they pass through the analyzer. This technique can enhance the contrast and quality of the final image.

- **Darkfield Microscopy**

By using specialized oblique illumination, the contrast can be improved of specimens that are normally not well captured using brightfield illumination. In the optical equipment, the direct light is stopped by an opaque block in the condenser, but light from oblique angles at all azimuths passing through the specimen are reflected, refracted, and diffracted into the microscope objective, creating a high contrast image with a dark background.

- **Differential Interference Contrast**

Differential interference contrast microscopy is a beam-shearing interference system in which the reference beam is sheared by a very small amount. The technique creates a shadow-cast image that increases contrast from the gradient of optical paths for both high and low spatial frequencies present in the specimen.

- **Confocal Microscopy**

Confocal microscopy is a new interesting technique that has the ability to control the depth of field, and collect serial optical sections from thick specimens. The approach is to use spatial filtering to remove out-of-focus light. The technique can be used to create very high-quality images and has become an important tool for optical microscopy.

- **Near-Field Scanning Optical Microscopy**

Near-field scanning optical microscopy (NSOM) can be used to create ultra-high optical resolution. To get the high resolution a sub-micron optical probe, positioned very close to the sample, transmits light through a small aperture. The region within a single wavelength from the specimen surface is defined as the near-field. Within this region, evanescent light is not diffraction limited and nanometer resolution can be obtained.



- **Fluorescence Microscopy**

Fluorescence microscopy is primarily used with episcopic illumination and is rapidly becoming a standard tool in the fields of genetics, and cell biology.

- **Stereo Microscopy**

Stereo microscopes have a different design than traditional microscopes. It uses two eyepieces (or sometimes two complete microscopes) to provide slightly different viewing angles to the left and right eyes. This produces a three-dimensional (3D) visualization of the sample being examined. The stereo microscope is often used to study the surfaces of solid polymers or investigate fracture surfaces.

## 2.4.2 Scanning Electron Microscopy

Scanning electron microscopy (SEM) is one of the most important surface characterization techniques that is used today. One of the major advantages of SEM is the great depth of field. In addition, SEM equipment are often coupled with energy dispersive spectroscopy (EDS) X-ray diffraction, see [Section 2.6.2](#), thereby enabling both high magnification characterization of surfaces and elemental composition analysis.

The scanning electron microscope uses electrons rather than light to form an image. This enables a larger depth of field, which allows a larger amount of the sample to be in focus at one time. The SEM also produces images of high resolution. Preparation of the samples is relatively easy since most SEMs only require the sample to be conductive. For polymers this can be achieved, for example, by gold coating the sample. The combination of higher magnification, larger depth of focus, greater resolution, and ease of sample observation makes the SEM one of the most heavily used instruments in experimental materials characterization.

In a typical SEM setup, electrons are emitted from a cathode filament toward an anode. The electron beam, which is typically

given an energy ranging from a few keV up to 50 keV, is focused by condenser lenses into a beam with a very fine spot size (about 5 nm). The electron beam then passes through the objective lens, and scanning coils which deflect the beam in a desired pattern on the sample surface. When the primary electrons reach the surface they are inelastically scattered by atoms in the sample. These scattering events cause the primary beam to spread and create emission of electrons and X-rays, which are then detected and used to produce an image of the surface.

In the most common imaging mode, low energy secondary electrons are detected by a scintillator-photomultiplier component and the resulting signal is used to create an image of the sample. Because the secondary electrons come from within 1 nm of the surface, the brightness of the signal depends on the surface area that is exposed to the primary beam. This surface area is relatively small for a flat surface, but increases for steep surfaces. Thus steep surface and edges (cliffs) tend to be brighter than flat surfaces resulting in images with good 3D contrast. Using this technique, resolutions of the order of 5 nm are possible.

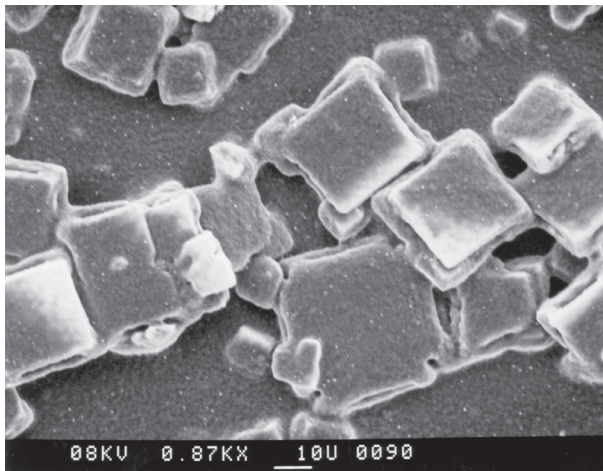
Another mode to operate an SEM is to detect backscattered electrons which are essentially elastically scattered primary electrons. Backscattered electrons may be used to detect both topological and compositional detail, although due to their much higher energy (approximately the same as the primary beam) these electrons may be scattered from fairly deep within the sample. This results in less topological contrast than for the case of secondary electrons. However, the probability of backscattering is a weak function of atomic number, thus some contrast between areas with different chemical compositions can be observed especially when the average atomic number of the different regions is quite different.

The spatial resolution of the SEM is controlled by the size of the spot of the electron beam, and the size of the material which interacts with the electron beam. Since these are larger than the interaction distance it is not possible to image atomic scale images.

A new surface characterization technique is *environmental SEM* (ESEM). This approach works with controlled environmental conditions and requires no conductive coating on the specimen. This enables studies of specimens in their natural state. The environment in an ESEM can be selected from among water vapor, air, N<sub>2</sub>, Ar, O<sub>2</sub>, etc. Dynamic characterization of wetting, drying, absorption, melting, corrosion, and crystallization can be performed using ESEM.

ESEMs are able to work with certain pressures and without surface charging because the secondary electron detector is designed on the principle of gas ionization. As primary electrons are emitted from the gun system, the secondary electrons on the specimen surface are accelerated toward the detector, which is biased by a moderate electric field. The collisions between the electrons and gas molecules liberate more free electrons, and thereby provide more signals. Positive ions created in the gas effectively neutralize the excess electron charge built up on the specimen.

One exemplar SEM image is shown in [Figure 2.54](#). This figure shows the topology and structure of Tin oxide particles on a polyetherimide (PEI) substrate. The bar in the legend of the figure is 10  $\mu\text{m}$  long.



**Figure 2.54** Exemplar SEM image showing Tin oxide particles on a PEI substrate.

### 2.4.3 Atomic Force Microscopy

The atomic force microscope (AFM) is a powerful high magnification microscope that was developed by Binnig, Quate, and Gerber in 1986 [33]. The AFM uses a cantilever with a sharp tip to study surface features. During operation the cantilever tip is brought into close proximity of the sample surface. The force from the sample on the cantilever beam causes a deflection of the cantilever beam tip, and this deflection is measured using laser techniques. Unlike traditional microscopes, AFM does not use lenses, hence the size of the probe tip rather than diffraction typically control the experimental resolution.

The AFM typically operate in one of two modes: (1) contacting mode with the tip contacting the specimen and (2) noncontacting or tapping mode. In contacting mode, the specimen is mounted on a piezoelectric crystal and the cantilever deflection is monitored in order to keep the distance between the probe tip and the sample constant. The tip is then scanned across the specimen surface and the vertical displacement necessary to maintain a constant force on the tip is recorded. The resulting height map represents the topography of the specimen surface (Figure 2.55).

In noncontacting or tapping mode, the cantilever is externally oscillated close to its resonance frequency. The vibration characteristics of the cantilever beam is influenced by the tip-sample interaction forces; these changes in oscillation provide information about the specimen geometry. One of the main advantages of the dynamic mode is that it generates lower lateral forces on the sample, and it is therefore widely used to image biological samples.

The noncontacting dynamic mode can be run using frequency modulation, or more commonly amplitude modulation. In amplitude modulation, changes in the oscillation amplitude yield topographic information about the sample. Additionally, changes in the phase of oscillation under tapping mode can be used to discriminate between different types of materials on the surface.

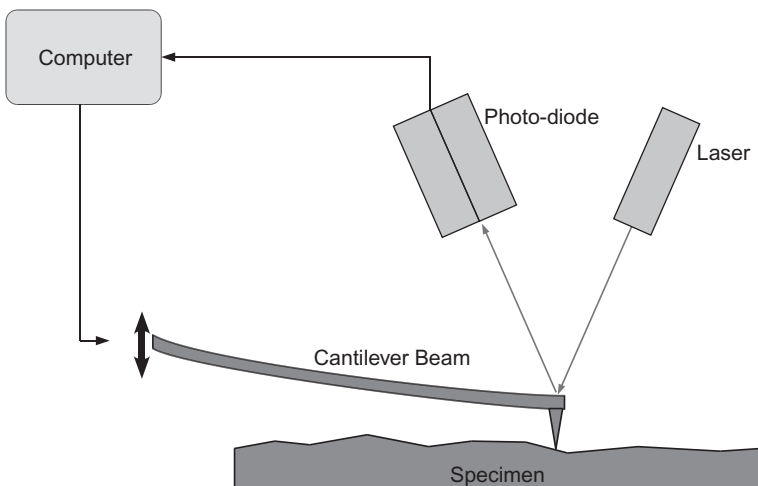
The AFM has several advantages over electron microscopy. One is that it provides a true 3D surface profile. Additionally,

samples viewed by an AFM do not require any special treatment that can destroy the sample and prevent its reuse. While an electron microscope needs a vacuum environment for proper operation, the AFM can be used also in an ambient or even liquid environment.

The main disadvantage of the AFM compared with the SEM is the image size. The SEM can show image representations of a few  $\text{mm}^2$  area and a depth of field of a few  $\text{mm}$ . The AFM, on the other hand, can only show a maximum area of about  $0.01 \text{ mm}^2$  with a max depth of field of a few  $\mu\text{m}$ .

Another common use of AFM is to measure the indentation resistance and stiffness of a specimen by measuring the force required to depress the probe tip a small distance into the specimen. In addition to this indentation-type test, AFM are commonly used also in the following applications:

- Examine dispersion and particle size of additives in polymer matrix.
- Study phases of blends and alloys.
- Image and quantify surface texture or roughness (topography).



**Figure 2.55** Schematic view of an AFM.

## 2.5 Volume Characterization Techniques

There are numerous experimental techniques that can be used to study the bulk properties of polymers. These volume characterization techniques are typically very different from the surface characterization techniques that were discussed in the previous section. The most common volume characterization techniques are discussed in the following sections.

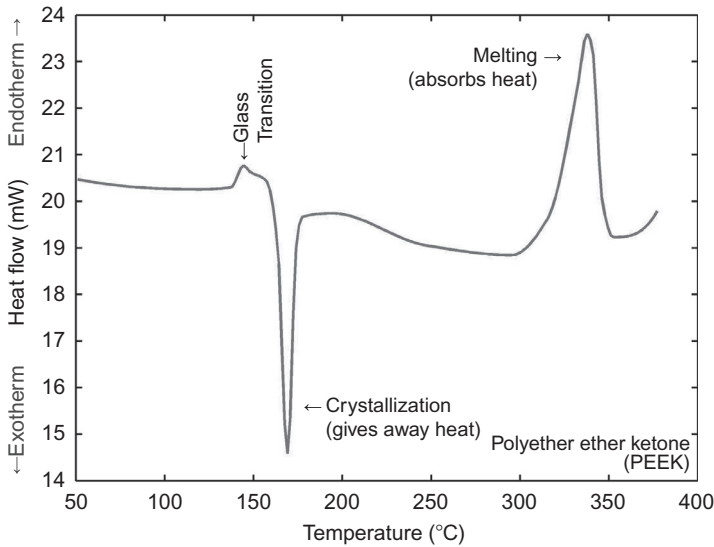
### 2.5.1 Differential Scanning Calorimetry

Differential scanning calorimetry (DSC) is a commonly used technique for analyzing thermal transitions in polymers and other materials. It provides information about the temperatures at which transitions occur as well as quantitative measurements of the heat transitions associated with the events.

All materials that undergo a change in physical state (e.g., recrystallization or melting), or undergo a chemical reaction, will release absorbed energy as part of the transition. A differential scanning calorimeter is designed to measure the enthalpy changes that occur during these transitions. In the experimental setup, both the test specimen and a reference sample are heated at a predefined temperature rate and the differential heat flow required to maintain the sample at the same temperature as the inert reference specimen is measured.

Results from a typical DSC run are shown in [Figure 2.56](#). In this experiment, a small sample (about 10 mm<sup>3</sup>) was heated at a rate of 20 °C/min. The figure shows the input heat flow that was required to maintain the prescribed temperature history. The graph provides direct information about the glass transition temperature ( $T_g$ ), the exotherm associated with crystallization, and the endotherm associated with melting.

DSC can also be used to study the influence of mechanical deformation and residual stresses on the required heat flow for the thermal transitions [34]. By coupling mechanical testing and thermal pretreatments it is possible to explore the micromechanisms controlling the activation energy for local rearrangements.



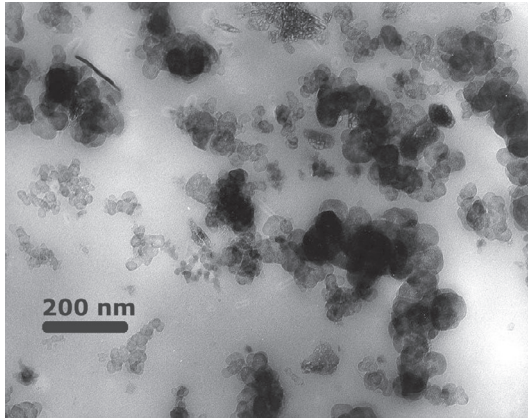
**Figure 2.56** Exemplar DSC scan of PEEK showing the glass transition temperature, the crystallization peak, and the endotherm during melting.

This method can provide valuable information about the processes that must be accounted for in the development of accurate constitutive models for the material behavior.

## 2.5.2 Transmission Electron Microscopy

Transmission electron microscopy (TEM) is a powerful technique that can produce higher magnification than what can be achieved by SEM. One of the main limitations of TEM is that the test specimen must be a foil so thin that an electron beam can pass through it. In addition, the specimen must be able to withstand the high vacuum inside the TEM instrument. Because of these requirements, most materials studied with TEM are of inorganic, nonbiological materials.

The TEM is widely used in materials science and metallurgy, and for the study of crystalline materials such as metals and semiconductors, but is also very useful for studies of filler morphology in polymers. One example of this is shown in [Figure 2.57](#)



**Figure 2.57** TEM of CR with 7 vol% CB [35].

illustrating a chloroprene rubber (CR) filled with 7 vol% N600 carbon black (CB). The black regions in the figure are the carbon black particles. The micrograph shows that the particles have a spherical shape with a diameter of 20 nm, and that the carbon black particles have a tendency to cluster together into larger aggregates.

The contrast in a TEM image is different from the contrast in a light microscope image. A crystalline material interacts with the electron beam mostly by diffraction rather than absorption. If the planes of atoms in a crystal are aligned at certain angles to the electron beam, the beam is transmitted strongly; while at other angles, the beam is diffracted, sending electrons in another direction. In the TEM, the specimen holder allows the user to rotate the specimen to any angle in order to establish the desired diffraction conditions.

In the most powerful diffraction contrast TEM instruments, it is possible to produce a diffraction pattern image which is directly analogous to the planes of atoms in the crystal. Although the way contrast arises in these atomic-resolution images is complex, and such images are often interpreted using computer modeling of the electron beam and magnetic lenses, these images have added a new layer of understanding to the study of crystalline polymer materials.



### 2.5.3 X-Ray Diffraction

X-ray diffraction is an important tool for studying both amorphous and semi-crystalline polymers [36, 37]. It can be used to analyze many features of the microstructure of the material, including lattice parameters, presence of imperfections, crystallographic orientations (texture), and degree of crystallinity.

In a typical X-ray diffraction experiment, the polymer sample is exposed to X-ray radiation which is a form of electromagnetic radiation with a short wave length ( $\lambda \approx 0.1$  nm). The X-ray radiation is often produced by bombarding a metal target with fast electrons in a vacuum tube. When created this way, the radiation often consists of two components: a continuous spectrum of white radiation, and a superimposed line spectrum having a frequency that depends on the metal being bombarded. When the X-ray beam transverses through the specimen it loses its intensity exponentially:  $I = I_0 \exp(-\mu x)$ , where  $I_0$  is the initial intensity and  $\mu$  is a linear absorption constant. Hence, the depth of X-ray penetration depends on the material and the energy of the X-ray source.

The incident beam of X-rays is partly absorbed, partly scattered, and the rest is transmitted unmodified through the specimen. The scattering occurs due to interaction between the incident X-ray beam with the electrons in the material, and the diffracted X-rays interact with each other to cause diffraction patterns that depend on the angle of the incident beam in relation to the specimen orientation. In this context, the words scattering and diffraction are used interchangeably.

X-ray diffraction can also be used to study amorphous polymers. The diffraction pattern from amorphous polymers is lacking sharp peaks that are characteristic of crystals, and instead consists of broad features. Quantitative diffraction analysis of amorphous polymers can provide important information about the local atomic structure, including bond lengths, morphology, and radial distribution information.

X-ray diffraction techniques are often categorized as wide-angle X-ray scattering (WAXS) and small-angle X-ray scattering

(SAXS). WAXS is typically used to study structures with a length scale of about 1 nm, and SAXS is used to study larger features with a length scale of 1 nm to about 400 nm.

### Wide-Angle X-Ray Diffraction

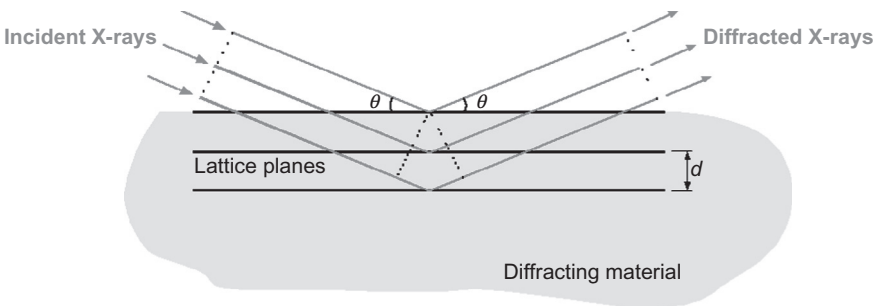
The principle of wide-angle X-ray diffraction can be exemplified by considering a beam of X-rays with a wavelength  $\lambda$  impinging at an angle  $\theta$  on a diffracting material with a set of crystal planes of spacing  $d$ , see [Figure 2.58](#).

A diffracted X-ray beam at the angle  $\theta$  will only exist if the rays from each successive lattice plane reinforce each other. For this to occur, the extra distance a ray has to travel must be equal to an integral number of wavelengths,  $n\lambda$ . This is expressed in the well-known Bragg law:

$$n\lambda = 2d \sin \theta, \quad (2.13)$$

where  $n$  is an integer,  $\lambda$  is the wavelength of the X-ray beam,  $d$  is the lattice spacing, and  $\theta$  is the angle between the lattice planes and the incident beam.

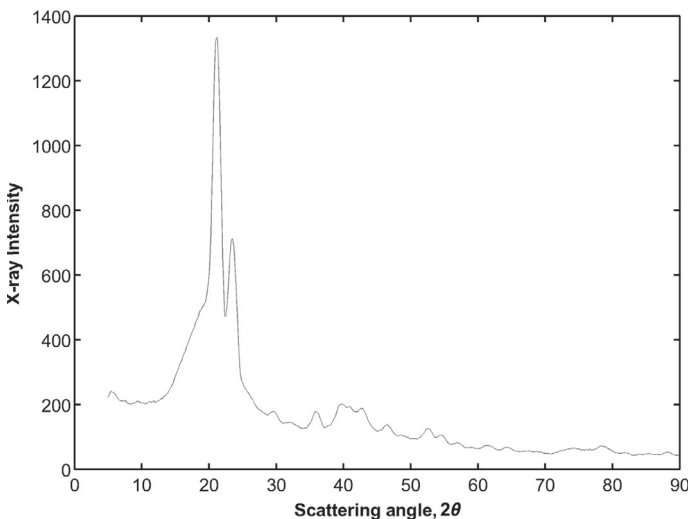
To ensure that Bragg's law is satisfied and that reflections from various crystal planes can occur, it is necessary to provide a range of either  $\theta$  or  $\lambda$ . The two most common methods to achieve this are the *Laue method* and the *powder method*.



**Figure 2.58** Wide-angle X-ray diffraction from crystal planes.

- In the *Laue method*, a stationary crystal is exposed to a beam of white radiation. Since a range of wave lengths is provided, each set of lattice planes will chose the appropriate  $\lambda$  to give Bragg reflection. The resulting refraction pattern can be used to obtain information about the crystalline structure.
- In the *powder method*, monochromatic radiation is used to bombard a finely powdered (or fine-grained) wire of the material. In this case, the Bragg angle  $\theta$  is variable and crystals with the appropriate orientation will give Bragg refraction.

One example of an X-ray diffraction intensity plot is shown in [Figure 2.59](#). The figure shows the scattering intensity as a function scattering angle for a polyethylene. The measured intensity consists of a Bragg diffraction peak from the crystalline phase superimposed on the diffuse scattering from the amorphous phase. By separating the contributions from the amorphous and crystalline phases it is possible to determine the degree of crystallinity of the material [38].



**Figure 2.59** Detected X-ray intensity as a function of scattering angle for polyethylene.

### **Small-Angle X-Ray Diffraction**

Small angle X-ray diffraction is a useful technique for studying structural features with a size between 1 nm and about 400 nm. In a common experimental setup, the X-ray radiation is created using a Cu  $K\alpha$  emission line with a wavelength of  $\lambda = 0.154$  nm giving  $\theta = 4.4^\circ$  and  $\theta = 0.04^\circ$ , for  $d = 1$  nm and  $d = 100$  nm, respectively.

The experimental setup is similar to what is used for WAXS, but since the scattering beam is so close to the unmodified transmitted beam it is often required to use a specialized experimental setup that collimates that incident beam very precisely.

The X-ray diffraction patterns can be determined by measuring the intensity of the scattered X-rays as a function of scattering direction. By studying the diffraction pattern it is possible to, for example, examine individual crystal defects by determining any differences in refracted intensity near the defect.

#### **2.5.4 Birefringence**

Birefringence spectroscopy is an optical technique that can be used to determine molecular orientation in a polymer sample by measuring the retardation of polarized light passing through the sample [39]. The experimental technique is based on the difference in refractive indexes,  $n_i$ , in different directions of the specimen:

$$n_i = \frac{\text{Speed of light in vacuum}}{\text{Speed of light in direction } i \text{ in the sample}}. \quad (2.14)$$

The refractive index for a polymer in a given direction is depending on the polarizability of the molecular bonds and the directional distribution of the macromolecules. For most polymers the refractive index parallel to the main chain encounters primarily C–C bonds, which have low polarizability and hence a refractive index close to 1. In a direction perpendicular to the molecular backbone, many polymers have side-groups that retard the light more creating a higher refractive index.

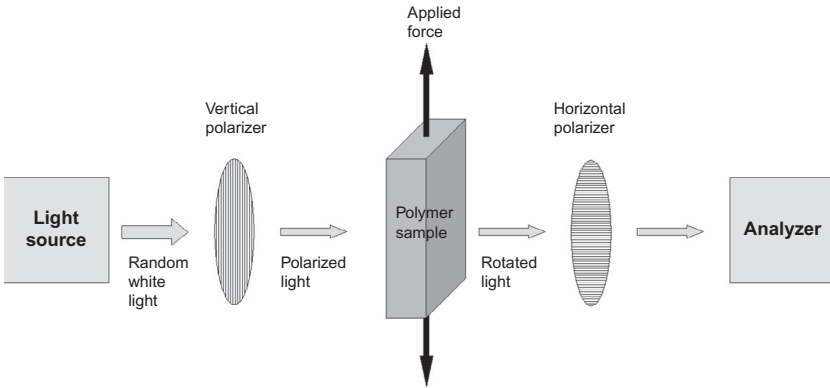
Birefringence  $\Delta n$  is the difference in the refractive indexes of two perpendicular directions in a material:  $\Delta n = n_1 - n_2$ . A polymer sample will have a nonzero birefringence if two conditions are satisfied:

1. *On the molecular scale there must be a finite difference in refractive indexes in different directions.* Virtually all polymers satisfy this condition due to the inherent anisotropic nature of the repeat unit.
2. *On the macroscale there must be an anisotropic orientation of the macromolecules.* Hence, if the molecular chains are randomly oriented then the different refractive indexes will cancel each other out and the birefringence will be zero.

The total birefringence of a polymer specimen is controlled by sum of the polarizability of all molecular chains. This connection between birefringence and molecular orientation is what makes this experimental technique interesting. A common experimental study is to measure the amount of birefringence as a function of applied stress and strain. Since the molecular chains are stretching and rotating during an applied macroscopic deformation, birefringence measurements during deformation can provide interesting insight into these molecular events.

As will be discussed in Chapter 5, a number of different molecular chain models have been developed. These models are used, for example, when developing hyperelastic models of rubbers as they describe the orientation of the individual segments of the molecular chains as a function of applied strain. These theoretical expressions can be used together with statistical methods to derive the birefringence as a function of applied load or deformation. Hence, birefringence can also be a useful tool for the development and validation of micromechanism inspired constitutive models.

Birefringence is also often used to study: molecular defects in polymeric solids, thermal strains and residual stresses, and morphological and structural changes. Birefringence can be used



**Figure 2.60** Experimental setup used for measuring birefringence.

to study everything from a single polymer chain subject to a flow field, amorphous polymers, semicrystalline polymers, and block copolymers.

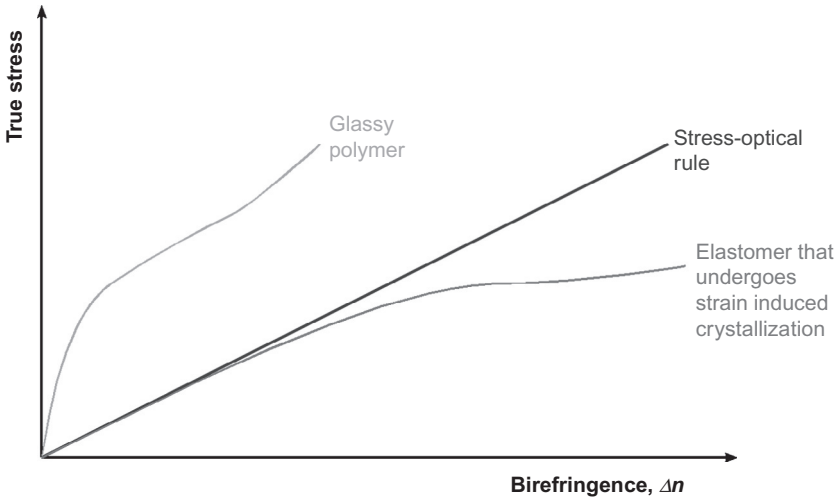
Birefringence in the solid state is often called strain birefringence, and birefringence in a liquid state is often called flow birefringence. Birefringence experiments are typically performed by uniaxially deforming a thin sample and measuring the difference in refractive indexes between the deformation direction and a lateral direction, see [Figure 2.60](#).

The birefringence of many polymer systems at small to moderate deformations is found to follow the *stress-optical rule* SOR:  $\Delta n = C\sigma$ , where  $\Delta n$  is the birefringence,  $\sigma$  is the tensile stress, and  $C$  is the stress-optical constant, see [Figure 2.61](#).

As is shown in this figure, glassy polymers do not follow the stress-optical rule (SOR) even at small strains, whereas elastomers typically follow the SOR at small strains but start to deviate at the onset of strain-induced crystallization.

### 2.5.5 Swell Testing

If a crosslinked polymer is immersed in a good solvent it will swell. The crosslinks in the material will keep the macromolecular structure intact and the material will not dissolve. The equilibrium amount of swelling is controlled by the elastic forces



**Figure 2.61** Illustration of stress-birefringence relationship for a stretching process of different types of polymers.

from molecular stretching that resist the swelling, and the forces derived from the free energy of mixing. The equilibrium amount of swelling can be used to determine the gel content, crosslink density, molecular weight between crosslinks, and the number of repeat units between crosslinks. These terms can be defined as follows:

- $g$ , gel content—the mass percentage of polymer insoluble in a specific solvent after extraction.
- $\nu_d$ , crosslink density—the average number of crosslinks per unit volume.
- $M_c$ , molecular weight between crosslinks—average molecular weight between crosslinks.
- $q_s$ , swell ratio—ratio of equilibrium volume in a swollen state to the initial volume.

There are two common techniques that can be used to study the swell ratio of crosslinked polymers: gravimetric measurements [40] and continuous height measurement during swelling [41].

In the gravimetric approach, specimens of the crosslinked plastic are weighted and then immersed in the solvent at a specified temperature for a specified time. After the solvent exposure the specimens are removed, dried, and reweighed. The amount of material extracted and the swell ratio is calculated from these weight measurements. The percent extracted is a measure of the amount of polymer that is soluble and not attached to the main gel phase.

In the height measurement approach, the change in specimen height is measured as a function of exposure time to the solvent at a given temperature. From the results from this experiment the swell ratio can be directly obtained from:

$$q_s = \frac{\text{Final volume}}{\text{Initial volume}}. \quad (2.15)$$

If the Flory interaction parameter  $\xi$  for the polymer-solvent system is known, then Flory's network theory [42] can be used to determine the crosslink density and the molecular weight between crosslinks can be determined from the steady-state swell ratio [41].

Swell testing experiments are useful for both quality control and for quantitative network studies. Since the amount of equilibrium swelling is sensitive to type of solvent and the test temperature, it is recommended that one of the applicable standards be followed.

## 2.6 Chemical Characterization Techniques

Most chemical characterization techniques are based on spectroscopy—the study of spectra—and how a physical property depend on frequency. Spectroscopy is used for the identification of substances, through the spectrum emitted or absorbed. Different spectroscopy techniques are classified according to the physical quantity which is measured or calculated, or the measurement process.

There are two main types of spectroscopy. The first is absorption spectroscopy which uses the range of electromagnetic spectra



in which a substance absorbs. The sample is vaporized and then light of a particular frequency is passed through the vapor. The amount of absorption can be related to the chemical composition of the material. The second type is emission spectroscopy which uses the range of electromagnetic spectra in which a substance radiates. It requires the substance to be vaporized at high temperatures by placing it in a spark gap.

Spectroscopy is often used in combination with chromatography, which is a broad range of physical methods used to separate and/or to analyze complex mixtures. The components to be separated are distributed between two phases: a stationary phase bed and a mobile phase which percolates through the stationary bed. These two techniques, and methods based on them, are discussed in more detail in the following sections.

### **2.6.1 Fourier Transform Infrared Spectroscopy**

Fourier transform infrared spectroscopy (FTIR) is an analytical technique used to identify organic (and in some cases inorganic) materials, including polymer compounds. This technique measures the absorption of various infrared light wavelengths by the material of interest. These infrared absorption bands identify specific molecular components and structures. Fourier transform spectroscopy is more sensitive and has a much shorter sampling time than conventional spectroscopic techniques.

In a conventional spectrometer, a sample is exposed to electromagnetic radiation and the responding intensity of transmitted radiation is monitored. The energy of the radiation is varied over the desired range and the response is plotted as a function of radiation energy (or frequency). At certain resonant frequencies characteristic of the specific sample, the radiation will be absorbed resulting in a series of peaks in the spectrum, which can then be used to identify the sample.

In Fourier transform spectroscopy, the sample is exposed to a single pulse of radiation. The resulting signal, called a free induction decay, contains a rapidly decaying signal of all possible

frequencies. Due to resonance in the sample, certain resonant frequencies will be dominant in the signal and by applying a Fourier transform to the response signal the frequency spectrum can be calculated. Using this approach the Fourier transform spectrometer can produce the same kind of spectrum as a conventional spectrometer, but in a much shorter time.

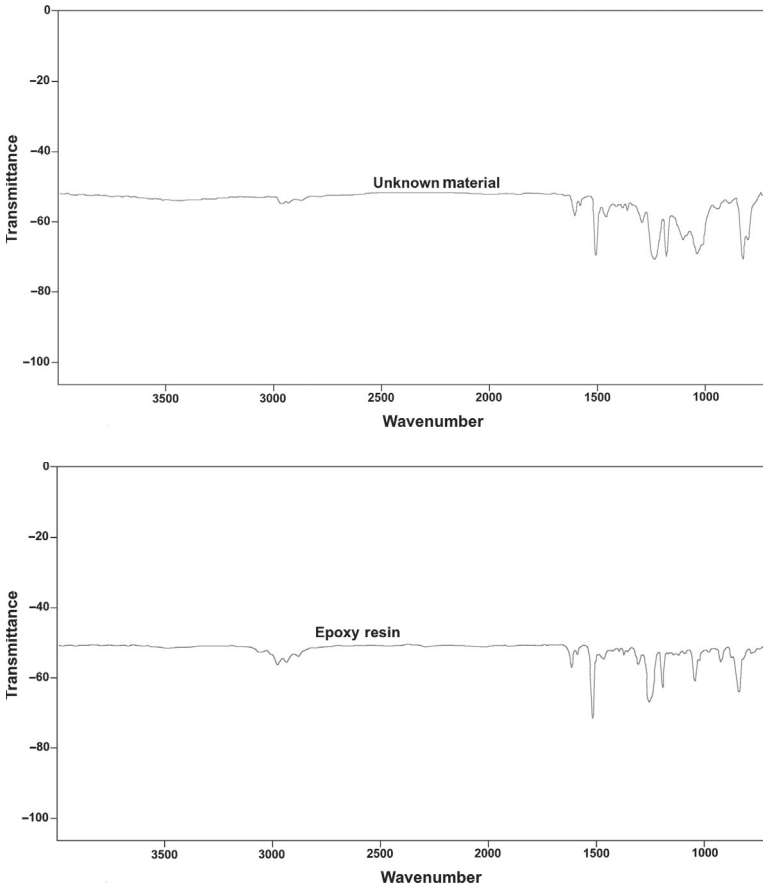
Absorption bands in the range of 4000-1500 wavenumbers are typical from common functional groups, for example,  $-\text{OH}$ ,  $\text{C}=\text{O}$ ,  $\text{N}-\text{H}$ , and  $\text{CH}_3$ . The region between 1500-400 wavenumbers is often referred to as the fingerprint region. Absorption bands in this region are generally due to intra-molecular phenomena and are highly material specific. The specificity of these bands allow computerized data searches to be performed against reference libraries to identify a material. FTIR can be used to identify the chemical structure of almost any polymer and is perhaps the most powerful tool for identifying types of chemical bonds (functional groups).

An exemplar FTIR spectrum is shown in [Figure 2.62](#). This figure shows both the measured spectrum and the best library spectrum that was found by the FTIR software to match the experimentally determined results.

## 2.6.2 Energy Dispersive Spectroscopy

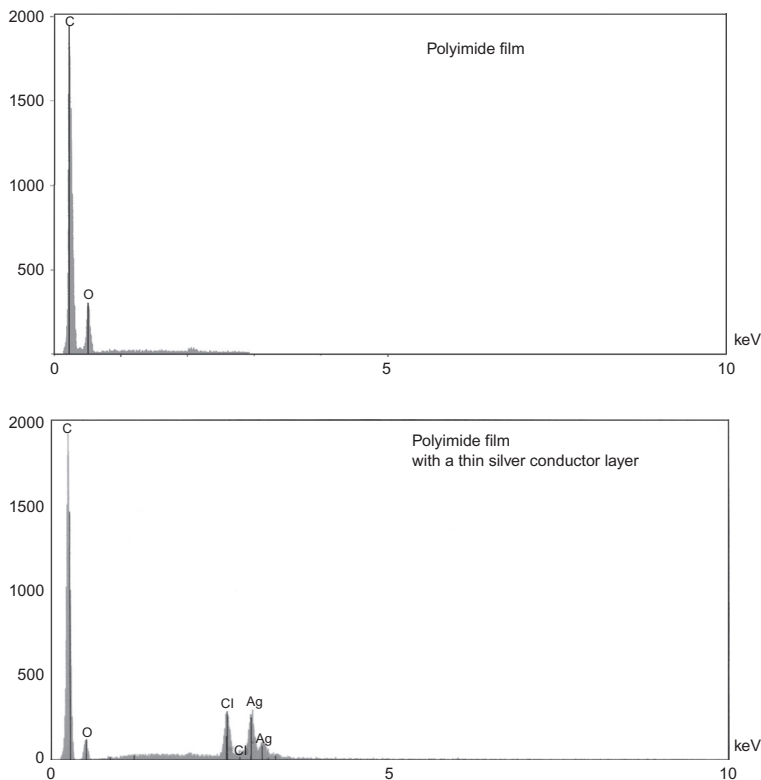
Energy dispersive X-ray spectroscopy (EDS) is a chemical microanalysis technique that, as was discussed in [Section 2.4.2](#), is typically performed in conjunction with an SEM. The EDS technique utilizes X-rays that are emitted from the sample during bombardment by the electron beam to characterize the elemental composition of the analyzed volume on a micro- or nano-scale.

In an SEM, an electron beam is scanned across the sample surface and generates X-ray fluorescence from the atoms in its path. The energies of the X-ray photons are characteristic of the element which produced it. The EDS X-ray detector measures the number of emitted X-rays versus their energy. The energy of the X-ray is characteristic of the chemical element from which the X-ray was emitted.



**Figure 2.62** Exemplar FTIR spectrum from an unknown material, and the best matching library spectrum.

By determining the energies of the X-rays emitted from the area being excited by the electron beam, the elements present in the sample can be determined. This mode of operation is called qualitative analysis since only the types of elements in the sample are determined. The rate of detection of these characteristic X-rays can also be used to measure the amounts of elements present. This mode is called quantitative analysis. If the electron beam is swept over an area of the sample, then the EDS systems can also acquire X-ray maps showing spatial variation of elements in the sample.



**Figure 2.63** (a) EDS spectrum of a pure polyimide film; (b) EDS spectrum of polyimide film with a thin silver conductor layer.

An exemplar EDS spectrum is shown in [Figure 2.63](#). This figure shows the spectrum for a polyimide film that is covered with a thin silver conductor layer. As indicated in the figure the EDS system only provides information about the chemical elements present in the sample, and do not specify the molecular structure.

Typical applications of EDS are in materials research, quality control, failure analysis, and forensic science.

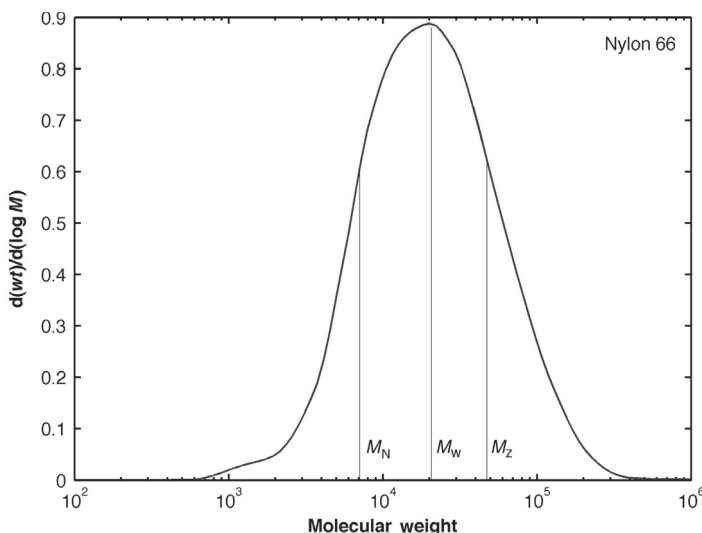
### 2.6.3 Size-Exclusion Chromatography

Size-exclusion chromatography (SEC), also known as gel permeation chromatography (GPC), is a chromatographic method in

which molecules are separated based on their size. This method is most widely used in the analysis of polymer molecular weights (or molar mass).

In SEC, a column typically made of steel with a diameter of 10 mm and a length of 500-1000 mm, is packed with a porous material (typically silica or crosslinked polystyrene) and solvent is forced through the column at rates typically 1 ml/min and pressures of 50-200 bar. A sample is dissolved in the same solvent that is running through the column and is then introduced into the solvent stream going through the column. A detector monitors the concentration of sample exiting the end of the column. Inside the column, molecules are separated based on their hydrodynamic volume (the volume the molecule occupies in a dilute solution). For polymers this can vary greatly with the particular solvent and the temperature. By studying the properties of polymers in particular solvents and by calibrating each column setup with samples of known molecular weight, it is possible to get a relative distribution of molecular weights for a given polymer sample. Using this data, it is possible to calculate number-average molecular weight, weight-average molecular weight, polydispersity, as well as higher order molecular weights to within a useful level of accuracy.

Inside the column, molecules are separated by whether or not they can fit within the pore size of the packing material. When columns are created they are packed with porous beads with a specific pore size so that they are most accurate at separating molecules with sizes similar to the pore size. As a molecule flows through the column it passes by a number of these porous beads. If the molecule can fit inside the pore then it is drawn in by the force of diffusion. There it stays a short while and then moves on. If a molecule cannot fit into a pore then it continues following the solvent flow. For this reason, in an SEC column, molecules with larger size will reach the end of the column before molecules with smaller size. The effective range of the column is determined by the pore size of the packing. Any molecules larger than all the pores in a column will be eluted together regardless of their size. Likewise, any molecules that can fit into all the pores in the packing material will elute at the same time.



**Figure 2.64** SEC results for a Nylon 66 showing the definition of  $M_N$ ,  $M_W$ , and  $M_Z$ .

It is important to remember that the only absolute measure in SEC is volume of the molecule (hydrodynamic volume), and even that measurement has certain error built into it. Interactions between the solvent, packing, and or the sample will affect the measurement as will concentration due to sample-sample interactions. Calculating the molecular weight from this molecular size introduces even more error into the system. SEC is a useful tool for determining molecular weight in polymers, but it is essential that the column and instrumentation be carefully equilibrated and properly calibrated for the results to be trusted.

The molecular weight of a polymer is typically continuously distributed over a certain range, see [Figure 2.64](#). Three different scalar quantities are commonly used to describe the molecular weight distribution:

1. *The number average* (arithmetic mean) molecular weight,  $M_N$ , is defined by

$$M_N = \sum_i \left\{ \frac{N_i}{\sum_j N_j} \right\} M_i = \frac{\sum_i N_i M_i}{\sum_j N_j}. \quad (2.16)$$

The number average is sensitive to small molecules.

2. *The weight average* molecular weight,  $M_W$ , is defined by

$$M_W = \sum_i \left\{ \frac{N_i M_i}{\sum_j N_j M_j} \right\} M_i = \frac{\sum_i N_i M_i^2}{\sum_j N_j M_j} = \frac{\sum_i w_i M_i}{\sum_j w_j}, \quad (2.17)$$

where  $w_i$  is the weight fraction of molecule  $i$ . The weight average molecular weight is more sensitive to heavy molecules.

3. *The Z-averaged* molecular weight,  $M_Z$ , is defined by

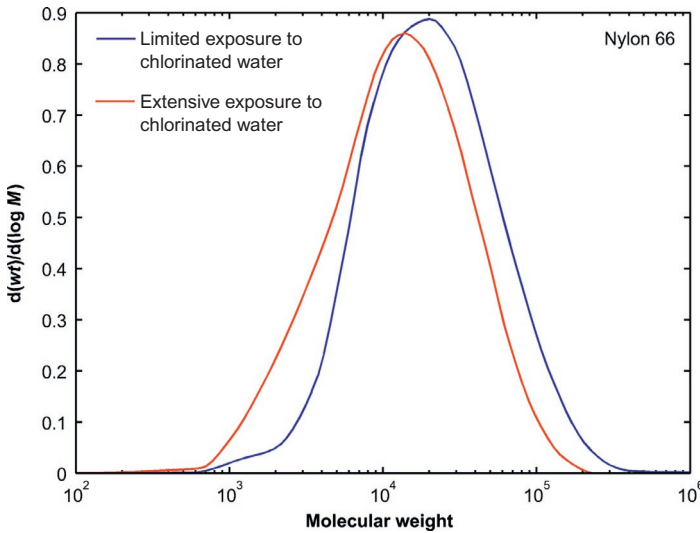
$$M_Z = \frac{\sum_i N_i M_i^3}{\sum_j N_j M_j^2}. \quad (2.18)$$

It is important to realize that the weight-averaged molecular weight is larger than or equal to the number average molecular weight:  $M_W \geq M_N$ . The ratio of the weight-average and number-average molecular weight is referred to as the polydispersity index:

$$\text{PDI} = \frac{M_W}{M_N}, \quad (2.19)$$

and is a measure of how widely distributed the molecular weight distribution is.

The mechanical properties of a polymeric material is often strongly dependent on the molecular weight of the material. [Figure 2.65](#) shows the results from an SEC investigation of two Nylon 66 specimens. One of the specimens had been exposed to chlorinated water for an extended amount of time. This specimen was heavily discolored and contained a large amount of surface cracks. The other specimen had only been exposed to chlorinated water for a short amount of time. As is shown in [Figure 2.65](#), the molecular weight of the Nylon 66 material is decreasing with



**Figure 2.65** SEC results for two Nylon 66 samples exposed to different amounts of chlorinated water.

exposure time to chlorinated water and the polydispersity of the material is also increased.

## 2.6.4 Thermogravimetric Analysis

Thermogravimetric analysis (TGA) is a thermal analysis technique that can measure changes in the weight (mass) of a sample as a function of temperature and time. TGA is commonly used to determine polymer degradation temperatures, residual solvent levels, absorbed moisture content, and the amount of inorganic (noncombustible) filler in polymer or composite material compositions.

In slightly simplified terms, in TGA a sample is placed into a tared TGA sample pan which is attached to a sensitive microbalance device. The sample holder portion is subsequently placed into a high temperature furnace. The balance device measures the initial sample weight at room temperature and changes in sample weight as heat is applied to the sample. TGA tests can be run either in heating mode at a controlled heating rate, or isothermally.



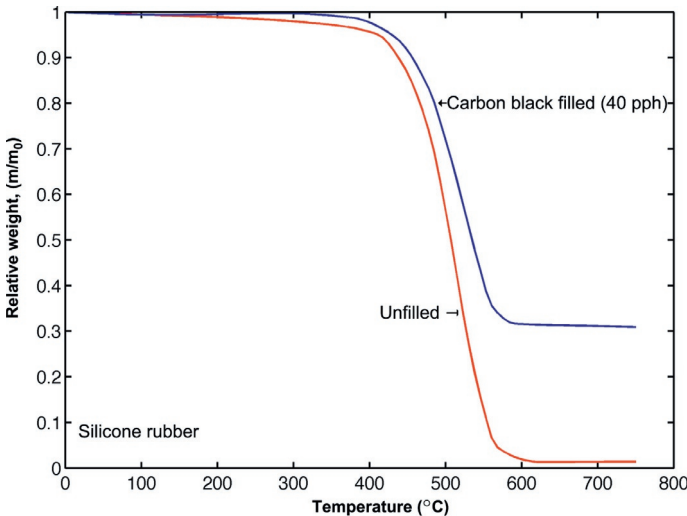
Typical weight loss profiles can then be analyzed for the amount or percent of weight loss at any given temperature, the amount or percent of noncombusted residue at some final temperature, and the temperatures of various sample degradation processes.

Various atmospheres can be used to investigate sample reactions. Furthermore, TGA can be coupled with infrared spectroscopy (TGA/FTIR) to identify the evolved gases.

Typical applications of TGA include:

- weight loss/gain;
- drying rate;
- reactivity with atmospheres;
- oxidative degradation;
- reaction kinetics;
- volatilization analysis;
- compound composition; and
- stabilizer effectiveness.

As an example of the use of TGA is shown in [Figure 2.66](#), two TGA scans of silicone rubber. The first scan is for an unfilled silicone rubber and the second scan is for a silicone rubber with



**Figure 2.66** TGA scans of two silicon rubbers.

40 pph carbon black. The figure shows that the highly filled rubber start to degrade at a higher temperature than the unfilled rubber.

### 2.6.5 Raman Spectroscopy

Raman spectroscopy is a spectroscopic technique used to study vibrational, rotational, and other low-frequency modes in a material system. It relies on inelastic scattering, or Raman scattering of monochromatic light, usually from a laser in the visible, near infrared, or near ultraviolet range. Phonons or other excitations in the system are absorbed or emitted by the laser light, resulting in the energy of the laser photons being shifted up or down. The shift in energy gives information about the phonon modes in the system. Infrared spectroscopy yields similar, but complementary information.

When light is scattered from a molecule, most photons are elastically scattered. The scattered photons have the same energy (frequency) and, therefore, wavelength, as the incident photons. However, a small fraction of light (approximately 1 in  $10^7$  photons) is scattered at optical frequencies different from, and usually lower than, the frequency of the incident photons. The process leading to this inelastic scatter is called the Raman effect. Raman scattering can occur with a change in vibrational, rotational, or electronic energy of a molecule. Polymer chemists are concerned primarily with the vibrational Raman effect. In the following, we will use the term Raman effect to mean vibrational Raman effect only.

The difference in energy between the incident photon and the Raman scattered photon is equal to the energy of a vibration of the scattering molecule. A plot of intensity of scattered light versus energy difference is called a Raman spectrum.

In Raman spectroscopy, a laser beam is used to irradiate a spot on the sample under investigation. The scattered radiation produced by the Raman effect contains information about the energies of molecular vibrations and rotations, and these depend on the particular atoms or ions that comprise the molecule, the chemical bonds that connect them, the symmetry of their

molecule structure, and the physico-chemical environment where they reside.

Typically, a sample is illuminated with a laser beam. Light from the illuminated spot is collected with a lens and sent through a monochromator. Wavelengths close to the laser line (due to elastic Rayleigh scattering) are filtered out and those in a certain spectral window away from the laser line are dispersed onto a detector.

Spontaneous Raman scattering is typically very weak, and as a result the main difficulty of Raman spectroscopy is separating the weak inelastically scattered light from the intense Rayleigh scattered laser light. Raman spectrometers typically use holographic diffraction gratings and multiple dispersion stages to achieve a high degree of laser rejection. A photon-counting photomultiplier tube (PMT) or, more commonly, a CCD camera is used to detect the Raman scattered light.

Raman spectroscopy is commonly used for polymers, since vibrational information is very specific for the chemical bonds in molecules. It therefore provides a fingerprint by which the molecule can be identified.

## 2.7 Exercises

1. Does a semicrystalline polymer have a glass transition temperature? Why/why not?
2. How does the Young's modulus change with temperature for an amorphous polymer? For a semi-crystalline polymer?
3. What is Mullins effect? What type of polymer exhibit the strongest Mullins effect?
4. What are the most common ways to reduce the influence of friction in a uniaxial compression test?
5. Does all polymers neck when pulled in tension? Why/why not?
6. What are some common mechanical tests that are performed on polymers to determine the mechanical behavior?

7. What tests would you recommend in order to get experimental data for calibrating a viscoplastic material model?
8. What is the difference between the Charpy and Izod impact tests?
9. What are the strengths and weaknesses of the DMA testing?
10. What are the two most common hardness tests that are performed on polymers?
11. What cannot SHPB testing be performed on soft polymers?
12. When is it necessary to perform bulk modulus testing on a polymer?
13. What are some of the most common experimental techniques for determining the failure behavior of a polymer?
14. What is the difference between verification and validation?
15. What experimental tests would you perform in order to validate an already calibrated material model?
16. List a few common techniques that can be used to study the surface characteristics of a polymer sample.
17. How does the DSC technique work? What data can be obtained from a DSC experiment?
18. Is SEM or TEM more commonly used for polymer mechanics analysis? Why?
19. What is the Bragg law and How is used in X-ray diffraction experiments?
20. Explain how birefringence can be used to aid the development of a hyperelastic material model for a rubber.
21. What are two techniques that can be used to determine the molecular weight of a polymer sample?
22. What are the definitions of number-average, weight-average, and z-average molecular weight?
23. What information can be obtained from an FTIR experiment?
24. Explain a situation when it might be useful to perform a TGA experiment.

## References

- [1] ASTM D575, Standard Test Methods for Rubber Properties in Compression, 2001.
- [2] M.C. Boyce, K. Kear, S. Socrate, K. Shaw, Deformation of thermoplastic vulcanizates, *J. Mech. Phys. Solids*, 49 (2001) 1073-1098.
- [3] J.S. Bergström, M.C. Boyce, Constitutive modeling of the large strain time-dependent behavior of elastomers, *J. Mech. Phys. Solids*, 46 (1998) 931-954.
- [4] ASTM D2990, Standard Test Methods for Tensile, Compressive, and Flexural Creep and Creep-Rupture of Plastics, 2001.
- [5] ASTM D412, Standard Test Methods for Vulcanized Rubber and Thermoplastic Elastomers—Tension, 2002.
- [6] ASTM D638, Standard Test Method for Tensile Properties of Plastics, 2003.
- [7] ASTM D256, Standard Test Methods for Determining the Izod Pendulum Impact Resistance of Plastics, 2004.
- [8] ASTM D6110, Standard Test Method for Determining the Charpy Impact Resistance of Notched Specimens of Plastics, 2004.
- [9] ISO 179, Plastics—Determination of Charpy Impact Properties—Part 1: Non-Instrumented Impact Test, 2000.
- [10] ISO 180, Plastics—Determination of Izod Impact Strength, 2000.
- [11] ASTM D5420, Standard Test Method for Impact Resistance of Flat, Rigid Plastic Specimens by Means of a Striker Impacted by a Falling Weight (Gardner Impact), 2004.
- [12] ASTM D1709, Standard Test Methods for Impact Resistance of Plastic Films by the Free-Falling Dart Method, 2004.
- [13] Veryst Engineering, Instrumented Impact Testing. Available from: <http://www.veryst.com/what-we-offer/material-testing-modeling>.
- [14] J.D. Ferry, *Viscoelastic Properties of Polymers*, John Wiley & Sons, Inc., New York, 1980.
- [15] K. Menard, *Dynamic Mechanical Analysis: A Practical Introduction*, CRC Press, Boca Raton, FL, 1999.
- [16] ASTM D4065, Standard Practice for Plastics: Dynamic Mechanical Properties: Determination and Report of Procedures, 2001.
- [17] ISO 6721, Plastics—Determination of Dynamic Mechanical Properties, 1994.
- [18] ASTM D785, Standard Test Method for Rockwell Hardness of Plastics and Electrical Insulating Materials, 2003.

- [19] ASTM D2240, Standard Test Method for Rubber Property—Durometer Hardness, 2004.
- [20] ASTM D2583, Standard Test Method for Indentation Hardness of Rigid Plastics by Means of a Barcol Impressor, 2001.
- [21] W.W. Chen, B. Song, Split Hopkinson (Kolsky) Bar: Design, Testing, and Applications, Springer, New York, 2010.
- [22] D. Walsh, P. Zoller, Standard Pressure Volume Temperature Data for Polymers, CRC Press, Boca Raton, FL, 1995.
- [23] W. Young, R. Budynas, A. Sadegh, Roark's Formulas for Stress and Strain, McGraw-Hill Professional, New York, 2011.
- [24] ASTM D0624, Standard Test Method for Tear Strength of Conventional Vulcanized Rubber and Thermoplastic Elastomers, 2012.
- [25] ASTM D6068, Standard Test Method for Determining J-R Curves of Plastic Materials, 2010.
- [26] ASTM F2183, Standard Test Method for Small Punch Testing of Ultra-High Molecular Weight Polyethylene Used in Surgical Implants, 2002.
- [27] J.S. Bergström, C.M. Rimnac, S.M. Kurtz, Prediction of multiaxial mechanical behavior for conventional and highly crosslinked UHMWPE using a hybrid constitutive model, *Biomaterials*, 24 (2003) 1365-1380.
- [28] J.S. Bergström, C.M. Rimnac, S.M. Kurtz, Molecular chain stretch is a multiaxial failure criterion for conventional and highly crosslinked UHMWPE, *J. Orthop. Res.*, 23 (2003) 367-375.
- [29] J.S. Bergström, C.M. Rimnac, S.M. Kurtz, An augmented hybrid constitutive model for simulation of unloading and cyclic loading behavior of conventional and highly crosslinked UHMWPE, *Biomaterials*, 25 (2004) 2171-2178.
- [30] ASTM D5379, Standard Test Method for Shear Properties of Composite Materials by the V-Notch Beam Method, 2012.
- [31] R.J. Cherry, *New Techniques of Optical Microscopy and Microspectroscopy*, CRC Press, Boca Raton, FL, 1991.
- [32] B. Herman, J.J. Lemasters, *Optical Microscopy: Emerging Methods and Applications*, Academic Press, San Diego, CA, 1992.
- [33] G. Binnig, C.F. Quate, Ch. Geber, *Atomic Force Microscope*, *Phys. Rev. Letters* (1986) 930.
- [34] O.A. Hasan, M.C. Boyce, Energy storage during inelastic deformation of glassy polymers, *Polymer*, 34 (24) (1993) 5085-5092.
- [35] J.S. Bergström, Large strain time-dependent behavior of elastomeric materials, Ph.D. thesis, MIT 1999.

- [36] R.A. Pethrick, C. Viney, X-ray diffraction from crystalline polymers, in: *Experimental Methods in Polymer Characterisation*, John Wiley & Sons, Sussex, 2003.
- [37] R.W. Richards, Wide angle X-ray scattering, in: *Scattering Methods in Polymer Science*, Ellis Horwood, London, 1995
- [38] H.P. Klug, L.E. Alexander, *X-Ray Diffraction Procedures: For Polycrystalline and Amorphous Materials*, second ed., Wiley-Interscience, New York, 1974.
- [39] G.H. Meeten, *Optical Properties of Polymers*, Elsevier, New York, 1986.
- [40] ASTM D2765, Standard Test Method for Determination of Gel Content and Swell Ratio of Crosslinked Ethylene Plastics, 2001.
- [41] ASTM F2214, Standard Test Method for In Situ Determination of Network Parameters of Crosslinked Ultra High Molecular Weight Polyethylene (UHMWPE), 2002.
- [42] P.J. Flory, *Principles of Polymer Chemistry*, Cornell University Press, Ithaca, NY, 1953.

# 3 Finite Element Analysis as an Engineering Tool

---

## Chapter Outline

3.1 Introduction	115
3.1.1 Required Inputs for FEA	117
3.2 Types of FEA	119
3.3 Review of Modeling Techniques	120
3.3.1 Deformation Modeling	120
3.3.2 Failure Modeling	125
3.4 Exercises	130
References	130

## 3.1 Introduction

Polymer mechanics is the study of how the mechanical behavior of polymers depend on external load environments. It is a broad subject that provides tools to engineers and scientist interested in understanding the behavior of polymer components and how their performance can be predicted and optimized. This chapter provides a broad overview of different approaches and techniques that can be used when studying polymer mechanics problems, and as indicated in the title, the focus is on the use of the finite element (FE) method.

Polymeric materials have during the last few decades been gradually transformed from being conceived as low-budget, low-technology materials—exemplified by expression such as “it is just a plastic”—to highly reliable and advanced materials with excellent mechanical properties. One of the driving forces for this transformation has been the development of a better understanding of polymer behavior and how polymer components should be designed. In other word, experimental and theoretical polymer



mechanics have paved the way for new and improved uses of polymers.

Designing components and products using polymers require specific knowledge about their behavior. This is important since their mechanical behavior is very different than traditional engineering metals, see [Table 3.1](#). One of the main challenges of polymer mechanics is that it is a multi-disciplinary subject that has strong ties to mechanical engineering, material science, bioengineering, and chemical engineering—all of which are core

**Table 3.1 Overview of Differences in Properties of Polymers and Metals**

<b>Polymers</b>	<b>Metals</b>
Exhibit nonlinear viscoelastic behavior when deformed at room temperature	Exhibit elastic-plastic behavior when deformed at room temperature
Often contain both amorphous and semi-crystalline domains	Crystalline microstructure <sup>a</sup>
Visco-plastic deformation driven by macromolecular reorganization	Plastic deformation driven by dislocation motion and twinning
At temperatures close to room temperature, the material response is very strongly dependent on deformation rate and temperature	Rather weak dependence on deformation rate and temperature at temperatures close to room temperatures
The maximum usage temperature is never more than 300 °C and is often less than 100 °C	Can be used in very high temperature applications (higher than 800 °C)
Do not corrode	Corrodes in an aggressive environment
Can dissolve in an aggressive environment	Generally impervious to solvents

<sup>a</sup>By using specialized manufacturing methods it is possible to create metals that are amorphous.

subjects of most educational institutions. Polymer mechanics, as a subject, draws heavily on these disciplines, although the emphasis of polymer mechanics in this context is perhaps most strongly connected with mechanical engineering.

There are different analytical techniques that can be used when studying the mechanical behavior of polymers. The traditional approach, and until about 30 years ago the only approach, is to use closed-form analysis to study deformations and stresses and how these influence ultimate properties such as fatigue and fracture.

The development of the FE method and computers have revolutionized not only polymer mechanics, but all fields of component design and analysis. The reason for this is that the FE method enables direct analysis of complex geometries with relatively little effort, problems that cannot even be solved using traditional closed-form analytical techniques. A summary of different classes of analysis techniques, and their strengths and weaknesses is presented in [Table 3.2](#).

As indicated in this table, closed-form analysis, when applicable, provides the most detailed information. For example, using this technique it is possible to determine not only the stress state for a given imposed deformation state but also the mathematical dependence of how geometry and load history directly influence the stress state.

FE tools have reached a high level of maturity and are widely used in both academia and industry. The easy access to commercial FE codes has created both great possibilities to solve advanced problems, and to some extent reduced the need for costly experimental tests. However, this computational modeling approach also presents serious challenges since the FE programs, albeit being easy to use, can provide inaccurate and misleading results if not used properly. One of the overall aims of this text is to assist the creation and selection of FE models and analysis techniques for polymer problems.

### 3.1.1 Required Inputs for FEA

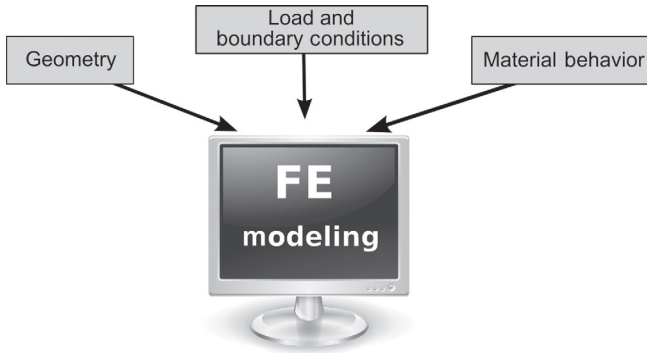
To perform an FE simulation, or indeed any stress analysis calculation, there are three different types of inputs that need to

**Table 3.2 Summary of Different Analysis Techniques, Their Strengths and Limitations, and How They Can Be Applied to Polymer Mechanics Problems**

Analysis Method	Strengths	Limitations
Closed-form calculation using a simple material model (e.g., linear elasticity)	+ Computationally efficient + Provides the mathematical relationship between deformation and stress	– Only applicable to simple geometries
FE simulation using a simple material model	+ Captures nonlinear geometric effects – Easy to calibrate material model	– Simple material models often do not give accurate results
FE simulation using an advanced material model	+ Captures nonlinear geometric effects + Can very accurately capture the material response	– More difficult to calibrate material model – More computationally costly

be specified, see [Figure 3.1](#). Note that these inputs are also needed for traditional closed-form solution methods.

The fundamental problem of polymer mechanics can be written in mathematical form as a boundary value problem (BVP), with governing equations for: compatibility, constitutive response, and equilibrium. More details of these equations are given in Chapter 4. One of the overall themes of this book is that of the three different types of input to the FE models: (1) geometry; (2) loading and boundary conditions (BC); and (3) material behavior; it is the specification of the material behavior that is typically most challenging.



**Figure 3.1** Inputs needed for FE analysis.

## 3.2 Types of FEA

The FE method can be described as a numerical tool for solving ordinary and partial differential equations over nontrivial geometric domains. In practice, finite element analysis (FEA) can be divided into two different categories: implicit and explicit simulations, see [Table 3.3](#). FEA can also be used to study eigenfrequencies and eigenmodes of deformation for a component or system.

As mentioned, one of the most difficult steps in an FE simulation is to specify the material model. A material model is here defined as a constitutive equation and a corresponding set of material parameters:

$$\boxed{\text{Constitutive model}} + \boxed{\text{Material parameters}} = \boxed{\text{Material model}}$$

FE software contains a library of different constitutive equations that can be chosen, but the material parameters are typically not provided and the selection of constitutive models that are available is typically targeted to metals. There are generally only a limited set of constitutive models that are suitable for predicting the deformation behavior of polymers. One way to get around this is to use an external user material subroutine (UMAT) to define the material behavior, see Chapter 10 for more details.

**Table 3.3 Comparison Between Implicit and Explicit FEA**

<b>Implicit Analysis</b>	<b>Explicit Analysis</b>
Solves the equilibrium equations at each time step	Solves the problem using Newton's law of motion
Good for static problems	Good for short duration dynamic problems
Is numerically stable	Is only numerically stable for small time increments
If the FE software finds a solution, that solution is likely to have small numerical errors	Often easy to find a solution, but care is needed to find a solution with small numerical errors
Contact problems are sometimes difficult to handle	Good at handling problems with contact

### 3.3 Review of Modeling Techniques

Most polymer mechanics problems can be divided into two main categories: predictions of deformation behavior and predictions of failure events. A common approach is to start with a deformation analysis to determine the magnitudes and distributions of stress and strain, and then, if needed, use this information as part of a predictive failure analysis. The following subsections provide brief examples of these modeling types. More detailed presentations of the modeling theories are given in the following chapters.

#### 3.3.1 Deformation Modeling

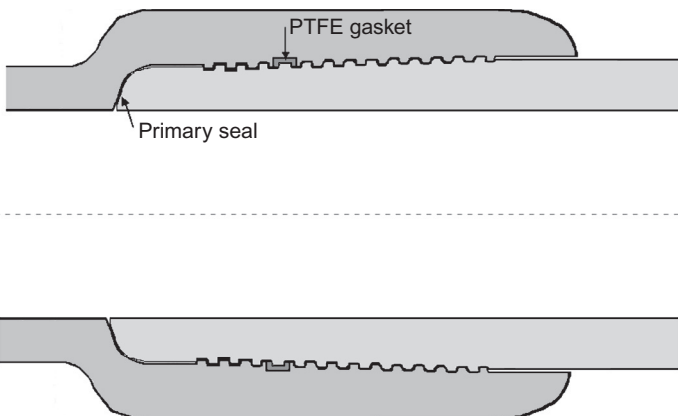
A key component of polymer mechanics analyses involves determining the deformation response as a function of applied loads. The following two examples illustrate common problems that a polymer mechanist may be exposed to.

**Example: Thermomechanical Deformations of a Threaded Connection Gasket.**

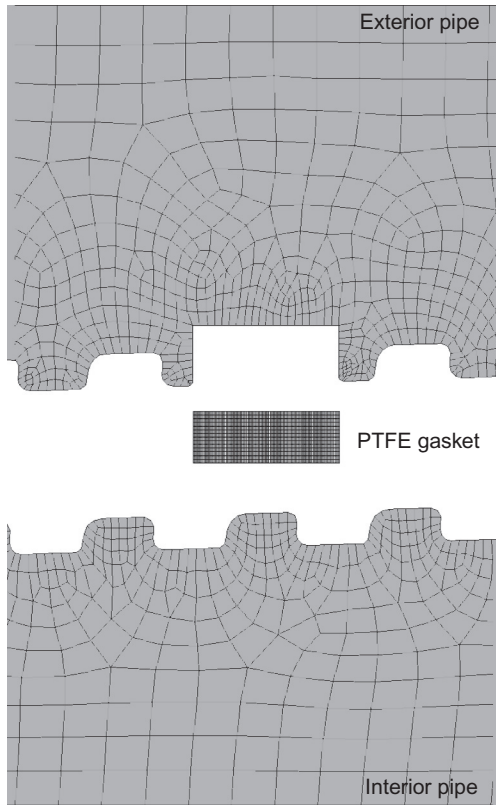
One example of a deformation model is a PTFE gasket in a threaded connection for high pressure pipelines, see the illustration in [Figure 3.2](#). The pipeline shown in this figure is used to transport gases at temperatures between 20 °C and 200 °C. The two parts of the pipeline are threaded together, and the primary seal to prevent gas leakage is created by a metal-to-metal seal. In this example, a PTFE gasket is used as a secondary seal to reduce the risk of gas leaks. The gasket used in this seal is made from PTFE filled with 10 vol% glass fibers.

The manufacturer of the pipeline wanted to reduce the number of costly experiments and enable faster design evaluations. To achieve these goals it was decided to develop an FE model of the PTFE gasket and threaded connection, see [Figure 3.3](#). In this figure, the interior pipe, the gasket, and the exterior pipe have been separated vertically to better illustrate the geometry.

The goal of the analysis was to predict the amount of residual sealing force between the PTFE gasket and the interior and exterior pipes as a function of time, temperatures, and seal design. During the assembly of the threaded connection, the gasket is exposed to very large deformations and pressures. It is



**Figure 3.2** Cross-section of a pipeline with a threaded connection containing a PTFE gasket as a secondary seal.



**Figure 3.3** Details of the FE mesh close to the PTFE gasket.

therefore necessary to use a constitutive model capable of predicting the large-strain time-dependent thermomechanical behavior of the PTFE. To accomplish this, an advanced material model specifically developed for PTFE was used (the Dual Network Fluoropolymer [DNF] model [1], see Section 8.4). The DNF model is not yet a built-in feature of commercial FE codes, but have been implemented as a UMAT in Abaqus, ANSYS, and LS-DYNA [2]. The results presented below were obtained using Abaqus/Standard.

The first step in this project was to experimentally characterize the nonlinear response of the gasket material. This testing included uniaxial monotonic and cyclic loading at different temperatures, volumetric compression experiments, and a multiaxial small punch test [3]. The second step was to calibrate the consti-

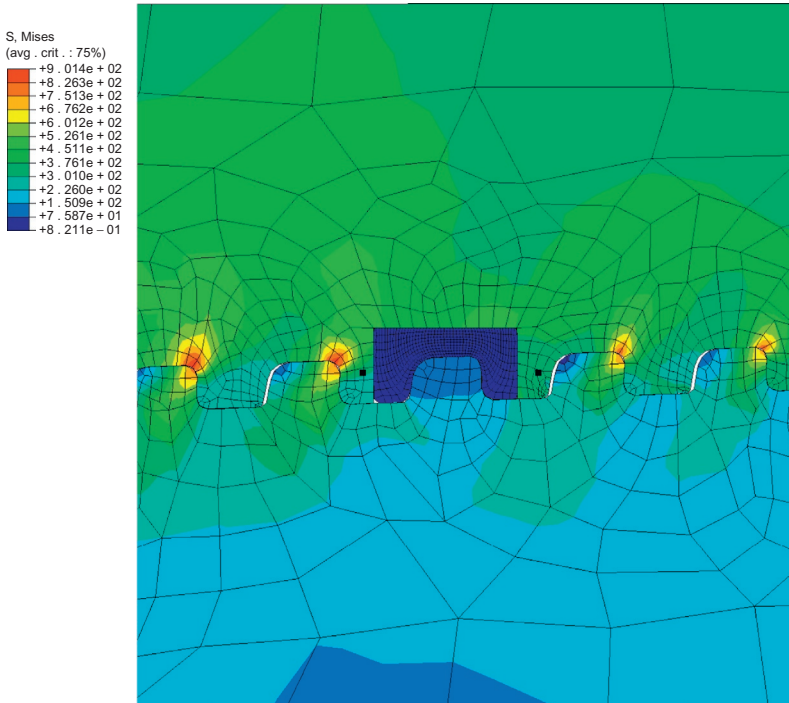
tutive model to the experimental data and to validate the material model by comparison to the small punch data (see Section 2.3.2). After the DNF model had been calibrated and validated, it could be used to simulate the deformation behavior of the PTFE gasket at the temperatures and deformation states of interest.

One example of the results that were obtained from the stress analysis are presented in Figures 3.4 and 3.5.

Figure 3.4 shows contours of Mises stress (in MPa) in the gasket region. It is clear that the stresses in the metal pipes are very high and that the PTFE gasket has been significantly deformed.

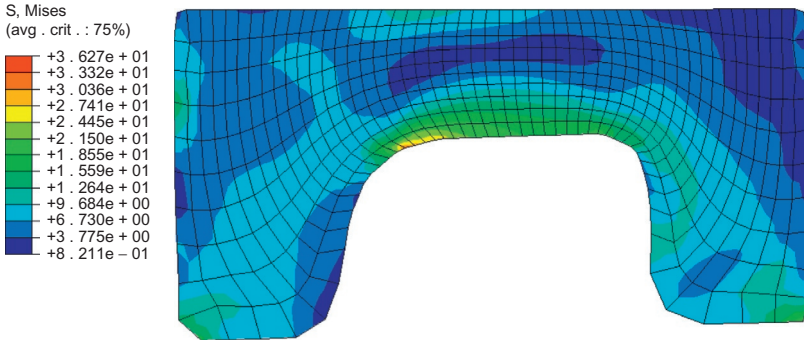
A more detailed picture of the stress state in the PTFE gasket is shown in Figure 3.5.

In summary, by calibrating an advanced material model to the PTFE gasket material it is possible to use FE simulations to



**Figure 3.4** Contours of Mises stress (in MPa) in the threaded connection and the PTFE gasket. A mesh refinement study is needed in order to make sure the mesh is refined enough.





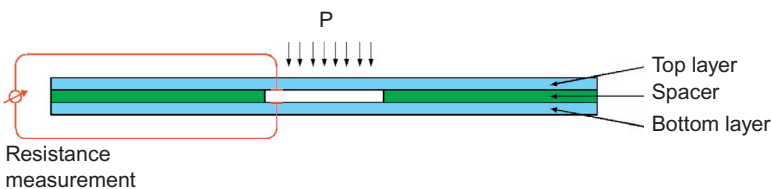
**Figure 3.5** Details of the Mises stress (in MPa) in the gasket.

directly predict the sealing capability as a function of seal design and loading conditions.

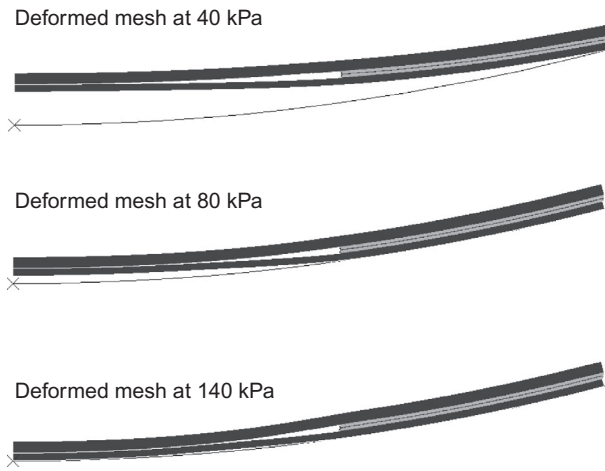
**Example: Deformation of a Flex Circuit Pressure Sensor.**

Flex circuit pressure sensors are used in a wide variety of applications, including touch displays and child seat sensors in car seats. This type of sensor typically consists of a top and bottom layer separated by a spacer layer. The actual sensing region consists of a cavity in the spacer layer, see Figure 3.6. The top and bottom layers are often made from a thermoplastic, and the spacer layer is often made from an adhesive. The bottom surface of the top layer and the top surface of the bottom layer are made conducting by adding conducting particles, such as carbon black. The applied pressure can be determined by measuring the resistance between the top and bottom layers.

This example illustrates the use of the FE method to examine the influence of sensor curvature on the sensor performance. The electrical resistance is directly related to the contact area, so the



**Figure 3.6** Schematic side view of a flex circuit pressure sensor.



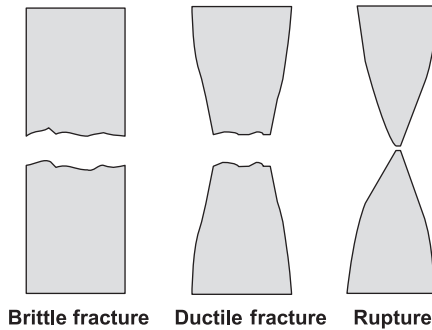
**Figure 3.7** Deformed shapes of the flex circuit pressure sensor at different applied pressures.

goal of these simulations is to determine the contact area as a function of applied pressure and curvature.

The performance of the pressure sensor will depend on the electric stability and creep characteristics of the materials in the sensor, and how the pressure is applied. In this case, the behavior of each layer of the sensor was first experimentally examined. Then appropriate constitutive models were calibrated to each material. One example of the type of simulations that were performed to study the pressure sensor design is shown in [Figure 3.7](#). Here, an axisymmetric representation of the sensor was placed on a rigid surface with a given radius of curvature. This approach enabled a direct determination of the influence of radius of curvature and applied pressure on the contact area.

### 3.3.2 Failure Modeling

The perhaps most obvious criterion when designing a part is that it should not fail in normal use during its intended lifetime. Here, *failure* is defined as material failure or fracture by the creation of new material surfaces and parts. Failure is a collective term for both fracture and rupture, where rupture is a special



**Figure 3.8** Schematic representation of different failure modes.

failure mode in which failure occurs by localized thinning to a point, see [Figure 3.8](#). Both fracture and rupture can occur either by monotonic or cyclic loading. Note, rupture is a somewhat uncommon event that is only seen in the most ductile polymers.

Failure can be caused by different mechanisms and be of different types; for example, monotonic overloading, cyclic fatigue, or wear. For polymers, failure is also often caused by a combination of mechanical loads and material degradation due to environmental exposure. This combination of mechanical loads and material degradation is commonly referred to as environmental stress cracking (ESC).

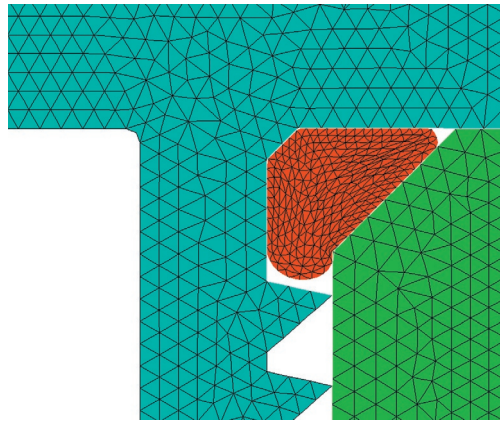
The following examples illustrate the analysis of common failure scenarios.

#### **Example: Failure of a Water Filter.**

Failure of polymer components can often be attributed to one or more of the following reasons:

- material selection;
- part design;
- manufacturing processes; and
- service/environmental conditions.

This example illustrates a failure that was caused by a combination of material selection and part design. The failed component in this case is a water purification system. The water filter consisted



**Figure 3.9** Details of a two-dimensional axisymmetric FE model of the tank (green), connector head (blue), and o-ring (red).

of a water inlet, a tank filled with the purification agent, and a water outlet. In this case, the water filter frequently failed by the connector head breaking off from the tank. The bottom threaded region of the connector head was made of acrylonitrile butadiene styrene (ABS). By examining the fracture surfaces of failed parts it was clear that the fracture initiation site was the top thread of the connector head.

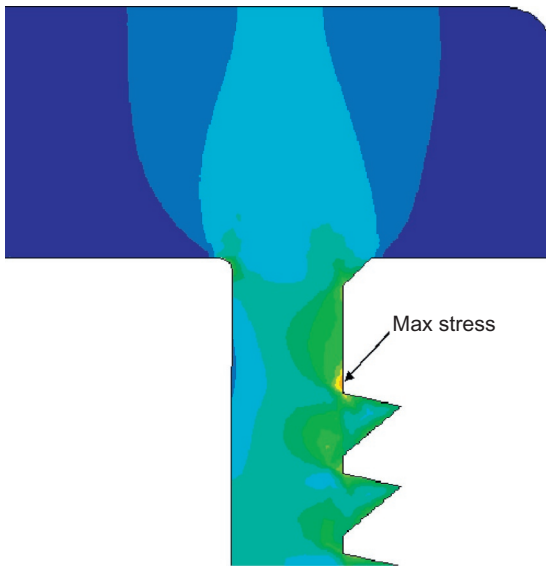
To study the failure mode a two-dimensional axisymmetric FE model was created to assess the stress level in the head, see [Figure 3.9](#). This figure shows that in its assembled state, an interior o-ring that is located between the connector head and the tank becomes severely distorted.

An example of the stress-contours in the connector head in the assembled state is shown in [Figure 3.10](#).

The figure shows that there is a stress concentration near the threads of the plastic head. The maximum Mises stress was 65% of the yield stress of the polymer. The conclusion of the stress analysis is that the material was not defective, either the design needs to be modified to reduce the stress or a different material should be used.

### **Example: Failure of Corrugated Hose.**

A corrugated PTFE hose was used to transport liquid  $\text{SO}_3$  (sulfur trioxide) at a chemical plant. The hose consisted of an exterior



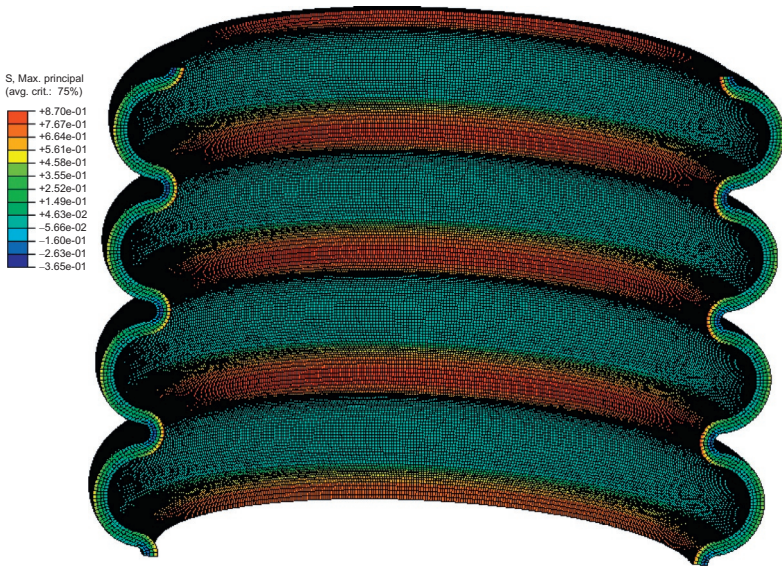
**Figure 3.10** Contours of Mises stress in the connector head in the assembled state.

layer of braided stainless steel with a corrugated PTFE liner inside. Over time, a small amount of  $\text{SO}_3$  diffused through the PTFE liner and reached the load-carrying stainless steel braid which gradually corroded due to its interaction with the  $\text{SO}_3$  and exterior water.

Eventually, all stainless steel corroded away in one section, and the axial load and internal pressure were applied directly to the PTFE liner. In the section where the steel had been corroded away, the PTFE liner could not take the applied loads and failed, causing a release of  $\text{SO}_3$ , and environmental problems.

An FEA was performed to better understand the failure events and what the internal pressure was at the time of the failure. Two examples illustrating the results from this analysis are presented in [Figures 3.11](#) and [3.12](#).

[Figure 3.11](#) shows the deformed shape of a cross-section of the hose, and contours of the maximum principal stress. In this simulation, the applied temperature was  $100^\circ\text{C}$ , the internal pressure was 120 kPa, and the results are shown 1 s after the application of the load.



**Figure 3.11** Contours of maximum principal stress (in MPa): temperature  $T = 100\text{ }^\circ\text{C}$ ,  $P = 120\text{ kPa}$ ,  $F = 0\text{ N}$ , loading time  $t = 1\text{ s}$ .



**Figure 3.12** Contours of maximum principal stress (in MPa): temperature  $T = 100\text{ }^\circ\text{C}$ ,  $P = 120\text{ kPa}$ ,  $F = 0\text{ N}$ , loading time  $t = 60\text{ s}$ .

Figure 3.12 shows the results from the same load scenario, except that the results in this case are for a time of 60 s after the application of the load. It is clear that the liner undergoes large viscoplastic deformations at this temperature and applied pressure. Also, note that the location of the maximum stress changes as a function of time from the internal surface to the exterior surface.

### 3.4 Exercises

1. What are some of the key differences between polymers and metals?
2. What are the three required input types for an FE simulation?
3. What are some of the key differences between explicit and implicit FE simulations? When is it appropriate to perform an implicit simulation, and when is it appropriate to perform an explicit simulation?
4. Describe the difference between a material model and a constitutive model.
5. What are the four most common reasons a polymer component might fail?

### References

- [1] J.S. Bergström, C.M. Rimnac, S.M. Kurtz, An augmented hybrid constitutive model for simulation of unloading and cyclic loading behavior of conventional and highly crosslinked UHMWPE, *Biomaterials*, 25 (2004) 2171-2178.
- [2] PolyUMod, <http://PolyUMod.com/>.
- [3] ASTM F2183, Standard test method for small punch testing of ultra-high molecular weight polyethylene used in surgical implants, 2002.

# 4 Continuum Mechanics Foundations

---

## Chapter Outline

4.1	Introduction	132
4.2	Classical Definitions of Stress and Strain	133
4.2.1	Uniaxial Loading	133
4.2.2	Multiaxial Loading	135
4.3	Large Strain Kinematics	137
4.4	Vector and Tensor Algebra	141
4.4.1	Vector Operations	141
4.4.2	The Dyadic Product	143
4.4.3	Tensor Operations	144
4.4.4	Derivatives of Scalar, Vector, and Tensor Fields	147
4.4.5	Coordinate Transformations	149
4.4.6	Invariants	150
4.5	Deformation Gradient	150
4.5.1	Eigenvalue and Spectral Decompositions	154
4.6	Strain, Stretch, and Rotation	157
4.7	Rates of Deformation	164
4.8	Stress Tensors	165
4.8.1	Stress Invariants	169
4.9	Balance Laws and Field Equations	171
4.9.1	Conservation of Mass	173
4.9.2	Balance of Linear Momentum	175
4.9.3	Balance of Angular Momentum	178
4.9.4	First Law of Thermodynamics	180
4.9.5	Second Law of Thermodynamics	183
4.10	Energy Balance and Stress Power	184
4.11	Constitutive Equations	187
4.11.1	Constitutive Equations for a Thermoelastic Material	189



4.12 Observer Transformation	194
4.12.1 Objective Rates	198
4.13 Material Symmetry	198
4.14 List of Symbols	199
4.15 Exercises	202
References	206

## 4.1 Introduction

Continuum mechanics is a subject that unifies solid mechanics, fluid mechanics, thermodynamics, and heat transfer, all of which are core subjects of mechanical engineering. The approach taken in this text is to use continuum mechanics as a universal tool in which to formulate the polymer mechanics theories that will be presented in the following chapters. In some undergraduate textbooks, the polymer mechanics frameworks are often presented using a traditional small-strain solid mechanics approach in an effort to make the theory less abstract. Here, a different approach is taken. Polymer components, due to their mechanical characteristics, are often both able and designed to undergo large deformations, and it is therefore important for the polymer mechanist to know that it is not sufficient to simply talk about *stress* and *strain* when working with polymers, but that there are in fact different types of stress and strain that can be used, and hence it is important to specify and correctly use the different types of stress and strain that are available.

The most logical approach for presenting these concepts is to use a continuum mechanics approach, specifically tailored to solid polymers. The approach taken here has been to reduce some of the abstraction, while at the same time keep the direct tensor notation due to its simplicity. The content presented in this chapter is meant to be self-sufficient, and has been given a somewhat condensed representation in order to fit into one chapter. There are many excellent references that present a more comprehensive treatment of the subject [1–8]. The chapter starts with a discussion about the small-strain definitions of stress and strain followed by a brief review of tensor algebra, and then introduces the different topics in a logical order.

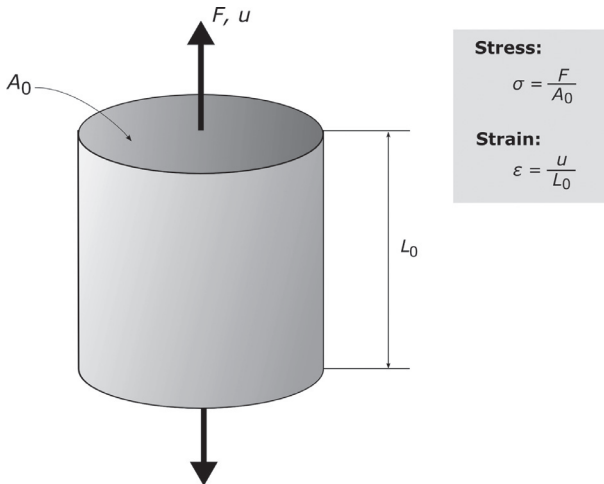
## 4.2 Classical Definitions of Stress and Strain

The classical approach to define stress and strain is based on the assumption of small deformations. As will be discussed below, this means that the functional dependence of the displacement of each material point<sup>1</sup> is assumed to be linear. The implications of this linearized theory for uniaxial loading is presented in [Section 4.2.1](#), and for multiaxial loading in [Section 4.2.2](#).

### 4.2.1 Uniaxial Loading

The classical approach to define stress and strain in uniaxial deformation is given in [Figure 4.1](#).

This figure shows a cylinder with an initial length  $L_0$  and initial cross-sectional area  $A_0$  that is uniaxially loaded with a force  $F$ . The bottom of the cylinder is held fixed, and the top is displaced a distance  $u$  as a result of the applied force  $F$ .



**Figure 4.1** Classical definition of stress and strain for uniaxial loading with small deformations.

<sup>1</sup>A material point is a location that is fixed in a body and moves with the material during deformations.

In general, the length of the cylinder  $L$  can be a nonlinear function of the applied force:  $L = \hat{L}(F)$ . This functional relationship can be represented as a Taylor series expansion:

$$L = L_0 + \frac{d\hat{L}(F)}{dF}F + \dots \quad (4.1)$$

In the small-strain theory, only the linear term (first derivative of  $\hat{L}(F)$ ) is included. By defining  $u = L - L_0$  and  $k = dF/dL$ , the force-displacement relationship becomes the classical linear elastic spring equation:

$$F = ku. \quad (4.2)$$

The mechanical stress is defined as the force intensity, that is, stress = force/area. One important question is what area should be used in the calculation of the stress. When a tensile force is applied on the cylindrical specimen it will not only get longer, but it will also typically<sup>2</sup> reduce its cross-sectional area. Let us start by defining the radius of the cylinder at a given applied force to be  $r$ , the initial (undeformed) radius to be  $r_0$ , and the change in radius to be  $\Delta r$ , then  $r = r_0 + \Delta r$ . In the small deformation approach, the change in radius is assumed to be much smaller than the initial radius:  $\Delta r \ll r_0$ . The cross-sectional area in a deformed state can then be calculated from:

$$A = \pi r^2 = \pi (r_0 + \Delta r)^2 = \pi (r_0^2 + 2\Delta r r_0 + \Delta r^2) \approx \pi r_0^2. \quad (4.3)$$

This means that for small deformations the cross-sectional area is constant and the stress is simply given by  $\sigma = F/A_0 = F/A$ .

As mentioned, the strain is given by the normalized displacement  $\varepsilon = u/L_0$ . With these definitions the stress is proportional to the applied force and inversely proportional to the cross-sectional area, and the strain is given by the normalized displacement. Note that the stress can take any value, but the strain has to be larger than  $-1$ .

---

<sup>2</sup>The amount of reduction of the cross-sectional area is determined by the Poisson's ratio. Almost all materials have a positive Poisson's ratio.

The small-strain classical theory is based on the assumption that the geometric changes that occur during the loading are so small that they can be represented using a first-order linear representation. Within this theory there is only one stress and one strain measure to consider.

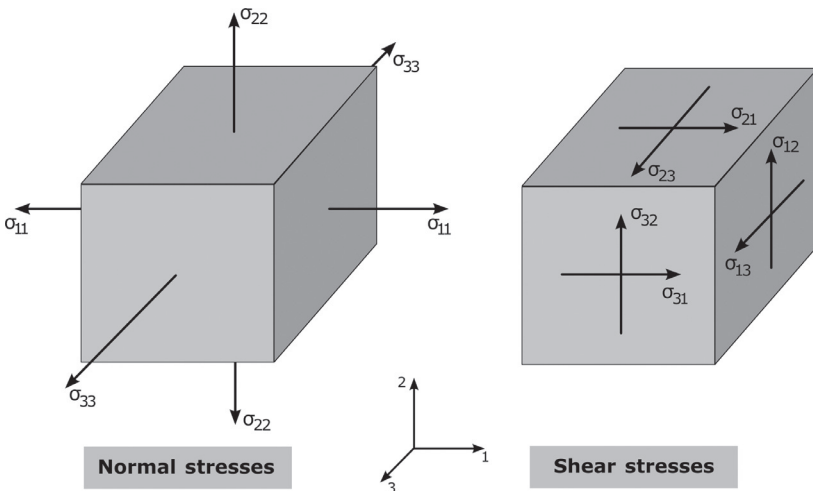
## 4.2.2 Multiaxial Loading

A small deformation multiaxial loading situation is shown in [Figure 4.2](#).

In the multiaxial case, the stress and strain at each point of the sample are characterized by six values:  $\sigma_{11}, \sigma_{12}, \sigma_{13}, \sigma_{22}, \sigma_{23}, \sigma_{33}$  and  $\varepsilon_{11}, \varepsilon_{12}, \varepsilon_{13}, \varepsilon_{22}, \varepsilon_{23}, \varepsilon_{33}$ . These values are often arranged into symmetric  $3 \times 3$  matrices:

$$\sigma_{ij} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{bmatrix}, \quad (4.4)$$

$$\varepsilon_{ij} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{12} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{13} & \varepsilon_{23} & \varepsilon_{33} \end{bmatrix}. \quad (4.5)$$



**Figure 4.2** Classical definition of multiaxial stresses. The strain components are defined analogously.

In the multiaxial case, the stress and strain at each point can be represented by normal and shear stresses on a small cube-shaped volume element, as shown in [Figure 4.2](#). The first subscript of a stress or strain quantity represents the normal direction of the face and the second subscript represents the direction of the stress or strain quantity.<sup>3</sup> Each face contains one normal stress and two shear stresses. To satisfy force equilibrium, the shear stress matrix is symmetric, that is,  $\sigma_{12} = \sigma_{21}$ ,  $\sigma_{13} = \sigma_{31}$ , and  $\sigma_{23} = \sigma_{32}$ .

The strain in the multiaxial case can be calculated in a similar way to the uniaxial case. Start by defining a displacement vector  $\mathbf{U} = [U_1, U_2, U_3]$  that varies with the position in the material specified by  $\mathbf{X} = [X_1, X_2, X_3]$ :

$$\mathbf{U}(\mathbf{X}) = \mathbf{x}(\mathbf{X}) - \mathbf{X}. \quad (4.6)$$

In this equation, (uppercase)  $\mathbf{X}$  is the initial location of the material point that in the deformed state is at location (lowercase)  $\mathbf{x}$ . The partial derivative of the displacement vector with respect to the initial position is the gradient of the displacements:

$$\text{Grad } \mathbf{U}(\mathbf{X}) = \frac{\partial \mathbf{U}(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial U_i}{\partial X_j} = \begin{bmatrix} \frac{\partial U_1}{\partial X_1} & \frac{\partial U_1}{\partial X_2} & \frac{\partial U_1}{\partial X_3} \\ \frac{\partial U_2}{\partial X_1} & \frac{\partial U_2}{\partial X_2} & \frac{\partial U_2}{\partial X_3} \\ \frac{\partial U_3}{\partial X_1} & \frac{\partial U_3}{\partial X_2} & \frac{\partial U_3}{\partial X_3} \end{bmatrix}. \quad (4.7)$$

The strain matrix is defined as the symmetric part of Equation (4.7):

$$\begin{aligned} \varepsilon_{ij} &= \frac{1}{2} \left[ \frac{\partial U_i}{\partial X_j} + \frac{\partial U_j}{\partial X_i} \right] \\ &= \begin{bmatrix} \frac{\partial U_1}{\partial X_1} & \frac{1}{2} \left( \frac{\partial U_1}{\partial X_2} + \frac{\partial U_2}{\partial X_1} \right) & \frac{1}{2} \left( \frac{\partial U_1}{\partial X_3} + \frac{\partial U_3}{\partial X_1} \right) \\ \frac{1}{2} \left( \frac{\partial U_2}{\partial X_1} + \frac{\partial U_1}{\partial X_2} \right) & \frac{\partial U_2}{\partial X_2} & \frac{1}{2} \left( \frac{\partial U_2}{\partial X_3} + \frac{\partial U_3}{\partial X_2} \right) \\ \frac{1}{2} \left( \frac{\partial U_3}{\partial X_1} + \frac{\partial U_1}{\partial X_3} \right) & \frac{1}{2} \left( \frac{\partial U_3}{\partial X_2} + \frac{\partial U_2}{\partial X_3} \right) & \frac{\partial U_3}{\partial X_3} \end{bmatrix}. \end{aligned} \quad (4.8)$$

<sup>3</sup>In some textbooks, the interpretation of the two subscripts is reversed. Since the stress and strain matrices are symmetric, this does not have a significant influence.

These definitions of stress and strain are sufficient for analyzing problems involving small deformations. The situation, however, becomes more complicated if the applied deformations become sufficient large such that the shape or size of the specimen becomes significantly different during the applied deformation. In these cases, a more general description of stress and strain is in order. The next section presents an extension of the theory suitable for large deformations.

### Example

To demonstrate the short comings of the small-strain approach consider a situation in which the cylinder in [Figure 4.1](#) is made from a flexible rubber and a large force is applied. In this case, the change in both the specimen length and cross-section area can be significant. In the small-strain theory, the stress is calculated from  $\sigma = F/A$ , but one obvious complication here is that the cross-sectional area is changing during the deformation. In the small-strain case,  $A$  is assumed to be constant, but in the large strain case that assumption is not valid. In fact, if we use the initial area  $A_0$  in the calculation of the stress we get the engineering stress:  $\sigma_{\text{eng}} = F/A_0$ , and if we use the current (actual) area we get the true stress:  $\sigma_{\text{true}} = F/A$ .

**Note:** *There is no hard rule for when the small-strain theory is sufficient, but a general rule is that the strain magnitude has to be less than 1%. If an effective strain is larger than that magnitude it is usually better to switch to large strain theory.*

*In Abaqus, you can switch on large-deformation theory by setting `NLGEOM=yes` on the `*STEP` command.*

*In ANSYS, you can switch on large-deformation theory by setting `nlgeom, on`.*

## 4.3 Large Strain Kinematics

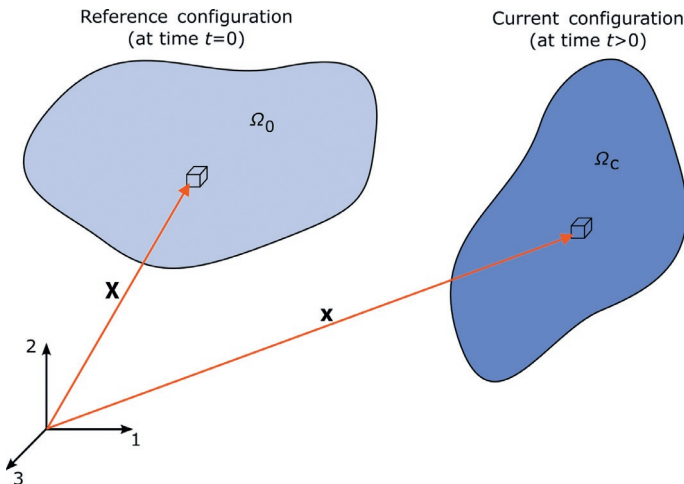
Polymer mechanics, and solid mechanics in general, are topics concerned with the evolution of stresses and strains when a component is exposed to external loads in the form of forces,

displacements, and temperatures. One of the key foundations of this topic is related to the kinematics of the deformation, that is how to mathematically express the displacements of different parts of a body of interest.

To study this topic, we will start by considering a body that initially (at time 0) has a configuration (shape and location)  $\Omega_0$  and then at a later time  $t$  has another configuration  $\Omega_c$ , see Figure 4.3.

The body can be considered to consist of a collection of small volume elements that collectively make up the body. Each of these infinitesimal volume elements is referred to as a material point. Kinematics is a topic that describes the movement of material points during a deformation event. As was discussed earlier, in the small-strain theory, the displacements are assumed to be so small that the configuration of the body does not significantly change during a loading event. This is one of the key factors that make small-strain theory easy to understand and work with.

When the deformations are finite, the shape and location of a body can undergo large changes during a loading event. The following example illustrates two different ways to keep track of the motion of a deforming body.



**Figure 4.3** Schematic figure showing a body that move and change shape as a function of time.

**Example: Lagrangian and Eulerian Formulations.**

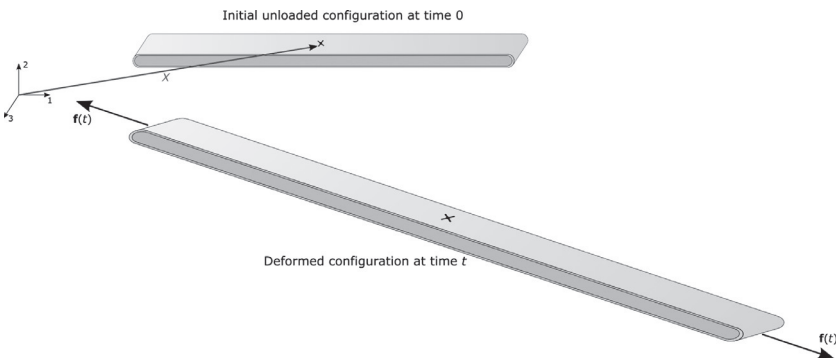
To illustrate the use of different kinematic formulations we will consider a rubber band that is axially stretched by a time-varying force  $\mathbf{f}(t)$ , see Figure 4.4. In this example, one material point on the rubber band has been marked with a  $\times$ .

One way to keep track of the stresses and strains in the rubber band is to focus on one material point at a time (e.g., the location on the rubber band with the  $\times$ ). If we specify the location of the  $\times$  in the unloaded configuration we can then uniquely express the stress and strain of that material point as a function of time. This approach where we focus on each material point labeled by its position in the reference configuration is called a *Lagrangian formulation*.

An alternative approach to specify the stresses and strains in the rubber band is to superimpose a fixed grid (coordinate system), as is shown in Figure 4.5, and then use the spatial coordinates to specify the stresses and strains in the rubber band during the loading event.

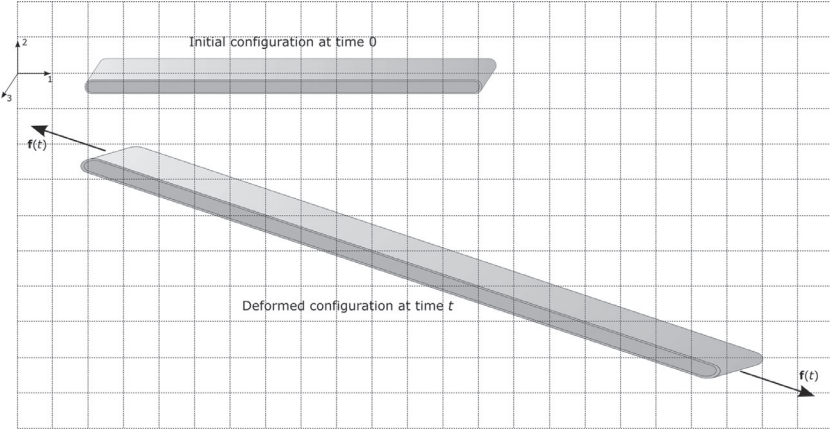
This way we can keep track of the rubber band in terms of its current spatial coordinates. This way to describe the motion is called an *Eulerian formulation*.

We can mathematically formalize the previous example by considering a material point that is initially located at  $\mathbf{X}$  and at time  $t$  located at  $\mathbf{x}(t)$ . The motion of the material point can be described by the mapping:



**Figure 4.4** Example of a Lagrangian representation of a deformation.





**Figure 4.5** Example of an Eulerian representation of a deformation.

$$\mathbf{x} = \mathcal{X}(\mathbf{X}, t), \quad (4.9)$$

where  $\mathcal{X}(\cdot)$  is a vector function that as input takes an initial position vector and a time, and as an output gives the position of that material point at the specified time. The vector  $\mathbf{X}$  is called the *reference* (or *material*) location, and the vector  $\mathbf{x}$  is called the *current* (or *spatial*) location of the material point.

When studying continuum quantities it is often necessary to follow and keep track of a certain region of a body. As discussed in the example above, there are two ways to do this. One is to label each point of the region of interest based on its initial reference location. This way we can make statements such as: “the material point that at time 0 was at the location  $\mathbf{X}$  is having a velocity of  $\mathbf{V}$  at time  $t$ .” This formulation, where everything is referred back to the initial configuration, is called a *Lagrangian formulation*.

The other way to keep track of the motion of a body is to utilize the current configuration to label the material points. This way we can make statements as: “the material point which at time  $t$  is at  $\mathbf{x}$  has a velocity of  $\mathbf{v}$ .” This formulation, where everything is referred to in the current configuration is called an *Eulerian formulation*.

The nomenclature in this text closely follows recent continuum mechanics texts (e.g., Holzapfel [2]). Specifically, quantities that are expressed in the reference configuration are denoted with

uppercase letters, and quantities that are expressed in the current configuration are denoted with lowercase letters.

The following discussion will utilize tensor notation and tensor algebra. In order to fully understand the continuum mechanics theory, it is important to have a good understanding of what tensors are and how they can be manipulated. The following section presents a brief summary of this topic.

## 4.4 Vector and Tensor Algebra

It can be shown that, in a mathematical sense, both stress and strain are second-order tensors [2, 4, 7], hence the rules of tensor algebra are of importance in polymer mechanics. For the novice, second-order tensors can at first appear to be complicated and abstract. This conception is fueled by the many different ways tensor notation have been written in different books. The approach taken here is to limit the discussion to results that are commonly used in polymer mechanics, and not to place too much emphasis of mathematical proofs. For the interested reader there are numerous resources [1–4, 9] that focus on the mathematical details.

The three types of variables that are of importance in polymer mechanics are scalars (e.g., temperature, density), vectors (e.g., force, velocity), and second-order tensors (e.g., stress, strain). In some texts, scalars are referred to as zeroth-order tensors, and vectors as first-order tensors. Here, we will not use those terms, and the second-order tensors will simply be called tensors.

### 4.4.1 Vector Operations

A vector represents a direction and a magnitude in three-dimensional (3D) space. In the following, a vector is represented by a bold-face letter, or by letters with indices, for example:

$$\mathbf{v} = v_1 \hat{\mathbf{e}}_1 + v_2 \hat{\mathbf{e}}_2 + v_3 \hat{\mathbf{e}}_3 = v_i \hat{\mathbf{e}}_i. \quad (4.10)$$

In this example,  $[\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \hat{\mathbf{e}}_3]$  is a set of orthogonal basis vectors. Here, and in the following, vectors with unit length are denoted

with a superscript hat; for example,  $\hat{\mathbf{e}}_1$ . The last term in Equation (4.10) illustrates the use of the Einstein repeated indexes summation convention: if exactly two variables in a term have the same subscript then that subscript is to be repeated with the values 1, 2, and 3. For example:

$$\begin{aligned} u_i u_i &= u_1 u_1 + u_2 u_2 + u_3 u_3 = |\mathbf{u}|^2, \\ A_{jj} &= A_{11} + A_{22} + A_{33}, \\ A_{i1} u_i &= A_{11} u_1 + A_{21} u_2 + A_{31} u_3. \end{aligned}$$

When working with vectors there are a number of operations that are commonly used. The following is a short summary of the most common vector functions and operations.

The length of a vector, also called the norm, is defined by:

$$|\mathbf{u}| = \sqrt{\mathbf{u} \cdot \mathbf{u}} = \sqrt{u_i u_i} = \sqrt{u_1^2 + u_2^2 + u_3^2}. \quad (4.11)$$

Two vectors can be added or subtracted by adding or subtracting the components of the vectors:

$$\mathbf{u} + \mathbf{v} = \sum_{i=1}^3 (u_i + v_i) \hat{\mathbf{e}}_i, \quad (4.12)$$

$$\mathbf{u} - \mathbf{v} = \sum_{i=1}^3 (u_i - v_i) \hat{\mathbf{e}}_i. \quad (4.13)$$

A scalar and a vector can be multiplied by multiplying the scalar with each of the components of the vector:

$$a\mathbf{u} = (a u_i) \hat{\mathbf{e}}_i. \quad (4.14)$$

There are different ways to multiply two vectors. A first way is through the dot-product which is defined by:

$$\mathbf{u} \cdot \mathbf{v} = u_i v_i = u_1 v_1 + u_2 v_2 + u_3 v_3 = |\mathbf{u}| \cdot |\mathbf{v}| \cdot \cos \theta, \quad (4.15)$$

where  $\theta$  is the angle between the two arbitrary vectors  $\mathbf{u}$  and  $\mathbf{v}$ . Note that the dot-product of two vectors becomes a scalar. A second way to multiply two vectors is through the cross-product which is defined by:

$$\mathbf{u} \times \mathbf{v} = \begin{bmatrix} \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 & \hat{\mathbf{e}}_3 \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{bmatrix} = u_i v_j \epsilon_{ijk} \hat{\mathbf{e}}_k, \quad (4.16)$$

where  $\epsilon_{ijk}$  is the *permutation symbol* defined by

$$\epsilon_{ijk} = \begin{cases} +1 & \text{if } (i, j, k) = (1, 2, 3) \text{ or } (2, 3, 1) \text{ or } (3, 1, 2) \\ -1 & \text{if } (i, j, k) = (1, 3, 2) \text{ or } (3, 2, 1) \text{ or } (2, 1, 3) . \\ 0 & \text{for any other combination of } i, j, k \end{cases} \quad (4.17)$$

The cross-product of two vectors is a vector quantity that is orthogonal to the two initial vectors.

#### 4.4.2 The Dyadic Product

A third way to multiply two vectors  $\mathbf{a}$  and  $\mathbf{b}$  is through the *dyad* (or tensor product) denoted by  $\mathbf{a} \otimes \mathbf{b}$ . The dyad is a second-order tensor, which will be discussed in more detail in the next section, and that can be defined by how it operates on an arbitrary vector  $\mathbf{x}$ :

$$(\mathbf{a} \otimes \mathbf{b}) \mathbf{x} = (\mathbf{b} \cdot \mathbf{x}) \mathbf{a}. \quad (4.18)$$

The dyad between two vectors can also more intuitively be written as a  $3 \times 3$  matrix:

$$\mathbf{a} \otimes \mathbf{b} = \begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix}. \quad (4.19)$$

In general, any general tensor can be written as a sum of nine dyad terms (also called a dyadic):

$$\mathbf{A} = A_{ij} \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_j \quad (\text{summation implied}). \quad (4.20)$$

For example, the identity tensor can be written as the sum of three dyads:

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \sum_{i=1}^3 \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_i = \hat{\mathbf{e}}_1 \otimes \hat{\mathbf{e}}_1 + \hat{\mathbf{e}}_2 \otimes \hat{\mathbf{e}}_2 + \hat{\mathbf{e}}_3 \otimes \hat{\mathbf{e}}_3. \quad (4.21)$$

The following pre- and post-multiplications between a dyad and a vector or second-order tensor ( $\mathbf{A}$ ) are commonly used:

$$(\mathbf{a} \otimes \mathbf{b})(\mathbf{c} \otimes \mathbf{d}) = (\mathbf{b} \cdot \mathbf{c})(\mathbf{a} \otimes \mathbf{d}), \quad (4.22)$$

$$(\mathbf{a} \otimes \mathbf{b})\mathbf{A} = \mathbf{a} \otimes (\mathbf{A}^\top \mathbf{b}), \quad (4.23)$$

$$\mathbf{A}(\mathbf{a} \otimes \mathbf{b}) = (\mathbf{A}\mathbf{a}) \otimes \mathbf{b}. \quad (4.24)$$

The proof of these relations are discussed in various text books [1–3, 10] and are here left as exercises.

### 4.4.3 Tensor Operations

A (second-order) tensor represents a quantity that contain more information than a vector: it assigns a value and direction for each value and direction, and can hence be thought of as a mapping from one vector to another. In index-form, a second-order tensor can be written as  $\mathbf{A} = A_{ij}$ .

There are different ways to interpret a second-order tensor. For example, as mentioned, a second-order tensor can be thought of as a linear operator  $\mathbf{A}$  that acts on a vector  $\mathbf{u}$  generating another vector  $\mathbf{v} = \mathbf{A}\mathbf{u}$ . In the context of this text, it is often sufficient to simply consider a second-order tensor as a  $3 \times 3$  matrix.

When working with tensors there are also numerous operations that are important. The following are definitions of the most common operations:

- Two tensors can be added (or subtracted) by adding (or subtracting) their indices:

$$(\mathbf{A} + \mathbf{B})_{ij} = A_{ij} + B_{ij} \quad (4.25)$$

$$(\mathbf{A} - \mathbf{B})_{ij} = A_{ij} - B_{ij}. \quad (4.26)$$

- A tensor can be operated on a vector creating another vector using the following multiplication and summation:

$$\mathbf{A}\mathbf{u} = A_{ij}u_j\hat{\mathbf{e}}_i = A_{11}u_1\hat{\mathbf{e}}_1 + A_{12}u_2\hat{\mathbf{e}}_2 + A_{13}u_3\hat{\mathbf{e}}_3. \quad (4.27)$$

- Two tensors can be multiplied, giving a new tensor, using the following multiplication and summation:

$$(\mathbf{AB})_{ij} = A_{ik}B_{kj} = A_{i1}B_{1j} + A_{i2}B_{2k} + A_{i3}B_{3k}. \quad (4.28)$$

- The *inner product* of two tensors (also called the *dot-product* or the *contraction*) is a scalar defined by

$$\mathbf{A} : \mathbf{B} = \text{tr}[\mathbf{AB}^\top] = A_{ij}B_{ij}. \quad (4.29)$$

- The *transpose* of a tensor  $\mathbf{A}$  is defined by

$$\mathbf{A}\mathbf{u} \cdot \mathbf{v} = \mathbf{u} \cdot \mathbf{A}^\top \mathbf{v}, \quad \text{for all vectors } \mathbf{u}, \mathbf{v}. \quad (4.30)$$

The transpose of a tensor can also be written in index notation

$$(A_{ij})^\top = A_{ji}. \quad (4.31)$$

which also gives the following useful equation

$$(\mathbf{AB})^\top = \mathbf{B}^\top \mathbf{A}^\top. \quad (4.32)$$

- The *trace* of a tensor is a scalar quantity that is given by the sum of the diagonal terms:

$$\text{tr}[\mathbf{A}] = A_{ii} = A_{11} + A_{22} + A_{33}. \quad (4.33)$$

- The *determinant* of a tensor can be calculated the same way as it is calculated for a  $3 \times 3$  matrix:

$$\det[\mathbf{A}] = \det \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}. \quad (4.34)$$

- A tensor can be uniquely decomposed into *deviatoric* and *volumetric* parts:

$$\mathbf{A} = \text{dev}[\mathbf{A}] + \text{vol}[\mathbf{A}], \quad (4.35)$$

where

$$\text{dev}[\mathbf{A}] = \mathbf{A} - \frac{1}{3} \text{tr}[\mathbf{A}]\mathbf{I}, \quad (4.36)$$

$$\text{vol}[\mathbf{A}] = \frac{1}{3} \text{tr}[\mathbf{A}]\mathbf{I}. \quad (4.37)$$

A deviatoric tensor has a trace of zero. This decomposition is useful when working with the deformation gradient, as will be discussed in later chapters.

- A tensor can also be decomposed into a product of *distortional* and *dilatational* parts:

$$\mathbf{A} = \text{distortional}[\mathbf{A}] \text{ dilatational}[\mathbf{A}], \quad (4.38)$$

where

$$\text{distortional}[\mathbf{A}] = \det[\mathbf{A}]^{-1/3} \mathbf{A}, \quad (4.39)$$

$$\text{dilatational}[\mathbf{A}] = (\det[\mathbf{A}])^{1/3} \mathbf{I}. \quad (4.40)$$

A distortional tensor has a determinant of zero. This deformation gradient is useful when working with the deformation gradient, as will be discussed in later chapters.

- An *orthogonal* tensor  $\mathbf{Q}$  is a tensor with the following properties:

$$\mathbf{Q}^\top = \mathbf{Q}^{-1} \quad (4.41)$$

$$\det \mathbf{Q} = \mathbf{1}. \quad (4.42)$$

- A *diagonal* tensor is a tensor with zero off-diagonal terms:

$$A_{ij} = 0, \quad \text{if } i \neq j. \quad (4.43)$$

- The components of a tensor  $A_{ij}$  can be determined from the unit vectors  $\hat{\mathbf{e}}_i$  and  $\hat{\mathbf{e}}_j$  as follows:

$$A_{ij} = \mathbf{A} \hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_j. \quad (4.44)$$

- It is often useful to calculate functions of a tensor, such as  $\exp(\mathbf{A})$ . One way to calculate these functions is to write the tensor  $\mathbf{A}$  in its spectral representation (see [Section 4.5.1](#)) and then apply the function on the principal values of the tensor:

$$f(\mathbf{F}) = \sum_{i=1}^3 f(\lambda_i) \hat{\mathbf{n}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.45)$$

This theorem is discussed in more detail in linear algebra texts [11, 12].

From the discussion above it is clear that direct notation is more compact and often easier to understand than index notation. For these reasons direct notation will be used almost exclusively in the following.

#### 4.4.4 Derivatives of Scalar, Vector, and Tensor Fields

When formulating and integrating certain advanced constitutive theories it is important to incorporate time and spacial derivatives of scalar, vector, and tensor fields. This section summarizes the most commonly used approach to perform these derivatives.

First consider a scalar function  $a(\mathbf{x}, t)$  that is expressed in the spatial configuration. The *spatial time derivative of the spatial field*  $a(\mathbf{x}, t)$  is given by the partial derivative of  $a$  with respect to  $t$  holding  $\mathbf{x}$  constant:

$$\frac{\partial a(\mathbf{x}, t)}{\partial t}. \quad (4.46)$$

Another commonly used time derivative is the *material time derivative of the spatial field*  $a(\mathbf{x}, t)$ . This time derivative is defined by the partial derivative of  $a$  with respect to  $t$  holding  $\mathbf{X}$  constant:

$$\frac{\partial a(\mathbf{x}, t)}{\partial t} = \frac{\partial a(\mathbf{x}, t)}{\partial t} + \frac{\partial a(\mathbf{x}, t)}{\partial \mathbf{x}} \cdot \frac{\partial \mathcal{X}(\mathbf{X}, t)}{\partial t}, \quad (4.47)$$

giving

$$\frac{D a(\mathbf{x}, t)}{D t} = \dot{a}(\mathbf{x}, t) = \frac{\partial a(\mathbf{x}, t)}{\partial t} + (\text{grad } a(\mathbf{x}, t)) \cdot \mathbf{v}(\mathbf{x}, t). \quad (4.48)$$

This can be written as a total derivative

$$\frac{D}{D t} a(\mathbf{x}, t) = \dot{a}(\mathbf{x}, t) = \left[ \frac{\partial}{\partial t} + \mathbf{v} \cdot \frac{\partial}{\partial \mathbf{x}} \right] a(\mathbf{x}, t). \quad (4.49)$$



Spacial derivatives are commonly used in addition to time derivatives. The gradient of a scalar and a vector field is defined as follows:

$$\text{grad } \phi(\mathbf{x}) = \frac{\partial \phi}{\partial x_i} \hat{\mathbf{e}}_i, \quad (4.50)$$

$$\text{grad } \mathbf{u}(\mathbf{x}) = \frac{\partial u_i}{\partial x_j} \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_j. \quad (4.51)$$

The divergence of a vector field  $\mathbf{u}(\mathbf{x})$  and a tensor field  $\mathbf{A}(\mathbf{x})$  is defined by

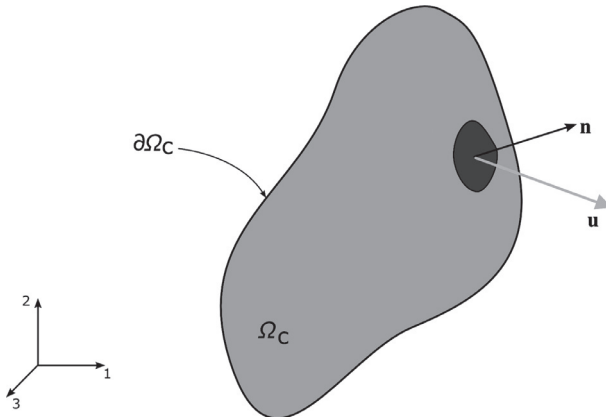
$$\text{div } \mathbf{u} = \frac{\partial u_i}{\partial x_i}. \quad (4.52)$$

$$\text{div } \mathbf{A} = \frac{\partial A_{ij}}{\partial x_j} \hat{\mathbf{e}}_i. \quad (4.53)$$

The divergence theorem is commonly used in theoretical polymer mechanics and will be extensively used in the following sections. As shown in [Figure 4.6](#), let  $\mathbf{u}(\mathbf{x})$  be a vector field defined on a domain  $\Omega_c$  with the boundary  $\partial\Omega_c$ .

The divergence theorem [[1](#), [2](#), [10](#)] then states that the surface integral of a vector field can be related to the volume integral of the divergence of the same vector field as follows:

$$\int_{\partial\Omega_c} \mathbf{u} \cdot \hat{\mathbf{n}} \, ds = \int_{\Omega_c} \text{div } \mathbf{u} \, dv, \quad (4.54)$$



**Figure 4.6** Definition of the domain  $\Omega_c$  that is studied in the divergence theorem.

where  $\hat{\mathbf{n}}$  is the surface normal,  $ds$  is a surface area element, and  $dv$  is a volume element.

There are other more general forms of the divergence theorem, and other integral theorems such as the Stokes theorem. These theorems and the proof of Equation (4.54) are discussed in introductory vector algebra books [10]. For the topics of this book it is sufficient to only cover the divergence theorem as presented in Equation (4.54).

#### 4.4.5 Coordinate Transformations

In polymer mechanics analysis, it is often necessary to perform coordinate transformations. To illustrate how to perform these transformations we will consider two coordinate systems  $\hat{\mathbf{e}}'_i$  and  $\hat{\mathbf{e}}_i$  that are related by a rotation  $\mathbf{Q}$

$$\hat{\mathbf{e}}'_i = \mathbf{Q}^\top \hat{\mathbf{e}}_i, \quad (4.55)$$

where  $\mathbf{Q}$  is an orthogonal tensor. Now consider one component of  $\mathbf{Q}$ :

$$Q_{ij} = \mathbf{Q} \hat{\mathbf{e}}_j \cdot \hat{\mathbf{e}}_i \quad (4.56)$$

$$= \hat{\mathbf{e}}_j \cdot \mathbf{Q}^\top \hat{\mathbf{e}}_i \hat{\mathbf{e}}_j \cdot \hat{\mathbf{e}}'_i \quad (4.57)$$

$$= \cos(\text{angle between } \hat{\mathbf{e}}'_i \text{ and } \hat{\mathbf{e}}_j). \quad (4.58)$$

Hence, each component of  $\mathbf{Q}$  is given by the dot-product of the corresponding unit vectors. Since an arbitrary vector can be written  $\mathbf{x} = x_i \hat{\mathbf{e}}_i$ , we see that the coordinate transformation implies the vector transformation:

$$\mathbf{x}' = \mathbf{Q}\mathbf{x}. \quad (4.59)$$

Similarly, as will be shown in Section 4.12, a second-order tensor is transformed according to:

$$\mathbf{A}' = \mathbf{Q}\mathbf{A}\mathbf{Q}^\top, \quad (4.60)$$

where  $Q_{ij}$  is equal to cosine between the basis vectors  $\hat{\mathbf{e}}'_i$  and  $\hat{\mathbf{e}}_j$ .

#### 4.4.6 Invariants

The invariants of a tensor are very important for many polymer mechanics constitutive theories. A second-order tensor has three invariants that are related to the eigenvalues defined by:

$$\mathbf{A}\hat{\mathbf{n}}_i = \lambda_i\hat{\mathbf{n}}_i, \quad (4.61)$$

which can also be written

$$(\mathbf{A} - \lambda_i\mathbf{I}) = 0. \quad (4.62)$$

This equation only has nontrivial solutions if

$$\det(\mathbf{A} - \lambda_i\mathbf{I}) = 0, \quad (4.63)$$

where

$$\det(\mathbf{A} - \lambda_i\mathbf{I}) = -\lambda_i^3 + I_1\lambda_i^2 - I_2\lambda_i + I_3 = 0. \quad (4.64)$$

This cubic polynomial in  $\lambda_i$  is called the *characteristic polynomial*. The scalar values  $I_1$ ,  $I_2$ , and  $I_3$  are the principal invariants of  $\mathbf{A}$  and are given by:

$$I_1(\mathbf{A}) = \text{tr } \mathbf{A} = \lambda_1 + \lambda_2 + \lambda_3, \quad (4.65)$$

$$I_2(\mathbf{A}) = \frac{1}{2} \left[ (\text{tr } \mathbf{A})^2 - \text{tr}(\mathbf{A}^2) \right] = \lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_3\lambda_1, \quad (4.66)$$

$$I_3(\mathbf{A}) = \det \mathbf{A} = \lambda_1\lambda_2\lambda_3. \quad (4.67)$$

As will be discussed in Chapter 5, the invariants of the deformation gradient are used to formulate hyperelastic constitutive models. What the deformation gradient is and how it can be used is the topic of the next section.

## 4.5 Deformation Gradient

One of the most important goals of polymer mechanics is to determine the stress state as a function of applied displacements and loads. The stress at a given material point is determined by how stretched and distorted the material is at that point relative to its initial undeformed configuration. One convenient way to express the local stretch state is to use the deformation gradient  $\mathbf{F}$  defined by:

$$\mathbf{F} = \frac{\partial \mathcal{X}(\mathbf{X}, t)}{\partial \mathbf{X}}, \quad (4.68)$$

which in index notation can be written  $F_{ij} = \partial x_i / \partial X_j$ . The deformation gradient is a very important descriptor of the applied deformation state and is extensively used in both theoretical and computational works. In fact, it can be said that: *deformation gradients rule theoretical polymer mechanics*.

The following examples illustrate the definition and use of the deformation gradient  $\mathbf{F}$ .

**Example: Deformation Gradient in the Undeformed State.**

If a specimen is undeformed, then the current state is equal to the reference state and the deformation gradient is equal to the identity tensor

$$\mathbf{F} = \mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

**Example: Uniaxial Tension.**

Consider a cube that is uniaxially stretched in the one-direction by a stretch ratio  $\lambda = L/L_0$ . Further assume that the contraction in the two- and three-directions are equal and that the total volume is conserved. In this case, the coordinates of a material point in the current configuration is given by

$$x_1 = \lambda \cdot X_1 \quad (4.69)$$

$$x_2 = \frac{1}{\sqrt{\lambda}} \cdot X_2 \quad (4.70)$$

$$x_3 = \frac{1}{\sqrt{\lambda}} \cdot X_3. \quad (4.71)$$

The deformation gradient in this case can be directly obtained from the definition (4.68)

$$\mathbf{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & 1/\sqrt{\lambda} & 0 \\ 0 & 0 & 1/\sqrt{\lambda} \end{bmatrix}. \quad (4.72)$$

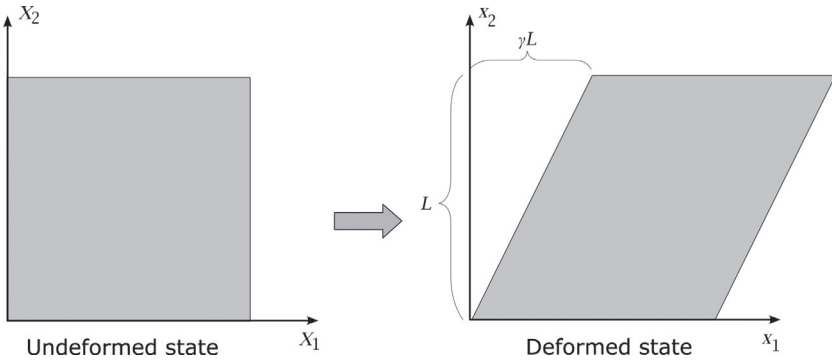
**Example: Simple Shear.**

Consider a specimen that is deformed in simple shear. In this case, the coordinates of a material point in the current configuration are given by

$$x_1 = X_1 + \gamma X_2, \quad (4.73)$$

$$x_2 = X_2, \quad (4.74)$$

$$x_3 = X_3. \quad (4.75)$$



Note that we have assumed that there is no deformation in the three-direction. The deformation gradient in this case is given by:

$$\mathbf{F} = \begin{bmatrix} 1 & \gamma & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (4.76)$$

illustrating that  $\mathbf{F}$  in general is not symmetrical.

From the definition of the deformation gradient we can also write  $\mathbf{F}$  as a mapping or linear transformation:

$$d\mathbf{x} = \mathbf{F}(\mathbf{X}) d\mathbf{X}. \quad (4.77)$$

That is, by operating  $\mathbf{F}$  on a vector  $d\mathbf{X}$  in the reference configuration we get the location of that vector in the current configuration. This is a general result: operating a second-order tensor (such as  $\mathbf{F}$ ) on a vector creates another vector that is the linear transformation of the second-order tensor ( $\mathbf{F}$ ). This can also be expressed in matrix terminology: multiplying a  $3 \times 3$  matrix (such as  $\mathbf{F}$ ) with a  $3 \times 1$  vector gives another  $3 \times 1$  vector.

Since the deformation gradient  $\mathbf{F}$  is a linear transformation from the reference configuration to the current configuration, the inverse of the deformation gradient  $\mathbf{F}^{-1}$  must also exist and is the transformation from the current configuration to the reference configuration:

$$d\mathbf{X} = \mathbf{F}^{-1} d\mathbf{x}. \quad (4.78)$$

Furthermore, if we first operate with  $\mathbf{F}$  and then with  $\mathbf{F}^{-1}$  (or first with  $\mathbf{F}^{-1}$  and then with  $\mathbf{F}$ ), then we get back the original configuration, that is  $\mathbf{F}^{-1}\mathbf{F} = \mathbf{F}\mathbf{F}^{-1} = \mathbf{I}$ , giving  $\mathbf{X} = \mathbf{F}^{-1}\mathbf{F}\mathbf{X}$ .

**Example: Volumetric Deformation.**

One important use of the deformation gradient is that it specifies how the volume is changed during a transformation. If  $dV$  is a small volume element in the reference configuration, then that volume element is transformed into the volume element

$$dv = \det(\mathbf{F}) dV \quad (4.79)$$

in the current configuration. The quantity  $\det(\mathbf{F})$  is commonly referred to as  $J$  (the Jacobian determinant)

$$J = \det \mathbf{F}. \quad (4.80)$$

The proof of this theorem is given in different text books [2, 10] and is here left as an exercise.

The deformation of a body can be divided into different classes depending on the structure of the deformation gradient:

- If  $\mathbf{F}$  does not vary from location to location in a body then the deformation is said to be *homogeneous*.
- If  $\mathbf{F}(\mathbf{X})$  is a function of the position then the deformation is said to be *inhomogeneous*.
- If  $J = \det \mathbf{F} = 1$  then the deformation is said to be *isochoric*.

### 4.5.1 Eigenvalue and Spectral Decompositions

As is known from linear algebra [12], any symmetric tensor  $\mathbf{A}$  can be characterized by its eigenvalues  $(\lambda_1, \lambda_2, \lambda_3)$  and its eigenvectors  $(\hat{\mathbf{n}}_1, \hat{\mathbf{n}}_2, \hat{\mathbf{n}}_3)$  by:

$$\mathbf{A}\hat{\mathbf{n}}_i = \lambda_i\hat{\mathbf{n}}_i \quad (\text{for } i=1, 2, 3; \text{ no summation implied}). \quad (4.61\text{-repeat})$$

In this equation, and in the following, unit vectors are distinguished from other vector quantities by a superimposed hat-sign; for example,  $\hat{\mathbf{n}}$ . For a symmetric tensor  $\mathbf{A}$ , the eigenvalues  $(\lambda_i)$  are real and the eigenvectors  $(\hat{\mathbf{n}}_i)$  form a mutually orthogonal basis of unit vectors. The tensor  $\mathbf{A}$  can in this case be written in its *eigenvalue representation* as:

$$\mathbf{A} = \sum_{i=1}^3 \lambda_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i. \quad (4.81)$$

This eigenvalue representation is based on the dyadic product that was defined and discussed in Section 4.4.2.

Another way of writing the eigenvalue decomposition of a symmetric tensor  $\mathbf{A}$  is:

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top, \quad (4.82)$$

where  $\mathbf{Q}$  is orthogonal ( $\mathbf{Q}^\top = \mathbf{Q}^{-1}$ ), and  $\mathbf{\Lambda}$  is diagonal ( $\mathbf{\Lambda} = \sum_{i=1}^3 \lambda_i \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_i$ ). Hence,  $\mathbf{A}$  can also be written

$$\mathbf{A} = \mathbf{Q} \left[ \sum_{i=1}^3 \lambda_i \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_i \right] \mathbf{Q}^\top = \sum_{i=1}^3 \lambda_i (\mathbf{Q}\hat{\mathbf{e}}_i) \otimes (\mathbf{Q}\hat{\mathbf{e}}_i) = \sum_{i=1}^3 \lambda_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i, \quad (4.81\text{-repeat})$$

which is another way of writing the eigenvalue representation of  $\mathbf{A}$ .

The focus of this section is on the deformation gradient  $\mathbf{F}$ , which in general is nonsymmetric and hence does not have real eigenvalues [12]. Instead of using the eigenvalue decomposition, the deformation gradient  $\mathbf{F}$  is often decomposed using the *singular value decomposition*:

$$\mathbf{F} = \mathbf{Q}_1 \mathbf{\Lambda} \mathbf{Q}_2, \quad (4.83)$$

where  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  are two different orthogonal tensors (rotations), and  $\mathbf{\Lambda}$  is diagonal. Hence, the singular value decomposition of  $\mathbf{F}$  can also be expressed as follows:

$$\mathbf{F} = \mathbf{Q}_1 \left[ \sum_{i=1}^3 \lambda_i \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_i \right] \mathbf{Q}_2^T \quad (4.84)$$

$$\mathbf{F} = \sum_{i=1}^3 \lambda_i (\mathbf{Q}_1 \hat{\mathbf{e}}_i) \otimes (\mathbf{Q}_2 \hat{\mathbf{e}}_i) \quad (4.85)$$

$$\mathbf{F} = \sum_{i=1}^3 \lambda_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.86)$$

In these equations,  $\lambda_i$  is the *principal stretches* of the deformation,  $\hat{\mathbf{n}}_i$  and  $\hat{\mathbf{N}}$  are the basis vectors of the deformation gradient. This way of writing a tensor as a sum of the dyadic product of its eigenvectors is called the *spectral representation*.

The following two examples illustrate how the deformation gradient can be used to transform line and area elements.

#### **Example: Transformation of Line Elements.**

Consider a vector  $d\mathbf{X} = dS \hat{\mathbf{N}}$  in the reference configuration, where  $dS$  is the length of the vector and  $\hat{\mathbf{N}}$  is a unit vector in the direction of the vector  $d\mathbf{X}$ .

If we operate with  $\mathbf{F}$  on  $d\mathbf{X}$  we get

$$d\mathbf{x} = \mathbf{F} d\mathbf{X}. \quad (4.77\text{-repeat})$$

The length of  $d\mathbf{x}$  is

$$|d\mathbf{x}| \equiv ds = \left| \mathbf{F} (dS \hat{\mathbf{N}}) \right| = dS \left| \mathbf{F} \hat{\mathbf{N}} \right|. \quad (4.87)$$

Hence, the length of a vector in the current configuration is equal to the length of that vector in the reference configuration times  $|\mathbf{F} \hat{\mathbf{N}}|$ .



**Example: Transformation of Area Elements (Nanson's Formula).**

Let  $d\mathbf{S}$  be an area element in the reference configuration with a unit normal vector  $\hat{\mathbf{N}}$ , and let  $ds$  be the corresponding area element in the current configuration with a unit normal  $\hat{\mathbf{n}}$ .

In this case, the corresponding volume element  $dv$  in the current configuration can be written

$$dv = ds \cdot d\mathbf{x} = Jd\mathbf{S} \cdot d\mathbf{X}, \quad (4.88)$$

but  $d\mathbf{x} = \mathbf{F}d\mathbf{X}$  giving

$$ds \cdot (\mathbf{F}d\mathbf{X}) - Jd\mathbf{S} \cdot d\mathbf{X} = 0 \quad (4.89)$$

$$\mathbf{F}^\top ds = Jd\mathbf{S} \quad (4.90)$$

$$ds = J\mathbf{F}^{-T}d\mathbf{S}. \quad (4.91)$$

This relation between area elements is often called the Nanson's formula and is useful when defining different stress measures.

When working with continuum mechanics formulations it is often necessary to consider the gradient and the divergence of scalar and vector fields. The following two examples illustrate how to perform these operations in the reference and the current configurations.

**Example: Gradient of a Scalar Field.**

Consider a scalar field that is  $a(\mathbf{x}, t)$  in spatial coordinates and  $A(\mathbf{X}, t)$  in reference (material) coordinates. The gradient of this field in the reference frame is then the space derivative of  $A(\mathbf{X}, t)$  and can be written

$$\frac{\partial A(\mathbf{X}, t)}{\partial X_i} = \frac{\partial a(\mathbf{x}, t)}{\partial x_k} \frac{\partial x_k}{\partial X_i}, \quad (4.92)$$

which is equivalent to

$$\text{Grad} A = \mathbf{F}^\top \text{grad} a. \quad (4.93)$$

In this equation, and in the following, we will use  $\text{Grad}$  to represent the gradient in the reference configuration and  $\text{grad}$  to represent the gradient with respect to the spatial coordinates.

**Example: Divergence of a Vector Field.**

The divergence of a reference vector field  $\mathbf{U}(\mathbf{X}, t)$  can be written

$$\text{Div } \mathbf{U}(\mathbf{X}, t) = \frac{\partial U_i(\mathbf{X}, t)}{\partial X_i} = \frac{\partial u_i(\mathbf{x}, t)}{\partial x_k} \frac{\partial x_k}{\partial X_i} = \text{grad } \mathbf{U}(\mathbf{X}, t) : \mathbf{F}^\top. \quad (4.94)$$

Similarly, the divergence of a spatial vector field  $\mathbf{u}(\mathbf{x}, t)$  can be written

$$\text{div } \mathbf{u}(\mathbf{x}, t) = \frac{\partial u_i(\mathbf{x}, t)}{\partial x_i}. \quad (4.95)$$

In these equations, and in the following, we will use  $\text{Div}$  to represent the gradient in the reference configuration and  $\text{div}$  to represent the gradient with respect to the spatial coordinates.

**4.6 Strain, Stretch, and Rotation**

From the polar decomposition theorem [2, 10] it is known that any general deformation can be uniquely decomposed into a rotation followed by a stretch component, or a stretch component followed by a rotation:

$$\mathbf{F} = \mathbf{R}\mathbf{U} = \mathbf{v}\mathbf{R}, \quad (4.96)$$

where

- $\mathbf{F}$  is the deformation gradient,
- $\mathbf{R}$  is the *rotation tensor*, which is orthogonal, ( $\mathbf{R}^{-1} = \mathbf{R}^\top$ ) and volume conserving ( $\det \mathbf{R} = 1$ ),
- $\mathbf{U}$  is the *right stretch tensor*, which is positive definite and symmetric ( $\mathbf{U} = \mathbf{U}^\top$ ),
- $\mathbf{v}$  is the *left stretch tensor*, which is positive definite and symmetric ( $\mathbf{v} = \mathbf{v}^\top$ ).

Note that if  $\mathbf{U} = \mathbf{v} = \mathbf{I}$ , then  $\mathbf{F} = \mathbf{R}$  is a *rigid body rotation*. Similarly, if  $\mathbf{R} = \mathbf{I}$  then  $\mathbf{F} = \mathbf{U} = \mathbf{v}$  is a *pure stretch*.

From Equation (4.96) we know that the right stretch tensor  $\mathbf{U}$  is symmetric, hence it can be written in a spectral representation as

$$\mathbf{U} = \sum_{i=1}^3 \lambda_i \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.97)$$

In this equation,  $\lambda_i$  is the principal stretches and  $\hat{\mathbf{N}}_i$  is the corresponding eigenvectors of the tensor  $\mathbf{U}$ . Similarly the left stretch tensor can be written

$$\mathbf{v} = \sum_{i=1}^3 \lambda_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i. \quad (4.98)$$

Note that  $\mathbf{U}$  and  $\mathbf{v}$  have the same principal stretches but different principal directions (basis vectors) since they are expressed in different reference frames. The eigenvectors of  $\mathbf{v}$  can be related to the eigenvectors of  $\mathbf{U}$  by solving for  $\mathbf{v}$  in Equation (4.96):

$$\mathbf{v} = \mathbf{R}\mathbf{U}\mathbf{R}^T = \sum_{i=1}^3 \lambda_i (\mathbf{R}\hat{\mathbf{N}}_i) \otimes (\mathbf{R}\hat{\mathbf{N}}_i). \quad (4.99)$$

Hence, the eigenvectors of  $\mathbf{U}$  and  $\mathbf{v}$  are related by

$$\hat{\mathbf{n}}_i = \mathbf{R}\hat{\mathbf{N}}_i \quad (4.100)$$

and

$$\hat{\mathbf{N}}_i = \mathbf{R}^T \hat{\mathbf{n}}_i. \quad (4.101)$$

The deformation gradient can also be expressed in spectral form by  $\mathbf{F} = \mathbf{R}\mathbf{U}$  giving

$$\mathbf{F} = \mathbf{R} \sum_{i=1}^3 \lambda_i \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i = \sum_{i=1}^3 \lambda_i (\mathbf{R}\hat{\mathbf{N}}_i) \otimes \hat{\mathbf{N}}_i = \sum_{i=1}^3 \lambda_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{N}}_i, \quad (4.102)$$

which is the same result that was presented in Equation (4.86). Since the basis vectors of  $\mathbf{F}$  contain both  $\hat{\mathbf{n}}_i$  and  $\hat{\mathbf{N}}_i$ ,  $\mathbf{F}$  is called a *two-point tensor*.

The rotation tensor  $\mathbf{R}$  can be written as

$$\mathbf{R} = \mathbf{R}\mathbf{I} = \mathbf{R} \sum_{i=1}^3 \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i, \quad (4.103)$$

giving

$$\mathbf{R} = \sum_{i=1}^3 \hat{\mathbf{n}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.104)$$

There are a number of other important and useful tensor quantities that are defined from the polar decomposition of  $\mathbf{F}$ :

- $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$  is the *right Cauchy-Green tensor*. Since  $\mathbf{F} = \mathbf{R}\mathbf{U}$ , we get  $\mathbf{C} = \mathbf{U}^\top \mathbf{R}^\top \mathbf{R} \mathbf{U} = \mathbf{U}^2$ . Hence,  $\mathbf{C}$  can also be written

$$\mathbf{C} = \mathbf{F}^\top \mathbf{F} = \sum_{i=1}^3 \lambda_i^2 \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.105)$$

- $\mathbf{b} = \mathbf{F}\mathbf{F}^\top$  is the *left Cauchy-Green tensor*.<sup>4</sup> From Equation (4.96) the tensor  $\mathbf{b}$  can also be written

$$\mathbf{b} = \mathbf{F}\mathbf{F}^\top = \mathbf{v}\mathbf{R}\mathbf{R}^\top \mathbf{v}^\top = \mathbf{v}^2. \quad (4.106)$$

Hence,  $\mathbf{b}$  is also given by

$$\mathbf{b} = \sum_{i=1}^3 \lambda_i^2 \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i. \quad (4.107)$$

### Example: How to Calculate the Polar Decomposition of a Tensor.

One of the easiest ways to calculate the polar decomposition of a tensor  $\mathbf{F} = \mathbf{R}\mathbf{U}$  is to recall that  $\mathbf{C} = \mathbf{F}^\top \mathbf{F} = \mathbf{U}^2$ . This allows  $\mathbf{U}$  to be determined from  $\mathbf{U} = \sqrt{\mathbf{F}\mathbf{F}^\top}$ . One way to perform the square root operation is to write  $\mathbf{U}^2$  in its spectral representation by calculating the eigenvalues and eigenvectors of  $\mathbf{U}^2$ :

$$\mathbf{U}^2 = \sum_{i=1}^3 \lambda_i^2 \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.108)$$

The tensor  $\mathbf{U}$  can then be calculated from

$$\mathbf{U} = \sum_{i=1}^3 \lambda_i \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i. \quad (4.109)$$

<sup>4</sup>The left versus right terms in the definitions of the Cauchy-Green tensors is determined by which side  $\mathbf{F}$  is compared to  $\mathbf{F}^\top$ .

Once  $\mathbf{U}$  has been determined, the tensor  $\mathbf{R}$  can be calculated from

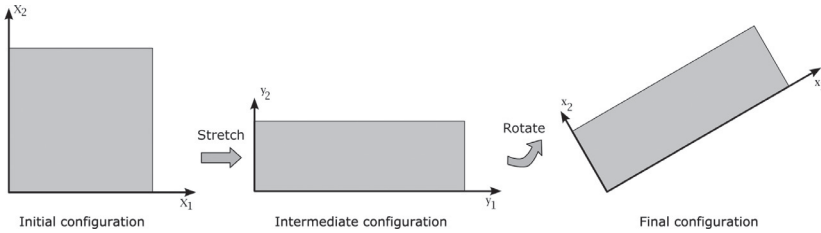
$$\mathbf{R} = \mathbf{F}\mathbf{U}^{-1} \quad (4.110)$$

and  $\mathbf{v}$  from

$$\mathbf{v} = \mathbf{R}\mathbf{r}^\top. \quad (4.111)$$

**Example: Determination of the Deformation Gradient for a Case of Stretch and Rotation.**

Consider a deformation that is performed in two steps: first, the material is stretched by a factor of two in the one-direction, and then rotated  $45^\circ$  around the three-direction, see the figure below.



The stretch deformation is given by:

$$\begin{cases} y_1 = 2X_1 \\ y_2 = 0.5X_2 \\ y_3 = X_3 \end{cases} \quad (4.112)$$

giving the right stretch tensor:

$$\mathbf{U} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (4.113)$$

The components of the rotation can be written

$$\begin{cases} x_1 = \cos \theta y_1 + \sin \theta y_2 \\ x_2 = -\sin \theta y_1 + \cos \theta y_2 \\ x_3 = y_3 \end{cases} \quad (4.114)$$

giving the rotation tensor

$$\mathbf{R} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (4.115)$$

In this case,  $\theta = 45^\circ$  giving the deformation gradient  $\mathbf{F} = \mathbf{R}\mathbf{U}$ :

$$\mathbf{F} = \begin{bmatrix} 1.41 & 0.35 & 0 \\ -1.41 & 0.35 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (4.116)$$

### Example: Numerical Calculation of the Deformation Gradient for a Case of Stretch and Rotation.

The polar decomposition is straightforward to calculate using a high-level math application or language. For example, Matlab [13], Mathematica [14], and Python [15] are all suitable tools. Many of the examples in this book are based on Python with NumPy and SciPy. Since these tools are very mature, powerful, and free, it is an excellent choice for numerical calculations.

In this example, we will start with the deformation gradient in Equation (4.116) and then calculate the  $\mathbf{U}$  and  $\mathbf{R}$  tensors. The Python code for this calculation is shown below.

Python code:	⇒	Output:
<pre> from pylab import * from scipy.linalg import sqrtm, inv  # initialize U U = array([[2,0,0], [0,0.5,0], [0,0,1]]) print('U=', U)  # initialize R v = 45.0 * pi/180.0 c = cos(v) s = sin(v) R = array([[c,s,0], [-s,c,0], [0,0,1]]) print('R=', R)  # calculate F F = dot(R,U) print('F=RU=', F)  # calculate U from F U = real(sqrtm(dot(F.T, F))) print('U=', U)  # calculate R from F and U R = dot(F, inv(U)) print('R=', R)  # if RU=F then it worked print('R*U-F=', dot(R,U)-F) </pre>		<pre> U= [[ 2.  0.  0. ]      [ 0.  0.5  0. ]      [ 0.  0.  1. ]]  R= [[ 0.70710678  0.70710678  0. ]      [-0.70710678  0.70710678  0. ]      [ 0.  0.  1. ]]  F=RU= [[ 1.41421356  0.35355339  0. ]         [-1.41421356  0.35355339  0. ]         [ 0.  0.  1. ]]  U= [[ 2.  0.  0. ]      [ 0.  0.5  0. ]      [ 0.  0.  1. ]]  R= [[ 0.70710678  0.70710678  0. ]      [-0.70710678  0.70710678  0. ]      [ 0.  0.  1. ]]  R*U-F= [[ 0.  0.  0. ]          [ 0.  0.  0. ]          [ 0.  0.  0. ]] </pre>

The strain at a material point, by definition, should not be influenced by rigid body rotations. This means that the strain tensor cannot directly depend on the deformation gradient  $\mathbf{F}$ , since  $\mathbf{F}$  also depends on rotations. Instead the strain tensor has to depend on the right or left stretch tensors. If the right stretch tensor ( $\mathbf{U}$ ) is used to define the strains then the strain will be expressed in the reference configuration, and if the left stretch tensor ( $\mathbf{v}$ ) is used, the strain will be expressed in the current configuration.

The strain tensor in the reference configuration can most generally be written as a function of  $\mathbf{U}$ :

$$\mathbf{E} = \hat{\mathbf{E}}(\mathbf{U}). \quad (4.117)$$

This tensorial dependence can be expressed in the spectral representation

$$\mathbf{E} = \mathbf{E}(\mathbf{U}) = \sum_{i=1}^3 f(\lambda_i) \hat{\mathbf{N}}_i \otimes \hat{\mathbf{N}}_i, \quad (4.118)$$

where  $f(\lambda_i)$  is a scalar function of the principal stretches. Hence,  $\mathbf{E}$  will be coaxial<sup>5</sup> with  $\mathbf{U}$  but have different eigenvalues. The function  $f(\lambda_i)$  need to fulfill three requirements for  $\mathbf{E}$  to become a valid strain tensor:

1. The strain has to be zero in the undeformed state, hence  $f(1) = 0$ .
2. At small deformations the strain should become equal to the classical strain defined in [Section 4.2](#), hence  $f'(1) = 1$ .
3. The strain has to increase monotonically with the applied deformation, that is  $f(\lambda)$  has to monotonically increase with increasing  $\lambda$ .

---

<sup>5</sup>Two tensors are said to be coaxial if they have the same basis vectors.

The following are commonly used strain measures expressed in the reference configuration. These strains are also called Lagrangian strains.

- The *Green-Lagrange strain*, which corresponds to  $f(\lambda_i) = \frac{1}{2}(\lambda_i^2 - 1)$ , can be written

$$\mathbf{E} = \frac{1}{2} [\mathbf{U}^2 - \mathbf{I}]. \quad (4.119)$$

- The *Hencky strain* (also called the *true strain*, or the *logarithmic strain*), which corresponds to  $f(\lambda_i) = \ln \lambda_i$ , can be written

$$\mathbf{E} = \ln \mathbf{U}. \quad (4.120)$$

- The *Biot strain*, corresponding to  $f(\lambda_i) = \lambda_i - 1$ , can be written

$$\mathbf{E} = \mathbf{U} - \mathbf{I}. \quad (4.121)$$

- The *Almansi strain*, corresponding to  $f(\lambda_i) = \frac{1}{2}(1 - \lambda_i^{-2})$ , can also be written

$$\mathbf{E} = \frac{1}{2} [\mathbf{I} - \mathbf{U}^{-2}]. \quad (4.122)$$

It is also possible to formulate the strain in the current (spatial) configuration:

$$\mathbf{e} = \mathbf{e}(\mathbf{v}), \quad (4.123)$$

which can also be expressed in its spectral representation

$$\mathbf{e} = \sum_{i=1}^3 f(\lambda_i) \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i. \quad (4.124)$$

The following are commonly used strain measures expressed in the current configuration. These strains are also called Eulerian strains.



- The *nominal strain*, corresponding to  $f(\lambda_i) = \lambda_i - 1$ , can be written

$$\mathbf{e} = \mathbf{v} - \mathbf{I}. \quad (4.125)$$

- The *Hencky strain* (also called the *true strain* or the *logarithmic strain*), which corresponds to  $f(\lambda_i) = \ln(\lambda_i)$ , can be written

$$\mathbf{e} = \ln \mathbf{v}. \quad (4.126)$$

- The *Euler-Almansi strain*, corresponding to  $f(\lambda_i) = \frac{1}{2}(1 - \lambda_i^{-2})$ , can also be written

$$\mathbf{e} = \frac{1}{2} [\mathbf{I} - \mathbf{b}^{-1}]. \quad (4.127)$$

**Note:** Most FE programs, for example Abaqus and ANSYS, can create contour plots of the Eulerian nominal (engineering) strain, and the Eulerian logarithmic (true) strain. These strains are often simply called the “engineering strain” and the “true strain,” respectively.

## 4.7 Rates of Deformation

Many advanced constitutive theories for polymers includes considerations of the inherent viscoelasticity and viscoplasticity of the material. To incorporate these effects into a model it is necessary to consider the time derivative of the deformation gradient

$$\dot{\mathbf{F}} = \frac{\partial}{\partial t} \frac{\partial x_i}{\partial X_j} = \frac{\partial}{\partial X_j} \frac{\partial x_i}{\partial t} = \frac{\partial \mathbf{v}}{\partial \mathbf{X}} = \text{Grad } \mathbf{v}. \quad (4.128)$$

From this equation we can also write the time derivative of  $\mathbf{F}$  as

$$\dot{\mathbf{F}} = \frac{\partial v_i}{\partial X_j} = \frac{\partial v_i}{\partial x_k} \frac{\partial x_k}{\partial X_j} = \mathbf{I}\mathbf{F}, \quad (4.129)$$

where  $\mathbf{I}$  is a tensor quantity called the *spatial velocity gradient*:

$$\mathbf{I} \equiv \frac{\partial \mathbf{v}}{\partial \mathbf{x}} = \text{grad } \mathbf{v}. \quad (4.130)$$

It is often useful to decompose the spatial velocity gradient into a symmetric ( $\mathbf{d} = \mathbf{d}^\top$ ) and an anti-symmetric ( $\mathbf{w} = -\mathbf{w}^\top$ ) part:

$$\mathbf{l} = \mathbf{d} + \mathbf{w}, \quad (4.131)$$

where

$$\mathbf{d} = \frac{1}{2} (\mathbf{l} + \mathbf{l}^\top), \quad (4.132)$$

$$\mathbf{w} = \frac{1}{2} (\mathbf{l} - \mathbf{l}^\top). \quad (4.133)$$

The tensor  $\mathbf{d}$  is called the *rate of deformation tensor* and  $\mathbf{w}$  is called the *spin tensor*. Note that  $\mathbf{d}$  captures rates of deformation and  $\mathbf{w}$  captures rates of rotation. All three of  $\mathbf{l}$ ,  $\mathbf{d}$ , and  $\mathbf{w}$  are spatial fields.

The spatial velocity gradient can also be used to determine the time rate of change of a vector in the current configuration:

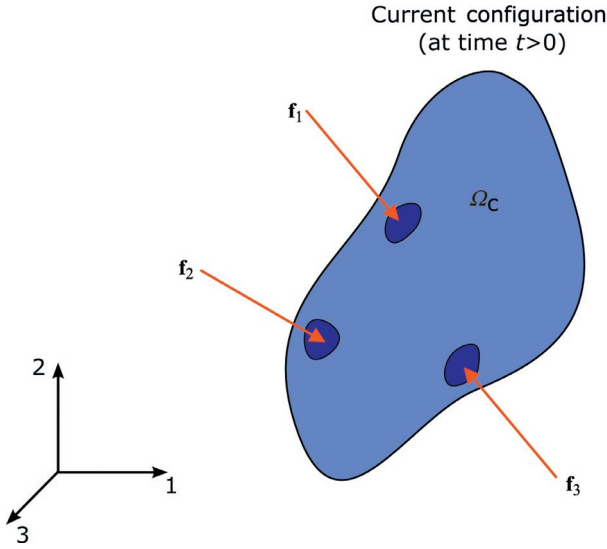
$$\mathbf{l} \, d\mathbf{x} = \mathbf{I} \mathbf{F} d\mathbf{X} = \dot{\mathbf{F}} d\mathbf{X} = \frac{D}{Dt} (d\mathbf{x}). \quad (4.134)$$

Another useful relationship is the time derivative of  $J = \det \mathbf{F}$ :

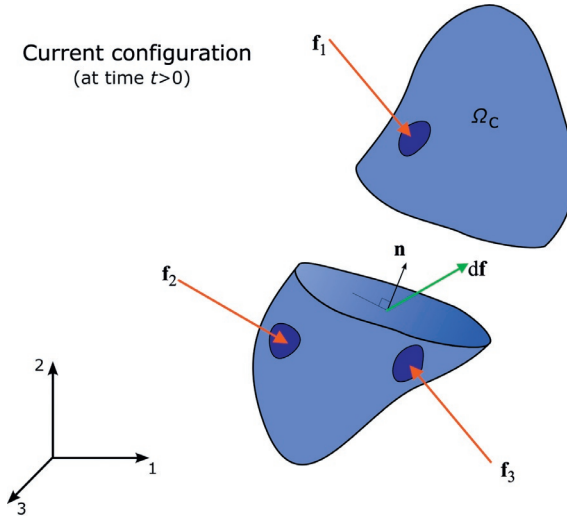
$$\frac{DJ}{Dt} = \dot{J} = J \operatorname{tr} \mathbf{l} = J \operatorname{div} \mathbf{v}. \quad (4.135)$$

## 4.8 Stress Tensors

In order to answer real-life polymer mechanics questions, such as: how much will a polymer component deform when subjected to an external load field, it is necessary to introduce the concept of mechanical stress. To do this we will consider a general body exposed to external forces on its surface as illustrated in [Figure 4.7](#). This figure shows the configuration and acting forces on the body at time  $t$ . Now let us perform a virtual cut along a plane through the body, see [Figure 4.8](#). In order to satisfy force equilibrium for each of the two parts of the body there must be internal surface forces on the cut plane. The magnitude of the internal surface forces will depend on the direction of the cut surface (specified by the normal  $\mathbf{n}$  to the surface) and the location  $\mathbf{x}$  of the force:



**Figure 4.7** Schematic figure showing a body in its deformed configuration  $\Omega_c$  loaded by external forces, here exemplified by forces distributed over an area.



**Figure 4.8** Schematic figure of a body in its deformed configuration showing that internal forces  $d\mathbf{f}$  are needed to keep the body in equilibrium when a virtual cut is made.

$$d\mathbf{f} = d\mathbf{f}(\mathbf{x}, \mathbf{n}). \quad (4.136)$$

These surface forces can be represented as tractions  $\mathbf{t}$  (i.e., force per unit area):

$$d\mathbf{f}(\mathbf{x}, \mathbf{n}) = \mathbf{t}(\mathbf{x}, \mathbf{n}) ds, \quad (4.137)$$

which is the force acting on the surface element  $ds$ . The force at each material point can be related to a stress field by the Cauchy stress theorem [2, 3] which states that there exists a unique tensorial stress field  $\boldsymbol{\sigma}(\mathbf{x})$  that is independent of the direction of the virtual cut (specified by the normal  $\mathbf{n}$ ) and is defined by:

$$\mathbf{t}(\mathbf{x}, \mathbf{n}) = \boldsymbol{\sigma}(\mathbf{x})\mathbf{n}. \quad (4.138)$$

In this equation,  $\mathbf{t}(\mathbf{x}, \mathbf{n})$  is the Cauchy (true) surface traction, and

$$\boldsymbol{\sigma} = \sum_{i=1}^3 \sigma_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i \quad (4.139)$$

is the Cauchy stress tensor.

The traction vector and the stress tensor in Equation (4.138) can also be written in the reference configuration as

$$\mathbf{T}(\mathbf{X}, \mathbf{N}) = \mathbf{P}(\mathbf{X})\mathbf{N}, \quad (4.140)$$

where  $\mathbf{T}$  is the *nominal traction vector* (also called the *first Piola-Kirchhoff traction vector*), and  $\mathbf{P}(\mathbf{X})$  is the *nominal stress tensor* (also called the *first Piola-Kirchhoff stress tensor*) which can be written

$$\mathbf{P} = \sum_{i=1}^3 P_i \hat{\mathbf{n}}_i \otimes \hat{\mathbf{N}}_i, \quad (4.141)$$

illustrating that  $\mathbf{P}$  is a two-point tensor.

The force vector in the reference and the current configurations have to be equal:

$$\mathbf{T}(\mathbf{X}, \mathbf{N}) dS = \mathbf{t}(\mathbf{x}, \mathbf{n}) ds \quad (4.142)$$

giving

$$\mathbf{P}(\mathbf{X})\mathbf{N} dS = \boldsymbol{\sigma}(\mathbf{x})\mathbf{n} ds. \quad (4.143)$$

From Nanson's formula  $d\mathbf{s} = J\mathbf{F}^{-1}d\mathbf{S}$  we get the following relationship between the Cauchy stress tensor and the first Piola-Kirchhoff stress tensor:

$$\mathbf{P}(\mathbf{X}) = J\boldsymbol{\sigma}\mathbf{F}^{-\top}, \quad (4.144)$$

or when solved for the Cauchy stress

$$\boldsymbol{\sigma} = J^{-1}\mathbf{P}\mathbf{F}^{\top}, \quad (4.145)$$

showing that the first Piola-Kirchhoff stress tensor is not symmetric.

There are numerous other stress tensors that have been defined and used in the literature. One common stress is the Kirchhoff stress defined by:

$$\boldsymbol{\tau} = J\boldsymbol{\sigma}. \quad (4.146)$$

Another common stress is the second Piola-Kirchhoff stress  $\mathbf{S}$ . If we apply  $\mathbf{F}^{-1}$  on the Cauchy surface traction vector we get a traction vector in the reference configuration denoted by  $\tilde{\mathbf{T}}$ :

$$\tilde{\mathbf{T}} = \mathbf{F}^{-1}\mathbf{t} = \mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{n}. \quad (4.147)$$

The traction vector  $\tilde{\mathbf{T}}$  is obtained from the second Piola-Kirchhoff stress:

$$\tilde{\mathbf{T}} = \mathbf{S}\mathbf{N}. \quad (4.148)$$

The force vector in the reference and the current configurations have to be the same:

$$\mathbf{S}\mathbf{N}dS = \mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{n}ds. \quad (4.149)$$

From Nanson's formula (Equation (4.91)) we get

$$\mathbf{S} = J\mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{F}^{-\top} = \mathbf{F}^{-1}\mathbf{P}. \quad (4.150)$$

### Example: Uniaxial Loading.

To illustrate the different stress measures let us consider a uniaxial tension case in which the deformation gradient is

$$\mathbf{F} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0.7 & 0.7 \\ 0 & 0 & 0.7 \end{bmatrix}$$

and the Cauchy stress is given by

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

In this case, the first Piola-Kirchhoff stress is given by Equation (4.144):

$$\mathbf{P} = \begin{bmatrix} 0.49\sigma_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and the second Piola-Kirchhoff stress is given by Equation (4.150):

$$\mathbf{S} = \begin{bmatrix} 0.245\sigma_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

and the Kirchhoff stress is given by Equation (4.146):

$$\boldsymbol{\tau} = \begin{bmatrix} 0.985\sigma_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

This example shows that the magnitude of the different stress measures can be quite different for a finite deformation load case.

### 4.8.1 Stress Invariants

Since a stress tensor contains six or nine different components (depending on whether it is symmetric or not) it is often convenient to consider a single scalar quantity that characterizes certain aspects of the stress tensor. These scalar values are often characteristic values or invariants of the stress tensor. Since multiplying or adding two invariants leads to another invariant, there are infinitely many invariants that can be defined and used.

The perhaps easiest invariants to define, interpret, and use are the principal values of the stress tensor. As was shown in

Equation (4.83), any tensor can be uniquely decomposed into a singular value decomposition:

$$\mathbf{A} = \mathbf{Q}_1 \mathbf{\Lambda} \mathbf{Q}_2, \quad (4.151)$$

where  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  are orthogonal tensors that specify rotations, and  $\mathbf{\Lambda}$  is a diagonal tensor with three principal values. If the stress tensor is symmetric (such as the Cauchy stress and the second Piola-Kirchhoff stress), the singular value decomposition becomes the same as the eigenvalue decomposition, and the principal values coincide with the eigenvalues of the stress tensor. The principal stresses are often referred to as  $\sigma_1 > \sigma_2 > \sigma_3$ .

From Equation (4.151) it is clear that the diagonal stress matrix  $\mathbf{\Lambda}$  is given by:

$$\mathbf{\Lambda} = \mathbf{Q}_1^\top \mathbf{A} \mathbf{Q}_2^\top. \quad (4.152)$$

Hence, by premultiplying a stress tensor with a suitable rotation  $\mathbf{Q}_1^\top$  and then postmultiplying with another rotation  $\mathbf{Q}_2^\top$  the stress tensor can be transformed into diagonal form, corresponding to a purely axial stress state with only normal stresses and no shear stresses. This means that  $\sigma_1$  corresponds to the largest tensile stress, and  $\sigma_3$  corresponds to the smallest tensile stress (i.e., the largest compressive stress).

There are two other invariants of the stress tensor that are commonly used when describing the magnitude of shear stress at a point: the Mises stress and the Tresca stress. The Mises stress is defined by:

$$\sigma_M = \sqrt{\frac{1}{2} [(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]} \quad (4.153)$$

and the Tresca stress is defined by:

$$\sigma_T = \sigma_1 - \sigma_3. \quad (4.154)$$

As will be discussed in more detail in a later chapter, the maximum principal stress ( $\sigma_1$ ) is the driving force for crack nucleation and growth, and the Mises and Tresca stresses are driving viscoplastic flow.

## 4.9 Balance Laws and Field Equations

When developing constitutive equations for a material, or when solving continuum mechanics problems, it is important to obey important physical principles such as: conservation of mass, conservation of linear momentum, conservation of angular momentum, conservation of energy, and the first and second laws of thermodynamics. These principles apply to all materials and loading conditions, and give rise to balance laws and field equations that need to be satisfied.

In this context, a *balance law* is a physical rule expressed in terms of a given volume of material, and a *field equation* is a physical rule expressed at a material point.

The following sections present these physical principles in the context of continuum mechanics. Field variables that are needed for this discussion, and have not been defined yet, are listed in [Table 4.1](#). A summary of all variables that are used in this chapter is given in [Section 4.14](#).

To facilitate the upcoming discussion of the balance laws and field equations it is useful to establish the following transport theorem expressing the time derivative of a quantity inside a volume:

$$\frac{D}{Dt} \int_{\Omega_c} \phi \, dv = \int_{\Omega_c} (\dot{\phi} + \phi \operatorname{tr} \mathbf{I}) \, dv. \quad (4.155)$$

In this equation,  $\phi(\mathbf{x}, t)$  is any spatial scalar field,  $\Omega_c$  is the volume of the studied region at time  $t$ , and  $\mathbf{I}$  is the spatial velocity gradient. The transport theorem can be proved by the help of [Equations \(4.79\)](#) and [\(4.135\)](#) as follows:

$$\begin{aligned} \frac{D}{Dt} \int_{\Omega_c} \phi(\mathbf{x}, t) \, dv &= \frac{D}{Dt} \int_{\Omega_0} \phi J \, dV \\ &= \int_{\Omega_0} \frac{D}{Dt} [\phi J] \, dV \\ &= \int_{\Omega_0} (J\dot{\phi} + \phi\dot{J}) \, dV \end{aligned}$$



$$\begin{aligned}
 &= \int_{\Omega_c} \left( \dot{\phi} + \phi \frac{J}{j} \right) dv \\
 &= \int_{\Omega_c} (\dot{\phi} + \phi \operatorname{div} \mathbf{v}) dv.
 \end{aligned}$$

The transport theorem and its derivation is discussed in more detail in the next section.

**Table 4.1 Field Variables Used in the Balance Laws and Field Equations**

<b>Specific mass (density)</b>	
$\rho_c$	Mass density in the current configuration
$\rho_0$	Mass density in the reference configuration
<b>Temperature</b>	
$\theta_c$	Temperature in the current configuration
$\theta_0$	Temperature in the reference configuration
<b>Specific internal energy</b>	
$e_c$	Internal energy per unit current volume
$e_0$	Internal energy per unit reference volume
<b>Specific Helmholtz free energy</b>	
$\psi$	Helmholtz free energy per unit current volume
$\Psi$	Helmholtz free energy per unit reference volume
<b>Specific internal entropy</b>	
$\eta_c$	Entropy per unit current volume
$\eta_0$	Entropy per unit reference volume
<b>Specific heat flux</b>	
$\mathbf{q}$	Heat flux per unit current surface area
$\mathbf{Q}$	Heat flux per unit reference surface area
<b>Specific heat supply</b>	
$r$	Rate of heat supply per unit current volume
$R$	Rate of heat supply per unit reference volume
<b>Specific body force</b>	
$\mathbf{b}_f$	Body force per unit reference volume
$\mathbf{B}_f$	Body force per unit current volume

### 4.9.1 Conservation of Mass

This section presents the principle of mass conservation and how it applies to mechanics of continuous media. The discussion presented here is based on the simple fact that the mass of a body is given by the sum of its parts. As before, we are focusing on a body that does not exchange material with its surroundings.

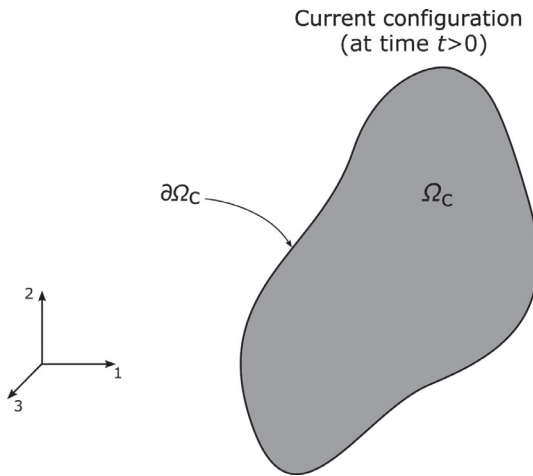
Let  $\Omega_c$  be the configuration of the body at time  $t$ , as illustrated in Figure 4.9. Since no material is entering or leaving the body, the total mass must be constant and the time derivative of the total mass has to be zero:

$$\frac{D}{Dt} \int_{\Omega_c} \rho_c(\mathbf{x}, t) dv = 0, \quad (4.156)$$

which when expressed in the reference configuration can be written

$$\frac{D}{Dt} \int_{\Omega_0} \rho_0(\mathbf{X}, t) dV = 0, \quad (4.157)$$

where  $\Omega_0$  is the configuration of the body at the initial time. These equations integrate the density over the volume of the body to get the total mass. Equations (4.156) and (4.157) are the *balance*



**Figure 4.9** Schematic figure showing the configuration of the body at time  $t$  is  $\Omega_c$ . The boundary of the body is denoted by  $\partial\Omega_c$ .

laws for mass conservation. The balance laws, which govern the response of a finite domain, can be converted to field equations that are valid at each point of the body. Let us first focus on Equation (4.157) that expresses the time rate of change of mass in the reference configuration. In this equation, the domain  $\Omega_0$  is not dependent on time and we can therefore move the time derivative operator inside the integral sign:

$$\int_{\Omega_0} \frac{D}{Dt} [\rho_0(\mathbf{X}, t)] dV = 0. \quad (4.158)$$

This equation has to be valid also for an arbitrary subdomain of  $\Omega_0$ , and hence the integrand has to be identically zero for the equation to always hold. Hence

$$\dot{\rho}_0(\mathbf{X}, t) = 0 \quad (4.159)$$

is the field equation for mass concentration in the reference configuration.

The field equation for mass conservation in the current (spatial) configuration can in a similar way be derived from Equation (4.156). In this case, the volume integral is over  $\Omega_c$  which is changing with time. We therefore cannot directly move the time-derivative operator inside the integral. To simplify the equation we will instead first perform a variable substitution to bring the integral back to the reference configuration. Specifically, from Equations (4.9) and (4.79), we chose the variable substitution  $\mathbf{x} = \mathcal{X}(\mathbf{X}, t)$ , giving  $dv = J dV$  and the integration domain is transformed from  $\Omega_c$  to  $\Omega_0$ :

$$\frac{D}{Dt} \int_{\Omega_0} \rho_c(\mathcal{X}(\mathbf{X}, t)) J dV. \quad (4.160)$$

We can now move the time-derivative operator inside the integral giving

$$\int_{\Omega_0} \frac{D}{Dt} [\rho_c(\mathcal{X}(\mathbf{X}, t)) J] dV. \quad (4.161)$$

Since this equation also has to be valid for an arbitrary subdomain of  $\Omega_0$ , the integrand has to be identically zero giving the useful equation:

$$\frac{D}{Dt} [\rho_c(\mathbf{X}, t)J] = 0. \quad (4.162)$$

The field equation for mass conservation expressed in the current configuration can be obtained by applying the chain-rule on Equation (4.161):

$$\int_{\Omega_0} [\dot{\rho}_c(\mathcal{X}(\mathbf{X}, t))J + \rho_c(\mathcal{X}(\mathbf{X}, t))\dot{J}] dV = 0. \quad (4.163)$$

By applying the reverse variable substitution,  $\mathbf{X} = \mathcal{X}^{-1}(\mathbf{x}, t)$ , this equation can be converted back to the current configuration:

$$\int_{\Omega_c} \left[ \dot{\rho}_c(\mathbf{x}, t) + \rho_c(\mathbf{x}, t) \frac{\dot{J}}{J} \right] dv = 0. \quad (4.164)$$

From Equation (4.135) we know that  $\dot{J}/J = \operatorname{div} \mathbf{x}$ , giving the field equation for mass conservation as

$$\dot{\rho}_c(\mathbf{x}, t) + \rho_c(\mathbf{x}, t) \operatorname{div} \mathbf{v} = 0. \quad (4.165)$$

From Equations (4.158) and (4.161) we also get the interesting result that

$$\rho_0(\mathbf{X}) = J\rho_c(\mathbf{x}). \quad (4.166)$$

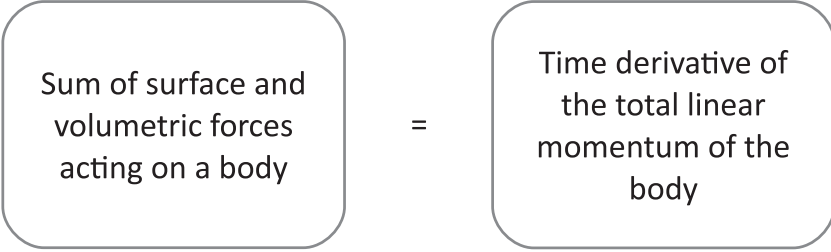
To summarize, a balance equation expresses the time derivative of an extensive quantity contained in a volume in terms of its fluxes through the boundary and the internal source of the quantity. In this case, as is commonly done for solids, we have limited our interest to closed systems where no mass can pass through the boundary, hence the mass fluxes are zero. When studying fluids, on the other hand, it is often convenient to use an open system in which material can enter and leave through the boundary of the domain of interest [16].

### 4.9.2 Balance of Linear Momentum

From the theory of rigid body dynamics we know that Newton's law of motion can be expressed as [17]:

*The force on a body is equal to the time-derivative of its linear momentum.*

In this text, we are focusing on deformable bodies undergoing large deformations, for which the balance of linear momentum can be written in the following form:



This principle can be written as a balance law expressed in the current configuration:

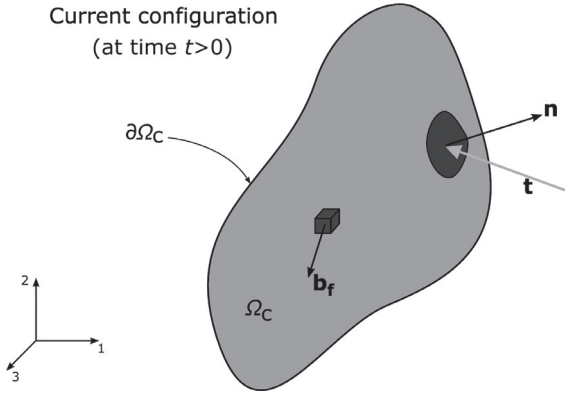
$$\int_{\partial\Omega_c} \mathbf{t} \, ds + \int_{\Omega_c} \mathbf{b}_f \, dv = \frac{D}{Dt} \int_{\Omega_c} \mathbf{v} \, \rho_c \, dv, \quad (4.167)$$

where  $\mathbf{t}$  is the distribution of Cauchy surface tractions on the boundary of the current configuration,  $ds$  is a surface area element in the current configuration,  $\partial\Omega_c$  is the surface of the body in the current configuration,  $\mathbf{v}$  is the velocity field, and  $\mathbf{b}_f$  is the vector field of body forces per unit current volume. The forces used in this equation are defined and illustrated in [Figure 4.10](#).

To convert Equation (4.167) to a field equation we will start by first applying the Cauchy stress theorem (Equation (4.138)):  $\mathbf{t} = \boldsymbol{\sigma} \mathbf{n}$ , giving

$$\int_{\partial\Omega_c} \boldsymbol{\sigma} \mathbf{n} \, ds + \int_{\Omega_c} \mathbf{b}_f \, dv = \frac{D}{Dt} \int_{\Omega_c} \mathbf{v} \, \rho_c \, dv. \quad (4.168)$$

Applying the divergence theorem (Equation (4.54)) on the first term in Equation (4.168) gives



**Figure 4.10** Schematic figure illustrating the definition of forces used in the balance of linear momentum.

$$\int_{\partial\Omega_c} \boldsymbol{\sigma} \mathbf{n} ds = \int_{\Omega_c} \operatorname{div} \boldsymbol{\sigma} dv. \quad (4.169)$$

The time-derivative of the linear momentum can be simplified further by using the same variable substitution that was used in the derivation of the field equation for mass conservation  $\mathbf{x} = \mathcal{X}(\mathbf{X}, t)$ :

$$\begin{aligned} \frac{D}{Dt} \int_{\Omega_c} \mathbf{v}(\mathbf{x}, t) \rho_c(\mathbf{x}, t) dv &= \frac{D}{Dt} \int_{\Omega_0} \mathbf{v}(\mathbf{X}, t) \rho_c(\mathbf{X}, t) J dV \quad (4.170) \\ &= \int_{\Omega_0} \left[ \dot{\mathbf{v}}(\mathbf{X}, t) \rho_c(\mathbf{X}, t) J + \mathbf{v}(\mathbf{X}, t) \frac{D}{Dt} (\rho_c(\mathbf{X}, t) J) \right] dV. \quad (4.171) \end{aligned}$$

From mass conservation (Equation (4.162)) the term  $D(\rho_c J)/Dt = 0$ , giving

$$\frac{D}{Dt} \int_{\Omega_c} \mathbf{v} \rho_c dv = \int_{\Omega_0} \dot{\mathbf{v}}(\mathbf{X}, t) \rho_c(\mathbf{X}, t) J dV = \int_{\Omega_c} \dot{\mathbf{v}} \rho_c dv. \quad (4.172)$$

The balance of linear momentum can therefore be obtained from Equations (4.168), (4.169), and (4.172) as

$$\int_{\Omega_c} [\operatorname{div} \boldsymbol{\sigma} + \mathbf{b}_f - \rho_c \dot{\mathbf{v}}] dv = 0. \quad (4.173)$$

This equation has to be valid also for an arbitrary sub-domain of  $\Omega_c$ , hence the integrand has to be identically zero giving the following field equation representation of the balance of linear momentum:

$$\operatorname{div} \boldsymbol{\sigma} + \mathbf{b}_f = \rho_c \dot{\mathbf{v}}. \quad (4.174)$$

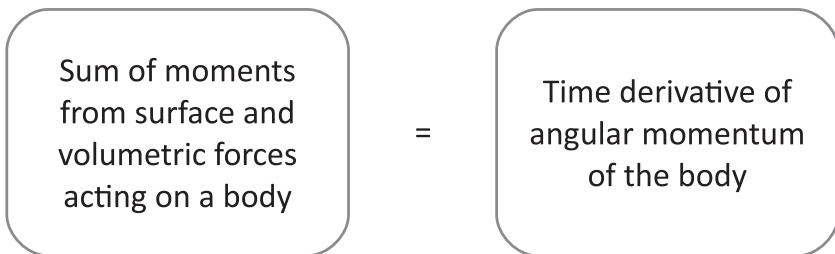
The derivation presented above can be repeated in the reference configuration. The details of the derivation are left as an exercise, the final field equation for the balance of linear momentum is:

$$\operatorname{Div} \mathbf{P} + \mathbf{B}_f = \rho_0 \dot{\mathbf{V}}. \quad (4.175)$$

### 4.9.3 Balance of Angular Momentum

The balance of angular momentum principle states that the moment that is applied on a body is equal to the time-derivative of the angular momentum. This principle is directly obtained from the linear momentum equation by taking the cross-product of both the force and the linear momentum with the position vector [17]. It is therefore to be expected that the balance of angular momentum will provide similar results to the balance of linear momentum.

For a deformable body undergoing large deformations the balance of angular momentum can be written as follows:



This can be written as a balance law expressed in the current configuration:

$$\int_{\partial\Omega_c} \mathbf{x} \times \mathbf{t} \, ds + \int_{\Omega_c} \mathbf{x} \times \mathbf{b}_f \, dv = \frac{D}{Dt} \int_{\Omega_c} \mathbf{x} \times \mathbf{v} \, \rho_c \, dv. \quad (4.176)$$

This equation can be simplified using the Cauchy stress theorem (Equation (4.138)),  $\mathbf{t} = \boldsymbol{\sigma} \mathbf{n}$ , and converting it to index form. From Equation (4.16) the index form becomes

$$\int_{\partial\Omega_c} \epsilon_{ijk} x_j \sigma_{kl} n_l ds + \int_{\Omega_c} \epsilon_{ijk} x_j b_{fk} dv = \frac{D}{Dt} \int_{\Omega_c} \epsilon_{ijk} x_j v_k \rho_c dv, \quad (4.177)$$

where  $b_{fk}$  is the  $k$ th component of the body force vector  $\mathbf{b}_f$ . The first term on the left-hand side can be simplified by applying the divergence theorem:

$$\int_{\partial\Omega_c} \epsilon_{ijk} x_j \sigma_{kl} n_l ds = \int_{\Omega_c} \epsilon_{ijk} \frac{\partial}{\partial x_l} [x_j \sigma_{kl}] dv \quad (4.178)$$

$$= \int_{\Omega_c} \left[ \epsilon_{ijk} \sigma_{kj} + \epsilon_{ijk} x_j \frac{\partial \sigma_{kl}}{\partial x_l} \right] dv. \quad (4.179)$$

The term on the right-hand side can be simplified by transforming the integration from the current to the reference configuration by the variable substitution  $\mathbf{x} = \mathcal{X}(\mathbf{X})$ :

$$\begin{aligned} \frac{D}{Dt} \int_{\Omega_c} \epsilon_{ijk} x_j v_k \rho_c dv &= \frac{D}{Dt} \int_{\Omega_0} \epsilon_{ijk} X_j V_k \rho_c J dV \\ &= \int_{\Omega_0} \left[ \epsilon_{ijk} V_j V_k \rho_c J + \epsilon_{ijk} X_j \dot{V}_k \rho_c J + \epsilon_{ijk} X_j V_k \frac{D}{Dt} (\rho_c J) \right] dV \\ &= \int_{\Omega_c} \epsilon_{ijk} x_j \dot{v}_k \rho_c dv. \end{aligned} \quad (4.180)$$

Inserting Equations (4.180) and (4.179) into Equation (4.177) gives

$$\int_{\Omega_c} \left\{ \epsilon_{ijk} \sigma_{kj} + \epsilon_{ijk} x_j \left[ \frac{\partial \sigma_{kl}}{\partial x_l} + b_{fk} - \dot{v}_k \rho_c \right] \right\} dv. \quad (4.181)$$

This equation has to be valid also for an arbitrary sub-domain of  $\Omega_c$ , hence the integrand has to be identically zero. Furthermore, the expression within the square bracket is also zero from the balance of linear momentum (Equation (4.172)).

By expanding the terms in  $\epsilon_{ijk} \sigma_{kj} = 0$ , we get the equivalent condition

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^\top. \quad (4.182)$$



Hence, the balance of angular momentum is satisfied if the balance of linear momentum is satisfied and the Cauchy stress is symmetrical.

The balance of angular momentum can also be expressed in the reference configuration using a similar derivation. The details are left as an exercise, the final field equation is:

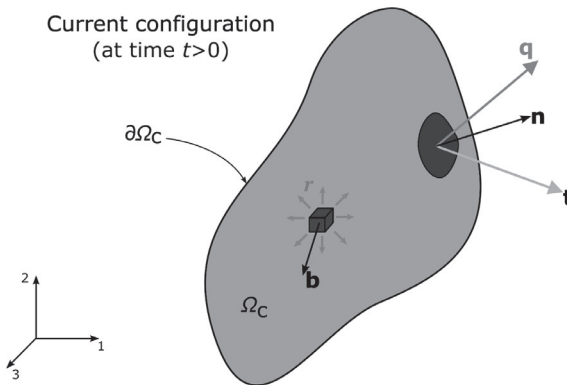
$$\mathbf{P}\mathbf{F}^T = \mathbf{F}\mathbf{P}^T, \quad (4.183)$$

where  $\mathbf{P}$  is the first Piola-Kirchhoff stress and  $\mathbf{F}$  is the deformation gradient.

#### 4.9.4 First Law of Thermodynamics

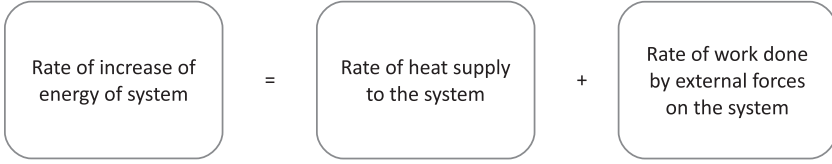
In this section, we will introduce the concept of energy conservation, specifically the first law of thermodynamics which has far reaching consequences in many fields of mechanical engineering.

The system we are studying is a closed system that can perform work and exchange heat with its surroundings, but that cannot transfer material through its boundary, see [Figure 4.11](#). The body has internal volumetric heat generation ( $r$ ) and heat flux ( $\mathbf{q}$ ) through its boundary to the surroundings. The body is also exposed to external surface tractions ( $\mathbf{t}$ ) and a volumetric body force ( $\mathbf{b}_f$ ).



**Figure 4.11** Schematic figure illustrating the definition of thermodynamic quantities in the current configuration.

The first law of thermodynamics states that [18, 19]:



This can be written as a balance law in the current configuration:

$$\begin{aligned} \frac{D}{Dt} \int_{\Omega_c} \left( e_c + \frac{\rho_c}{2} \mathbf{v} \cdot \mathbf{v} \right) dv = & \left\{ \int_{\partial\Omega_c} -\mathbf{q} \cdot \mathbf{n} ds + \int_{\Omega_c} r dv \right\} \\ & + \left\{ \int_{\partial\Omega_c} \mathbf{t} \cdot \mathbf{v} ds + \int_{\Omega_c} \mathbf{b}_f \cdot \mathbf{v} dv \right\}, \end{aligned} \quad (4.184)$$

or as a balance law in the reference configuration:

$$\begin{aligned} \frac{D}{Dt} \int_{\Omega_0} \left( e_0 + \frac{\rho_0}{2} \mathbf{V} \cdot \mathbf{V} \right) dV = & \left\{ \int_{\partial\Omega_0} -\mathbf{Q} \cdot \mathbf{N} dS + \int_{\Omega_0} R dV \right\} \\ & + \left\{ \int_{\partial\Omega_0} \mathbf{T} \cdot \mathbf{V} dS + \int_{\Omega_0} \mathbf{B}_f \cdot \mathbf{V} dV \right\}. \end{aligned} \quad (4.185)$$

In the following, we will focus on the expression for the current configuration. The first term on the left-hand side in Equation (4.184) expresses the time derivative of the internal and kinetic energies. This term can be simplified by introducing the variable substitution,  $\mathbf{x}(x) = \mathcal{X}(\mathbf{X}, t)$ , transforming the integration to the reference configuration:

$$\begin{aligned} \frac{D}{Dt} \int_{\Omega_c} \left( e_c + \frac{\rho_c}{2} v^2 \right) dv &= \int_{\Omega_0} \frac{D}{Dt} \left( e_c J + \frac{\rho_c}{2} v^2 J \right) dV \\ &= \int_{\Omega_0} \left( \dot{e}_c J + e_c \dot{J} + \frac{1}{2} \frac{D}{Dt} (\rho_c + J) v^2 + \dot{\mathbf{v}} \cdot \mathbf{v} J \right) dV. \end{aligned} \quad (4.186)$$

From Equation (4.162) we know that  $D/Dt(\rho_c J) = 0$ , giving

$$\frac{D}{Dt} \int_{\Omega_c} \left( e_c + \frac{\rho_c}{2} v^2 \right) dv = \int_{\Omega_c} [\dot{e}_c + e_c \operatorname{div} \mathbf{v} + \rho_c \dot{\mathbf{v}} \cdot \mathbf{v}] dv. \quad (4.187)$$

The rate of heat supply to the system is given by

$$\int_{\partial\Omega_c} -\mathbf{q} \cdot \mathbf{n} ds + \int_{\Omega_c} r dv = \int_{\Omega_c} [-\operatorname{div} \mathbf{q} + r] dv. \quad (4.188)$$

By using the divergence theorem (Equation (4.54)), the rate of work done by external surface forces on the system can be written

$$\begin{aligned} \int_{\partial\Omega_c} \mathbf{t} \cdot \mathbf{v} ds &= \int_{\partial\Omega_c} (\boldsymbol{\sigma} \mathbf{n}) \cdot \mathbf{v} ds = \int_{\partial\Omega_c} \sigma_{ij} n_j n_i ds \\ &= \int_{\Omega_c} \frac{\partial(\sigma_{ij} v_i)}{\partial x_j} \\ &= \int_{\Omega_c} \left( \frac{\partial \sigma_{ij}}{\partial x_j} v_i + \sigma_{ij} \frac{\partial v_i}{\partial x_j} \right) dv \\ &= \int_{\Omega_c} (\operatorname{div} \boldsymbol{\sigma} \cdot \mathbf{v} + \boldsymbol{\sigma} : \mathbf{d}) dv. \end{aligned} \quad (4.189)$$

By inserting Equations (4.187), (4.188), and (4.189) into Equation (4.184) we get the following expression

$$\begin{aligned} \int_{\Omega_c} (\boldsymbol{\sigma} : \mathbf{d} - \operatorname{div} \mathbf{q} + r - [\operatorname{div} \boldsymbol{\sigma} + \mathbf{b}_f - \rho_c \dot{\mathbf{v}}] \cdot \mathbf{v} \\ - \dot{e}_c - e_c \operatorname{div} \mathbf{v}) dv. \end{aligned} \quad (4.190)$$

From the balance of linear momentum (Equation (4.174)), this equation gives the field equation for energy conservation in the current configuration:

$$\boldsymbol{\sigma} : \mathbf{d} - \operatorname{div} \mathbf{q} + r = \dot{e}_c + e_c \operatorname{div} \mathbf{v}. \quad (4.191)$$

Using a similar derivation the corresponding field equation for energy conservation in the reference configuration can be written:

$$\mathbf{P} : \dot{\mathbf{F}} - \operatorname{Div} \mathbf{Q} + R = \dot{e}_0. \quad (4.192)$$

### 4.9.5 Second Law of Thermodynamics

The second law of thermodynamics applies to all systems and can be written in many different forms. One form suitable for the discussion in this section is that [18, 19]:

*The entropy of a thermally isolated macroscopic system never decreases.*

Entropy is a measure of the amount of energy in a system that cannot be converted to work. The unit of entropy is  $J/K$ . Through the field of statistical mechanics it be shown that entropy is also be a measure of the disorder in the system [20]. As is taught in introductory thermodynamics [21], if a system is undergoing a reversible process during which a given amount of heat is applied at a known temperature, then the change in entropy of the system is given by the heat divided by the temperature.

For a deformable body the second law of thermodynamics can for a closed system be written as the following balance law, see Figure 4.12.

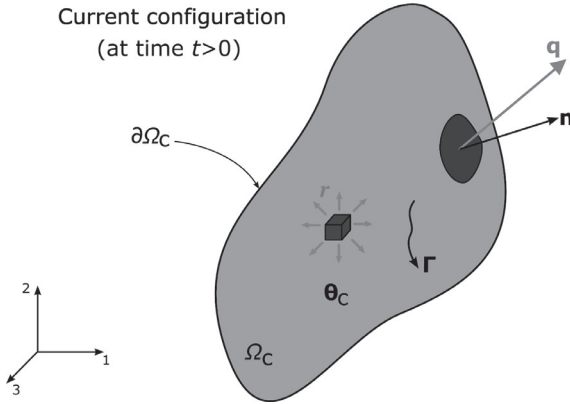


By using the nomenclature presented in Table 4.1, we can write the second law of thermodynamics as the following balance law in the current configuration:

$$\frac{D}{Dt} \int_{\Omega_c} \eta_c \, dv = \int_{\partial\Omega_c} -\frac{\mathbf{q} \cdot \mathbf{n}}{\theta_c} \, ds + \int_{\Omega_c} \frac{r}{\theta_c} \, dv + \Gamma, \quad (4.193)$$

or when expressed in the reference configuration

$$\frac{D}{Dt} \int_{\Omega_0} \eta_0 \, dV = \int_{\partial\Omega_0} -\frac{\mathbf{Q} \cdot \mathbf{N}}{\theta_0} \, dS + \int_{\Omega_0} \frac{R}{\theta_0} \, dV + \Gamma. \quad (4.194)$$



**Figure 4.12** Schematic figure showing the definition of thermodynamic quantities in the current configuration.

In these equations,  $\Gamma \geq 0$  is the entropy generation rate due to irreversible mechanisms. These equations are often referred to as the Clausius-Duhem equation.

By using the divergence theorem (Equation (4.54)), the field equation in the current configuration becomes:

$$\dot{\eta}_c + \eta_c \operatorname{div} \mathbf{v} \geq -\operatorname{div} \left( \frac{\mathbf{q}}{\theta_c} \right) + \frac{r}{\theta_c}. \quad (4.195)$$

and in the reference configuration:

$$\dot{\eta}_0 \geq -\operatorname{Div} \left( \frac{\mathbf{Q}}{\theta_0} \right) + \frac{R}{\theta_0}. \quad (4.196)$$

These equations are very useful when developing constitutive equations as will be discussed in [Section 4.11](#).

## 4.10 Energy Balance and Stress Power

The balance of mechanical effect is introduced in this section, and it is shown how that leads to the definition of stress power. To start, define the rate of work done by external forces on a body in configuration  $\Omega_c$ :

$$\mathcal{P}_{\text{ext}} = \int_{\partial\Omega_c} \mathbf{t} \cdot \mathbf{v} \, ds + \int_{\Omega_c} \mathbf{b}_f \cdot \mathbf{v} \, dv, \quad (4.197)$$

where  $\mathbf{t}$  is a surface traction,  $\mathbf{b}_f$  is the body force, and  $\mathbf{v}$  is the velocity field, all of which are expressed in the current configuration. By using the Cauchy stress theorem (Equation (4.138)) and the divergence theorem (Equation (4.54)), this equation can be written

$$\mathcal{P}_{\text{ext}} = \int_{\Omega_c} [\text{div}(\boldsymbol{\sigma}\mathbf{v}) + \mathbf{b}_f \cdot \mathbf{v}] \, dv \quad (4.198)$$

$$= \int_{\Omega_c} [\boldsymbol{\sigma} : \mathbf{l} + (\text{div } \boldsymbol{\sigma}) \cdot \mathbf{v} + \mathbf{b}_f \cdot \mathbf{v}] \, dv \quad (4.199)$$

$$= \int_{\Omega_c} \boldsymbol{\sigma} : \mathbf{d} \, dv + \int_{\Omega_c} \rho_c \dot{\mathbf{v}} \cdot \mathbf{v} \, dv \quad (4.200)$$

$$= \int_{\Omega_c} \boldsymbol{\sigma} : \mathbf{d} \, dv + \frac{D}{Dt} \int_{\Omega_c} \frac{1}{2} \rho_c \mathbf{v}^2 \, dv \quad (4.201)$$

$$= \mathcal{P}_{\text{int}} + \frac{D}{Dt} \mathcal{K}. \quad (4.202)$$

Hence, the rate of external work ( $\mathcal{P}_{\text{ext}}$ ) is equal to the rate of internal mechanical work (also called the stress-power  $\mathcal{P}_{\text{int}}$ ) plus the time derivative of the kinetic energy ( $D\mathcal{K}/Dt$ ).

From this equation we see that the stress power per unit current volume is  $\boldsymbol{\sigma} : \mathbf{d}$ , which can be converted to reference volume by Equation (4.79) giving:

$$J\boldsymbol{\sigma} : \mathbf{d}, \quad (4.203)$$

where  $J\boldsymbol{\sigma}$  is the Kirchhoff stress  $\boldsymbol{\tau}$ . From this equation, we say that  $\boldsymbol{\tau}$  and  $\mathbf{d}$  are *work conjugate* stress and deformation rate measures.

The expression for stress power can be converted to other stress measures. First, recall from Equation (4.150) that  $J\boldsymbol{\sigma} = \mathbf{P}\mathbf{F}^T$ , inserting this into Equation (4.203) gives

$$\begin{aligned} J\boldsymbol{\sigma} : \mathbf{d} &= \mathbf{P}\mathbf{F}^T : \mathbf{d} \\ &= \mathbf{P}\mathbf{F}^T : \mathbf{l} \\ &= \mathbf{P} : \mathbf{I}\dot{\mathbf{F}} \\ &= \mathbf{P} : \dot{\mathbf{F}}, \end{aligned} \quad (4.204)$$

illustrating that the first Piola-Kirchhoff stress is work conjugate to the time derivative of the deformation gradient.

The stress power per unit reference volume can also be expressed in terms of the second Piola-Kirchhoff stress  $\mathbf{S}$ . Recall from Equation (4.150) that  $\boldsymbol{\sigma} = \mathbf{F}\mathbf{S}\mathbf{F}^T/J$ , giving

$$\begin{aligned} J\boldsymbol{\sigma} : \mathbf{d} &= \mathbf{F}\mathbf{S}\mathbf{F}^T : \mathbf{d} \\ &= \mathbf{S} : \mathbf{F}^T \mathbf{d}\mathbf{F}. \end{aligned} \quad (4.205)$$

The term  $\mathbf{F}^T \mathbf{d}\mathbf{F}$  can be simplified by inserting  $\mathbf{F} = \mathbf{R}\mathbf{U}$ ,  $\mathbf{d} = \dot{\mathbf{I}} - \mathbf{w}$ ,  $\dot{\mathbf{F}} = \dot{\mathbf{R}}\mathbf{U} + \mathbf{R}\dot{\mathbf{U}}$ , and  $\dot{\mathbf{R}} = \mathbf{w}\mathbf{R}$ , giving

$$\mathbf{F}^T \mathbf{d}\mathbf{F} = \dot{\mathbf{U}}\mathbf{U} = \dot{\mathbf{E}}^G, \quad (4.206)$$

where the Green strain (Equation (4.119)) is defined by:  $\mathbf{E}^G = \frac{1}{2} [\mathbf{U}^2 - \mathbf{I}]$ , giving

$$J\boldsymbol{\sigma} : \mathbf{d} = \mathbf{S} : \dot{\mathbf{E}}^G. \quad (4.207)$$

Hence, the second Piola-Kirchhoff stress is work conjugate to the time derivative of the Green strain.

When developing constitutive equations it is important to use work conjugate stress and strain measures. The following table summarizes the three most commonly used pairs of stress and strain measures:

**Table 4.2 Work Conjugate Stress and Deformation Rate Measures**

Stress Measure	Work Conjugate Deformation Rate Measure
Kirchhoff stress $J\boldsymbol{\sigma} \equiv \boldsymbol{\tau}$	Stress rate tensor $\mathbf{d}$
First Piola-Kirchhoff stress $\mathbf{P}$	Time derivative of deformation gradient $\dot{\mathbf{F}}$
Second Piola-Kirchhoff stress $\mathbf{S}$	Time derivative of Green strain $\dot{\mathbf{E}}^G$

## 4.11 Constitutive Equations

A constitutive equation (sometimes also called a *material model* or *constitutive model*) is a relationship that describes how a material behaves, for example, the stress response for a given strain, or the heat transfer for a given temperature gradient. The constitutive equations are what distinguish the response of different materials. Everything presented so far in this chapter is valid for all materials. Next, we will discuss how the continuum mechanics framework that we have developed can be used to formulate the constitutive equations for a material. The presentation will focus on general relations that all constitutive equations need to fulfill. The following chapters of this book will demonstrate how these results can be used to develop specific material models for polymers.

There are different classes of constitutive models that can be developed and used. For example, a nonlinear elastic model is based on a stress function that only depend on the applied deformation:

$$\boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}(\mathbf{F}). \quad (4.208)$$

In this equation, we have indicated that the Cauchy stress  $\hat{\boldsymbol{\sigma}}(\mathbf{F})$  is a function of the deformation gradient. Note that this and other constitutive models can equally well be written also in terms of other stress measures, for example

$$\mathbf{P} = \hat{\mathbf{P}}(\mathbf{F}), \quad (4.209)$$

$$\mathbf{S} = \hat{\mathbf{S}}(\mathbf{F}). \quad (4.210)$$

For simplicity we will in the following focus mostly on the Cauchy stress in the derivations.

As a second example of a constitutive equation, consider a more common polymeric material characterized by a rate-dependent response (i.e., a material with a slight memory of past deformation). For this material the stress depends on both the current deformation state ( $\mathbf{F}$ ) and the rate of deformation ( $\dot{\mathbf{F}}$ ):

$$\boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}(\mathbf{F}, \dot{\mathbf{F}}). \quad (4.211)$$



In general, to specify the response of a material that is exposed to thermal and mechanical loads it is necessary to specify how the stress ( $\sigma$ ), internal energy ( $\epsilon$ ), heat flux ( $\mathbf{q}$ ), and entropy ( $\eta$ ) depend on the applied fields. To completely know the state of the material it is necessary to know the following field variables at each point:

- the velocity  $\mathbf{v}$ ,
- the stress tensor  $\sigma$ ,
- the mass density  $\rho$ ,
- the internal energy per unit current volume  $\epsilon_c$ ,
- the temperature  $\theta$ ,
- the heat flux vector  $\mathbf{q}$ , and
- the entropy  $\eta$ .

In total, by using the symmetry of  $\sigma$ , there are therefore 16 unknown field variables that need to be determined. The following equations are needed to solve for the field variables:

Governing Field Equations	Number of Equations
Conservation of mass	1
Conservation of linear momentum	3
Conservation of angular momentum	0 (already used)
First law of thermodynamics	1
Second law of thermodynamics	0 (only an inequality)

Total of 5 equations

Since there are only 5 governing field equations and there are 16 unknown field variables, more equations are needed. The missing equations are the constitutive equations which provide the following relationships.

Constitutive Equation	Number of Equations
Stress $\boldsymbol{\sigma}(\cdot)$	6
Internal energy $\epsilon_c(\cdot)$	1
Heat flux per unit volume $\mathbf{q}(\cdot)$	3
Entropy per unit volume $\eta_c(\cdot)$	1

Total of 11 equations

In summary, by combining the constitutive equations with the field equations we get 16 governing equations to solve for the 16 unknown field variables.

As mentioned, there are many different classes of constitutive equations that can be used to describe the behavior of a solid material. One of the most basic approaches is to model the material as thermoelastic and specifying constitutive equations that depend on the deformation gradient ( $\mathbf{F}$ ), the temperature ( $\theta_c$ ), and the gradient of the temperature ( $\text{grad } \theta_c$ ). This case is discussed in more detail in the next section.

#### 4.11.1 Constitutive Equations for a Thermoelastic Material

A thermoelastic material is characterized by having the heat flux vector ( $\mathbf{q}$ ), Cauchy stress ( $\boldsymbol{\sigma}$ ), internal energy per unit current volume ( $\epsilon_c$ ), and entropy per unit current volume ( $\eta_c$ ) all depend on the deformation gradient, temperature, and temperature gradient:

$$\boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}(\mathbf{F}, \theta_c, \text{grad } \theta_c), \quad (4.212)$$

$$\mathbf{q} = \hat{\mathbf{q}}(\mathbf{F}, \theta_c, \text{grad } \theta_c), \quad (4.213)$$

$$\epsilon_c = \hat{\epsilon}_c(\mathbf{F}, \theta_c, \text{grad } \theta_c), \quad (4.214)$$

$$\eta_c = \hat{\eta}_c(\mathbf{F}, \theta_c, \text{grad } \theta_c). \quad (4.215)$$

The requirements on these constitutive equations for a thermoelastic material can be further specified by using an approach that was originally developed by Truesdell et al. [7]. To derive the governing equations we will start by listing the balance laws in the reference configuration:

- linear momentum

$$\text{Div } \mathbf{P} + \mathbf{B}_f = \rho_0 \dot{\mathbf{V}}, \quad (4.175\text{-repeat})$$

- angular momentum

$$\mathbf{P}\mathbf{F}^\top = \mathbf{F}\mathbf{P}^\top, \quad (4.183\text{-repeat})$$

- first law of thermodynamics

$$\mathbf{P} : \dot{\mathbf{F}} - \text{Div } \mathbf{Q} + R = \dot{e}_0, \quad (4.192\text{-repeat})$$

- second law of thermodynamics

$$\dot{\eta}_0 \geq -\text{Div} \left( \frac{\mathbf{Q}}{\theta_0} \right) + \frac{R}{\theta_0}. \quad (4.196\text{-repeat})$$

The constitutive functions in Equations (4.212)–(4.215) need to satisfy all of these governing equations in order to be valid. These requirements can be simplified by first solve for  $R/\theta_0$  in Equation (4.196) and then inserting the result into Equation (4.196):

$$-\text{Div} \left( \frac{\mathbf{Q}}{\theta_0} \right) + \frac{\dot{e}_0}{\theta_0} - \frac{\mathbf{P} : \dot{\mathbf{F}}}{\theta_0} + \frac{\text{Div } \mathbf{Q}}{\theta_0} \geq \dot{\eta}_0. \quad (4.216)$$

The term  $-\text{Div}(\mathbf{Q}/\theta_0)$  can be simplified by expanding the divergence operator:

$$\begin{aligned} -\text{Div} \left( \frac{\mathbf{Q}}{\theta_0} \right) &= -\frac{\partial}{\partial X_i} \left( \frac{Q_i}{\theta_0} \right) \\ &= -\frac{1}{\theta_0} \frac{\partial Q_i}{\partial X_i} + \frac{Q_i}{\theta_0^2} \frac{\partial \theta_0}{\partial X_i} \\ &= -\frac{1}{\theta_0} \text{Div } \mathbf{Q} + \frac{\mathbf{Q} \cdot \text{Grad } \theta_0}{\theta_0^2}, \end{aligned} \quad (4.217)$$

which when inserted into Equation (4.216) give

$$\frac{\mathbf{Q} \cdot \text{Grad } \theta_0}{\theta_0^2} + \frac{\dot{e}_0}{\theta_0} - \frac{\mathbf{P} : \dot{\mathbf{F}}}{\theta_0} \geq \dot{\eta}_0. \quad (4.218)$$

When working with thermoelastic materials it is often more convenient to work with the Helmholtz free energy ( $\Psi$ ) instead of the internal energy ( $e_c$ ). The Helmholtz free energy per unit reference volume is defined by:

$$\Psi = e_0 - \theta_0 \eta_0. \quad (4.219)$$

From the definition of  $\Psi$  we can solve for  $\dot{e}_0$ :

$$\dot{e}_0 = \dot{\Psi} + \dot{\theta}_0 \eta_0 + \theta_0 \dot{\eta}_0. \quad (4.220)$$

Inserting Equation (4.220) into Equation (4.218) gives

$$\dot{\Psi} + \eta_0 \dot{\theta}_0 - \mathbf{P} : \dot{\mathbf{F}} + \frac{\mathbf{Q} \cdot \text{Grad } \theta_0}{\theta_0} \geq 0. \quad (4.221)$$

As shown in Equations (4.212)–(4.215), the constitutive equations for a thermoelastic material are given by the following functional forms

$$\mathbf{P} = \hat{\mathbf{P}}(\mathbf{F}, \theta_0, \text{Grad } \theta_0), \quad (4.222)$$

$$\mathbf{Q} = \hat{\mathbf{Q}}(\mathbf{F}, \theta_0, \text{Grad } \theta_0), \quad (4.223)$$

$$\Psi = \hat{\Psi}(\mathbf{F}, \theta_0, \text{Grad } \theta_0), \quad (4.224)$$

$$\eta_0 = \hat{\eta}_0(\mathbf{F}, \theta_0, \text{Grad } \theta_0). \quad (4.225)$$

Inserting Equation (4.224) into Equation (4.221) gives:

$$\left\{ \frac{\partial \hat{\Psi}}{\partial \mathbf{F}} : \dot{\mathbf{F}} + \frac{\partial \hat{\Psi}}{\partial \theta_0} \dot{\theta}_0 + \frac{\partial \hat{\Psi}}{\partial (\text{Grad } \theta_0)} \cdot \left( \frac{d(\text{Grad } \theta_0)}{dt} \right) \right\} + \hat{\eta}_0 \dot{\theta}_0 - \hat{\mathbf{P}} : \dot{\mathbf{F}} + \frac{1}{\theta_0} \hat{\mathbf{Q}} \cdot (\text{Grad } \theta_0) \geq 0, \quad (4.226)$$

which also can be written:

$$\left\{ \frac{\partial \hat{\Psi}}{\partial \mathbf{F}} - \hat{\mathbf{P}} \right\} : \dot{\mathbf{F}} + \left\{ \frac{\partial \hat{\Psi}}{\partial \theta_0} + \hat{\eta}_0 \right\} \dot{\theta}_0 + \left\{ \frac{\partial \hat{\Psi}}{\partial (\text{Grad } \theta_0)} \cdot \frac{d}{dt} (\text{Grad } \theta_0) \right\} + \frac{1}{\theta_0} \hat{\mathbf{Q}} \cdot (\text{Grad } \theta_0) \geq 0. \quad (4.227)$$

This equation has to be valid for all processes, hence for all possible values and histories of  $\mathbf{F}$ ,  $\dot{\mathbf{F}}$ ,  $\theta$ ,  $\dot{\theta}$ ,  $\text{Grad } \theta_0$ , and  $d(\text{Grad } \theta_0)/dt$ .

If we consider a process in which  $\dot{\theta}_0 = 0$  and  $d(\text{Grad } \theta_0)/dt = 0$ , then the entropy inequality (Equation (4.226)) becomes:

$$\left\{ \frac{\partial \hat{\Psi}}{\partial \mathbf{F}} - \hat{\mathbf{P}} \right\} : \dot{\mathbf{F}} + \frac{1}{\theta_0} \hat{\mathbf{Q}} \cdot (\text{Grad } \theta_0) \geq 0. \quad (4.228)$$

This equation has to be true for all  $\dot{\mathbf{F}}$ , hence the first Piola-Kirchhoff stress has to be equal to the partial derivative of the Helmholtz free energy with respect to the deformation gradient:

$$\hat{\mathbf{P}}(\mathbf{F}, \theta_0) = \frac{\partial \hat{\Psi}}{\partial \mathbf{F}}. \quad (4.229)$$

If we instead consider a process in which  $\dot{\mathbf{F}} = 0$  and  $d(\text{Grad } \theta_0)/dt = 0$ , then the entropy inequality (Equation (4.226)) becomes:

$$\left\{ \frac{\partial \hat{\Psi}}{\partial \theta_0} + \hat{\eta}_0 \right\} \dot{\theta}_0 + \frac{1}{\theta_0} \hat{\mathbf{Q}} \cdot (\text{Grad } \theta_0) \geq 0. \quad (4.230)$$

This equation has to be true for all  $\dot{\theta}_0$ , hence the entropy per unit reference volume has to be given by the partial derivative of the Helmholtz free energy with respect to the temperature:

$$\hat{\eta}_0(\mathbf{F}, \theta) = -\frac{\partial \hat{\Psi}}{\partial \theta_0}. \quad (4.231)$$

Finally, if we consider a process in which  $\dot{\mathbf{F}} = 0$  and  $\dot{\theta}_0 = 0$ , the entropy inequality (Equation (4.226)) becomes:

$$\left\{ \frac{\partial \hat{\Psi}}{\partial (\text{Grad } \theta_0)} \right\} \cdot \left( \frac{d}{dt} \text{Grad } \theta_0 \right) + \frac{1}{\theta_0} \hat{\mathbf{Q}} \cdot (\text{Grad } \theta_0) \geq 0. \quad (4.232)$$

For this to hold for any value of  $d(\text{Grad } \theta_0)/dt$ , it is clear that  $\hat{\Psi}$  cannot depend on  $\text{Grad } \theta_0$ , that is

$$\frac{\partial \hat{\Psi}}{\partial (\text{Grad } \theta_0)} = 0. \quad (4.233)$$

By inserting Equations (4.228), (4.230), and (4.232) into Equation (4.226) we finally get the entropy inequality as:

$$\hat{\mathbf{Q}} \cdot (\text{Grad } \theta_0) \geq 0, \quad (4.234)$$

which simply states that the heat has to flow in the same direction as the temperature gradient. In summary, the constitutive behavior of a thermoelastic material is completely specified by the two functions  $\hat{\Psi}(\mathbf{F}, \theta_0)$  and  $\hat{\mathbf{Q}}(\mathbf{F}, \theta_0)$ . The remaining field variables can be determined from these two functions:

$$\hat{\mathbf{P}}(\mathbf{F}, \theta_0) = \frac{\partial \hat{\Psi}(\mathbf{F}, \theta_0)}{\partial \mathbf{F}} \quad (4.228\text{-repeat})$$

$$\hat{\eta}_0(\mathbf{F}, \theta_0) = \frac{\partial \hat{\Psi}(\mathbf{F}, \theta_0)}{\partial \theta_0}. \quad (4.230\text{-repeat})$$

From these equations, we can also determine expressions for other stress measures. For example, from Equation (4.145), we get that the Cauchy stress can be determined from the Helmholtz free energy by

$$\boldsymbol{\sigma}(\mathbf{F}, \theta_0) = \frac{1}{J} \frac{\partial \hat{\Psi}(\mathbf{F}, \theta_0)}{\partial \mathbf{F}} \mathbf{F}^\top. \quad (4.235)$$

For many materials, the rate of heat flow  $\mathbf{Q}$  is often modeled using Fourier's equation of heat conduction [18]:

$$\mathbf{Q} = \mathbf{K}(\mathbf{F}, \theta_0)(\text{Grad } \theta_0). \quad (4.236)$$

If the heat conductivity tensor  $\mathbf{K}$  is positive definite then the entropy inequality (Equation (4.233)) is always satisfied.

There are many other classes of material models that can be defined and used. The following two examples present two simple cases. More advanced models are discussed in the following chapters.

**Example**

An adiabatic thermoelastic material is defined by having no heat flux ( $\mathbf{q} = 0$ ), and having the Cauchy stress ( $\boldsymbol{\sigma}$ ), internal energy per unit volume ( $e_c$ ), and entropy per unit volume ( $\eta_c$ ) only depend on the deformation gradient and the temperature

$$\boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}(\mathbf{F}, \theta), \quad (4.237)$$

$$e_c = \hat{e}_c(\mathbf{F}, \theta), \quad (4.238)$$

$$\eta_c = \hat{\eta}_c(\mathbf{F}, \theta). \quad (4.239)$$

For this material, the second law of thermodynamics is satisfied if

$$\hat{\mathbf{P}}(\mathbf{F}, \theta) = \frac{\partial \Psi(\mathbf{F}, \theta)}{\partial \mathbf{F}}, \quad (4.240)$$

$$\hat{\eta}_0(\mathbf{F}, \theta) = -\frac{\Psi(\mathbf{F}, \theta)}{\theta_0}. \quad (4.241)$$

That is, for this material, only one thermoelastic function is needed.

**Example**

An isothermal thermoelastic material is defined by zero heat flux and entropy generation, and having Cauchy stress ( $\boldsymbol{\sigma}(\mathbf{F})$ ), internal energy ( $e_c$ ) only depend on  $\mathbf{F}$ . For this material, the second law of thermodynamics is satisfied if

$$\hat{\mathbf{P}}(\mathbf{F}) = \frac{\partial \Psi(\mathbf{F})}{\partial \mathbf{F}} = \frac{\partial e_c(\mathbf{F})}{\partial \mathbf{F}}. \quad (4.242)$$

In this case, the Helmholtz free energy and the internal energy become equal.

**4.12 Observer Transformation**

The functions  $\hat{\boldsymbol{\sigma}}(\cdot)$ ,  $\hat{\mathbf{q}}(\cdot)$ ,  $\hat{e}_c(\cdot)$ , and  $\hat{\eta}_c(\cdot)$  cannot take arbitrary forms. One of the requirements on the constitutive functions is that they satisfy *material frame indifference*. That is, if we first

apply a deformation field specified by  $\mathbf{F}$  and then apply a rotation  $\mathbf{Q}$ , then the combined deformation

$$\mathbf{x}^* \equiv \mathbf{Q}\mathbf{F}\mathbf{X} = \mathbf{Q}\mathbf{x} \quad (4.243)$$

should not change the specific internal energy or the specific entropy:

$$\hat{e}_c(\mathbf{Q}\mathbf{F}) = \hat{e}_c(\mathbf{F}), \quad (4.244)$$

$$\hat{\eta}_c(\mathbf{Q}\mathbf{F}) = \hat{\eta}_c(\mathbf{F}). \quad (4.245)$$

These equations must hold for all nonsingular deformation gradients  $\mathbf{F}$  and for all proper orthogonal rotations  $\mathbf{Q}$ . If we chose  $\mathbf{Q} = \mathbf{R}^\top$  then we get the following necessary condition on  $\hat{e}_c(\cdot)$  and  $\hat{\eta}_c(\cdot)$ :

$$\hat{e}_c(\mathbf{R}^\top \mathbf{F}) = \hat{e}_c(\mathbf{U}) = \hat{e}_c(\mathbf{F}), \quad (4.246)$$

$$\hat{\eta}_c(\mathbf{R}^\top \mathbf{F}) = \hat{\eta}_c(\mathbf{U}) = \hat{\eta}_c(\mathbf{F}). \quad (4.247)$$

Now if we instead insert  $\mathbf{F} = \mathbf{R}\mathbf{U}$  into Equations (4.246) and (4.247) we get the both necessary and sufficient conditions:

$$\hat{e}_c(\mathbf{F}) = \hat{e}_c(\mathbf{U}), \quad (4.248)$$

$$\hat{\eta}_c(\mathbf{F}) = \hat{\eta}_c(\mathbf{U}). \quad (4.249)$$

Hence, to fulfill frame-indifference the functions  $\hat{e}_c(\cdot)$  and  $\hat{\eta}_c(\cdot)$  cannot depend directly on  $\mathbf{F}$  but only on  $\mathbf{U}$ .

If we, as before, after deforming the body apply a proper rotation  $\mathbf{Q}$ , then the resulting surface traction vector and the heat flux vectors need to be rotated as well:

$$\mathbf{t}^* \equiv \hat{\mathbf{t}}(\mathbf{Q}\mathbf{F}) = \mathbf{Q}\hat{\mathbf{t}}(\mathbf{F}), \quad (4.250)$$

$$\mathbf{q}^* \equiv \hat{\mathbf{q}}(\mathbf{Q}\mathbf{F}) = \mathbf{Q}\hat{\mathbf{q}}(\mathbf{F}). \quad (4.251)$$

Recall from the Cauchy stress theorem (Equation (4.138)) that  $\mathbf{t} = \boldsymbol{\sigma}\mathbf{n}$ , which here gives

$$\mathbf{t}^* = \boldsymbol{\sigma}^* \mathbf{n}^* = \mathbf{Q}\mathbf{t} = \mathbf{Q}\boldsymbol{\sigma}\mathbf{n}, \quad (4.252)$$

but  $\mathbf{n}^* = \mathbf{Q}\mathbf{n}$  giving

$$\mathbf{t}^* = \boldsymbol{\sigma}^* \mathbf{Q}\mathbf{n} = \mathbf{Q}\boldsymbol{\sigma}\mathbf{n}. \quad (4.253)$$



This has to hold for any vector  $\mathbf{n}$ , hence

$$\boldsymbol{\sigma}^* = \mathbf{Q}\boldsymbol{\sigma}\mathbf{Q}^\top. \quad (4.254)$$

In summary, to satisfy the requirement of frame indifference the stress and heat flux need to have the following forms

$$\hat{\boldsymbol{\sigma}}(\mathbf{Q}\mathbf{F}) = \mathbf{Q}\hat{\boldsymbol{\sigma}}(\mathbf{F})\mathbf{Q}^\top, \quad (4.255)$$

$$\hat{\mathbf{q}}(\mathbf{Q}\mathbf{F}) = \mathbf{Q}\hat{\mathbf{q}}(\mathbf{F}). \quad (4.256)$$

These equations have to be valid for any arbitrary proper orthogonal  $\mathbf{Q}$ . If we chose  $\mathbf{Q} = \mathbf{R}^\top$  then we get a necessary condition for Equations (4.255) and (4.256) to be true:

$$\hat{\boldsymbol{\sigma}}(\mathbf{R}^\top\mathbf{F}) = \hat{\boldsymbol{\sigma}}(\mathbf{U}) = \mathbf{R}^\top\hat{\boldsymbol{\sigma}}(\mathbf{F})\mathbf{R}, \quad (4.257)$$

$$\hat{\mathbf{q}}(\mathbf{R}^\top\mathbf{F}) = \hat{\mathbf{q}}(\mathbf{U}) = \mathbf{R}^\top\hat{\mathbf{q}}(\mathbf{F}). \quad (4.258)$$

By inserting  $\mathbf{F} = \mathbf{R}\mathbf{U}$  into Equations (4.255) and (4.256) we get the both necessary and sufficient conditions:

$$\hat{\boldsymbol{\sigma}}(\mathbf{Q}\mathbf{R}\mathbf{U}) = \mathbf{Q}\hat{\boldsymbol{\sigma}}(\mathbf{R}\mathbf{U})\mathbf{Q}^\top \quad (4.259)$$

$$\Rightarrow \mathbf{Q}\mathbf{R}\hat{\boldsymbol{\sigma}}(\mathbf{U})\mathbf{R}^\top\mathbf{Q}^\top = \mathbf{Q}\hat{\boldsymbol{\sigma}}(\mathbf{F})\mathbf{Q}^\top \quad (4.260)$$

$$\Rightarrow \hat{\boldsymbol{\sigma}}(\mathbf{F}) = \mathbf{R}\hat{\boldsymbol{\sigma}}(\mathbf{U})\mathbf{R}^\top \quad (4.261)$$

and

$$\hat{\mathbf{q}}(\mathbf{Q}\mathbf{R}\mathbf{U}) = \mathbf{Q}\hat{\mathbf{q}}(\mathbf{R}\mathbf{U}) \quad (4.262)$$

$$\Rightarrow \mathbf{Q}\mathbf{R}\hat{\mathbf{q}}(\mathbf{U}) = \mathbf{Q}\hat{\mathbf{q}}(\mathbf{F}) \quad (4.263)$$

$$\Rightarrow \hat{\mathbf{q}}(\mathbf{F}) = \mathbf{R}\hat{\mathbf{q}}(\mathbf{U}). \quad (4.264)$$

That is, the Cauchy stress cannot directly depend on  $\mathbf{F}$ , but has to have the following form

$$\hat{\boldsymbol{\sigma}}(\mathbf{F}) = \mathbf{R}\hat{\boldsymbol{\sigma}}(\mathbf{U})\mathbf{R}^\top \quad (4.265)$$

to satisfy material frame indifference. Similarly, the heat flux vector has to be given by

$$\hat{\mathbf{q}}(\mathbf{F}) = \mathbf{R}\hat{\mathbf{q}}(\mathbf{U}). \quad (4.266)$$

These restrictions on the constitutive response can be formulated in a more general observer transformation. If we have two observers inspecting a mechanical experiment or deformation event then the measured distances between points should not depend on the locations of the observers. The two observers can be specified by their coordinate systems which are related by:

$$\mathbf{x}' = \mathbf{c} + \mathbf{Q}\mathbf{x}, \quad (4.267)$$

where  $\mathbf{c}$  is a displacement and  $\mathbf{Q}$  is a proper orthogonal tensor (a rotation).

If one observer determines that the deformation gradient is  $\mathbf{F}$  at one specific location, then the second observer should see the deformation gradient  $\mathbf{QF}$  at the same location. If we now consider a scalar field that is  $\phi$  when referred to by the first observer, and as  $\phi'$  when referred to by the second observer then

$$\phi'(\mathbf{x}) = \phi(\mathbf{x}). \quad (4.268)$$

If we have a vector field that the first observer refers to as  $\mathbf{u}$  and the second observer refers to as  $\mathbf{u}'$  then

$$\mathbf{u}'(\mathbf{x}') = \mathbf{Q}\mathbf{u}(\mathbf{x}). \quad (4.269)$$

Finally, if we have a tensor field that the first observer refers to as  $\mathbf{A}$  and the second observer refers to as  $\mathbf{A}'$ , and if  $\mathbf{n}$  and  $\mathbf{n}'$  are the representations of an arbitrary vector from the two observers, then

$$\mathbf{A}'\mathbf{n}' = \mathbf{Q}(\mathbf{A}\mathbf{n}), \quad (4.270)$$

but since  $\mathbf{n}' = \mathbf{Q}\mathbf{n}$  we also get

$$\mathbf{A}'\mathbf{Q}\mathbf{n} = \mathbf{Q}\mathbf{A}\mathbf{n}. \quad (4.271)$$

This must hold for any  $\mathbf{n}$ , hence  $\mathbf{A}'\mathbf{Q} = \mathbf{Q}\mathbf{A}$  which is equivalent to

$$\mathbf{A}' = \mathbf{Q}\mathbf{A}\mathbf{Q}^\top. \quad (4.272)$$

If the scalar, vector, or tensor field satisfies these conditions then they are said to be *objective*, or invariant to observer transformation.

### 4.12.1 Objective Rates

To demonstrate the time-derivative of second-order tensors consider a case where the basis vectors in the current spatial configuration rotate as a function of time with respect to the reference configuration:

$$\hat{\mathbf{n}}_i = \mathbf{Q}(t)\hat{\mathbf{N}}_i. \quad (4.273)$$

Then  $\dot{\hat{\mathbf{n}}}_i = \dot{\mathbf{Q}}\hat{\mathbf{N}}_i = \dot{\mathbf{Q}}\mathbf{Q}^\top \hat{\mathbf{n}}_i \equiv \boldsymbol{\Omega}\hat{\mathbf{n}}_i$ . Consider a tensor  $\boldsymbol{\sigma} = \sigma_{ij}\hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_j$ . From the chain rule, the time derivative of this tensor can be written

$$\dot{\boldsymbol{\sigma}} = \dot{\sigma}_{ij}\hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_j + \sigma_{ij}\dot{\hat{\mathbf{n}}}_i \otimes \hat{\mathbf{n}}_j + \sigma_{ij}\hat{\mathbf{n}}_i \otimes \dot{\hat{\mathbf{n}}}_j. \quad (4.274)$$

Define  $\hat{\boldsymbol{\sigma}} \equiv \dot{\sigma}_{ij}\hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_j$  to be the co-rotational rate, giving

$$\dot{\boldsymbol{\sigma}} = \hat{\boldsymbol{\sigma}} + \sigma_{ij}\boldsymbol{\Omega}\hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_j + \sigma_{ij}\hat{\mathbf{n}}_i \otimes (\boldsymbol{\Omega}\hat{\mathbf{n}}_j), \quad (4.275)$$

which also can be written

$$\dot{\boldsymbol{\sigma}} = \hat{\boldsymbol{\sigma}} + \boldsymbol{\Omega}\boldsymbol{\sigma} + \boldsymbol{\sigma}\boldsymbol{\Omega}^\top. \quad (4.276)$$

If we are spinning with the material ( $\boldsymbol{\Omega} = \mathbf{W}$ ) this becomes:

$$\dot{\boldsymbol{\sigma}} = \hat{\boldsymbol{\sigma}} + \mathbf{W}\boldsymbol{\sigma} + \boldsymbol{\sigma}\mathbf{W}^\top, \quad (4.277)$$

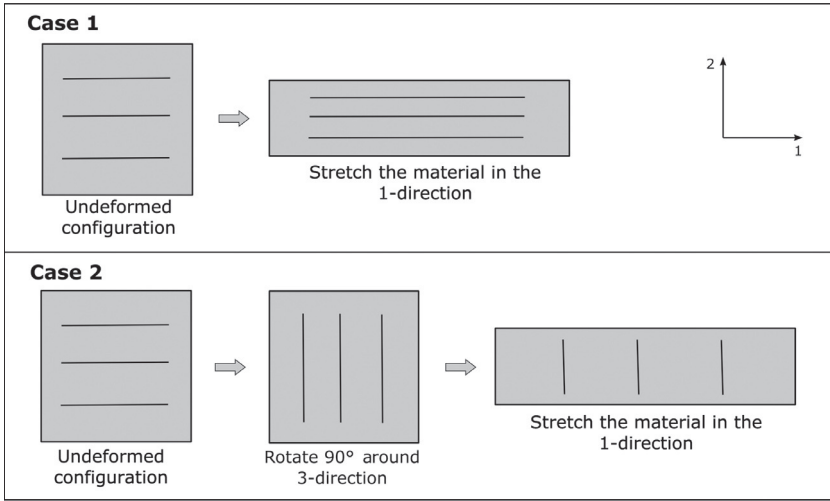
where

$$\hat{\boldsymbol{\sigma}} = \dot{\boldsymbol{\sigma}} - \mathbf{W}\boldsymbol{\sigma} - \boldsymbol{\sigma}\mathbf{W}^\top \quad (4.278)$$

is a commonly used rate definition called the Jaumann rate.

## 4.13 Material Symmetry

Consider a case in which a body is exposed to a given deformation  $\mathbf{F}$  resulting in a stress state  $\boldsymbol{\sigma}$ . Now, consider another body that is identical to the first except that it has been rotated  $90^\circ$  before the deformation was applied, as is shown in [Figure 4.13](#). For some materials, the initial  $90^\circ$  prerotation (i.e., change of reference configuration) does not influence the final stress state  $\boldsymbol{\sigma}$ .



**Figure 4.13** Example of material symmetry during deformation.

These materials are said to have material symmetry with respect to  $90^\circ$  rotation.

If the material response is invariant to any rotation then the material is said to be *isotropic*. A material that is not invariant to arbitrary rotations will have different properties in different directions and is called *anisotropic*.

In mathematical terms, if  $\mathbf{P}$  is the mapping from the initial configuration to an alternative configuration,  $\mathbf{F}$  is the deformation gradient, and the stress is independent of the initial mapping:

$$\boldsymbol{\sigma} = \hat{\boldsymbol{\sigma}}(\mathbf{F}) = \hat{\boldsymbol{\sigma}}(\mathbf{FP}), \quad (4.279)$$

then the mapping  $\mathbf{P}$  is said to be a *symmetry group* of the material. The use of symmetry groups is discussed in more detail in Chapter 5.

## 4.14 List of Symbols

The following is a list of the variables that are used in this chapter and the rest of the book. The nomenclature is similar to various recent text books, particular [2]. Direct notation is used throughout, with the same conventions as in [1, 3, 7].

Symbol	Description	Definition
$\mathbf{b}_f$	Body force per unit current volume	—
$\mathbf{B}_f$	Body force per unit reference volume	—
$\mathbf{b}$	Left Cauchy-Green tensor (spatial configuration)	$\mathbf{b} = \mathbf{F}\mathbf{F}^\top$
$\mathbf{C}$	Right Cauchy-Green tensor (reference configuration)	$\mathbf{C} = \mathbf{F}^\top\mathbf{F}$
$\mathbf{d}$	Rate of deformation tensor (spatial configuration)	$\mathbf{l} = \mathbf{d} + \mathbf{w}$
$\mathbf{e}$	Strain tensor (spatial configuration)	$\mathbf{e} = \hat{\mathbf{e}}(\mathbf{v})$
$e_0$	Internal energy per unit reference volume	—
$e_c$	Internal energy per unit current volume	—
$\eta_c$	Entropy per unit current volume	—
$\eta_0$	Entropy per unit reference volume	—
$\mathbf{E}$	Strain tensor (reference configuration)	$\mathbf{E} = \hat{\mathbf{E}}(\mathbf{U})$
$\mathbf{f}$	Force vector (spatial configuration)	$d\mathbf{f} = \mathbf{T}d\mathbf{s}$
$\mathbf{F}$	Force vector (reference configuration)	$d\mathbf{F} = \mathbf{T}d\mathbf{S}$
$\mathbf{F}$	Deformation Gradient (two-point tensor)	$\frac{\partial \mathbf{x}}{\partial \mathbf{X}}$
$J$	Jacobian determinant	—
$\mathcal{K}$	Kinetic energy of the body	—
$\mathbf{l}$	Spatial velocity gradient (spatial configuration)	$\mathbf{l} = \text{grad } \mathbf{v}$
$\mathbf{n}_i$	Eigenvectors of $\mathbf{v}$ (spatial configuration)	—
$\mathbf{N}_i$	Eigenvectors of $\mathbf{U}$ (reference configuration)	—
$\Omega_0$	Volume of body in reference configuration	—

$\Omega_c$	Volume of body in spatial configuration	—
$\partial\Omega_0$	Surface area of body in reference configuration	—
$\partial\Omega_c$	Surface area of body in the current configuration	—
<b>P</b>	First Piola-Kirchhoff stress (two-point tensor)	<b>T = PN</b>
$\mathcal{P}_{\text{ext}}$	External work done on the body	—
$\mathcal{P}_{\text{int}}$	Internal work done on the body	—
$\Psi$	Helmholtz free energy per unit current volume	$\Psi = e_0 - \theta_0 \eta_0$
$\psi$	Helmholtz free energy per unit reference volume	—
<b>q</b>	Heat flux vector per unit current area	—
$q_n$	Heat flux per unit current surface area	—
<b>Q</b>	Heat flux vector per unit reference area	—
$Q_n$	Heat flux per unit reference surface area	—
$\rho_0$	Mass density in reference configuration	—
$\rho_c$	Mass density in spatial configuration	—
$r$	Rate of heat supply per unit current volume	—
$R$	Rate of heat supply per unit reference volume	—
<b>R</b>	Rotation tensor (two-point tensor)	<b>F = RU = vR</b>
$\sigma$	Cauchy stress in spatial configuration	—
$s$	Area element in spatial configuration	—
	Area element in reference configuration	—
<b>S</b>	Second Piola-Kirchhoff stress (reference configuration)	<b>S = F<sup>-1</sup>σF<sup>-T</sup></b>
$\sigma_i$	Principal Cauchy stresses (spatial configuration)	—

$\sigma_M$	Mises stress	—
$\sigma_T$	Tresca stress	—
$\theta_0$	Temperature (reference configuration)	—
$\theta_c$	Temperature (spatial configuration)	—
$\mathbf{t}$	Cauchy surface traction (spatial configuration)	$\mathbf{t} = \boldsymbol{\sigma} \mathbf{n}$
$\mathbf{T}$	Nominal surface traction (reference configuration)	$\mathbf{T} = \mathbf{P} \mathbf{N}$
$\boldsymbol{\tau}$	Kirchhoff stress (spatial configuration)	$\boldsymbol{\tau} = J \boldsymbol{\sigma}$
$\mathbf{U}$	Right stretch tensor (reference configuration)	$\mathbf{F} = \mathbf{R} \mathbf{U}$
$\mathbf{v}$	Left stretch tensor (spatial configuration)	$\mathbf{F} = \mathbf{v} \mathbf{R}$
$\mathbf{v}$	Velocity of a material point in the current configuration	$\frac{d\mathbf{x}}{dt}$
$\mathbf{V}$	Velocity of a material point in the reference configuration)	$\frac{d\mathbf{x}}{dt}$
$v$	Volume element in spatial configuration	—
$V$	Volume element in reference configuration	—
$\mathbf{w}$	Spin tensor (spatial configuration)	$\mathbf{l} = \mathbf{d} + \mathbf{w}$
$\mathbf{x}$	Position vector in spatial configuration	—
$\mathbf{X}$	Position vector in reference configuration	—

## 4.15 Exercises

1. What is the difference between a Lagrangian and an Eulerian reference frame?
2. How many different stress and strain measures can be defined? Which stress and strain measure is your favorite?
3. Use an applied math software to calculate the singular value decomposition of the following matrix:

$$\mathbf{F} = \begin{bmatrix} 1.6 & 0.15 & 0 \\ 0.1 & 1.1 & 0.1 \\ 0.3 & 0.05 & 0.8 \end{bmatrix}.$$

4. Use the definition of a dyad (Equation (4.18)) to prove the following equations:

$$(\mathbf{a} \otimes \mathbf{b})(\mathbf{c} \otimes \mathbf{d}) = (\mathbf{b} \cdot \mathbf{c})(\mathbf{a} \otimes \mathbf{d}), \quad (4.22\text{-repeat})$$

$$(\mathbf{a} \otimes \mathbf{b})\mathbf{A} = \mathbf{a} \otimes (\mathbf{A}^\top \mathbf{b}), \quad (4.23\text{-repeat})$$

$$\mathbf{A}(\mathbf{a} \otimes \mathbf{b}) = (\mathbf{A}\mathbf{a}) \otimes \mathbf{b}. \quad (4.24\text{-repeat})$$

5. Calculate the trace, determinant, invariants ( $I_1, I_2, I_3$ ), deviatoric, volumetric, distortional, and dilatational parts of the following tensor:

$$\mathbf{F} = \begin{bmatrix} 1.3 & 0.25 & 0 \\ 0.2 & 1.1 & 0.1 \\ 0 & 0.05 & 1.1 \end{bmatrix}.$$

6. Typically a second-order tensor is said to have the invariants  $I_1, I_2$ , and  $I_3$ . Is it possible to define other invariants of a second-order tensor?
7. What deformation gradients have the same left and right Cauchy-Green tensors?
8. Calculate the natural logarithm of the following deformation gradient using Equation (4.45):

$$\mathbf{F} = \begin{bmatrix} 2.1 & 0.25 & 0 \\ 0.25 & 1.1 & 0.1 \\ 0 & 0.1 & 0.9 \end{bmatrix}.$$

9. A material point in a body is exposed to the following deformation gradient:

$$\mathbf{F} = \begin{bmatrix} 2.6 & 0.15 & 0 \\ 0.10 & 1.1 & 0.1 \\ 0 & 0.1 & 0.9 \end{bmatrix}.$$

- (a) Calculate the rotation tensor  $\mathbf{R}$ , the right stretch tensor  $\mathbf{U}$ , the left stretch tensor  $\mathbf{v}$ , the right



Cauchy-green tensor  $\mathbf{C}$ , the left Cauchy-Green tensor  $\mathbf{b}$ , and the spectral representation of  $\mathbf{F}$ .

- (b) Calculate the following Lagrangian strain: the Green strain, the Hencky strain, the Biot strain, and the Almansi strain.
- (c) Calculate the following Eulerian strains: the Euler-Almansi strain and the Hencky strain.

10. Consider two coordinate systems defined by:

$$\begin{cases} \hat{\mathbf{e}}_1 &= \frac{1}{\sqrt{2}} [1.0\hat{\mathbf{e}}_x + 1.0\hat{\mathbf{e}}_y] \\ \hat{\mathbf{e}}_2 &= \frac{1}{\sqrt{2}} [1.0\hat{\mathbf{e}}_y + 1.0\hat{\mathbf{e}}_z] \\ \hat{\mathbf{e}}_3 &= \frac{1}{\sqrt{3}} [1.0\hat{\mathbf{e}}_x - 1.0\hat{\mathbf{e}}_y + 1.0\hat{\mathbf{e}}_z] \end{cases}$$

and

$$\begin{cases} \hat{\mathbf{e}}'_1 &= \frac{1}{\sqrt{2}} [2.0\hat{\mathbf{e}}_x - 1.0\hat{\mathbf{e}}_y] \\ \hat{\mathbf{e}}'_2 &= \frac{1}{\sqrt{2}} [-1.0\hat{\mathbf{e}}_y + 1.0\hat{\mathbf{e}}_z] \\ \hat{\mathbf{e}}'_3 &= \frac{1}{3} [-1.0\hat{\mathbf{e}}_x - 2.0\hat{\mathbf{e}}_y - 2.0\hat{\mathbf{e}}_z] \end{cases}.$$

Calculate the rotation tensor  $\mathbf{Q}$  defined in Equation (4.55).

11. A polymer melt is exposed to the following deformation history for times  $t \in [0, 1]$ :

$$\begin{cases} x_1 = X_1 e^t, \\ x_2 = X_2 t, \\ x_3 = X_3. \end{cases}$$

Determine:

- (a) the deformation gradient  $\mathbf{F}$ ,
- (b) the velocity gradient  $\mathbf{L}$ ,
- (c) the components of the velocity of a material point at time  $t$ ,
- (d)  $\text{Grad } \mathbf{v}$  and  $\text{grad } \mathbf{v}$ , and
- (e)  $\text{Div } \mathbf{v}$  and  $\text{div } \mathbf{v}$ .

12. A test sample is deformed in simple shear:

$$\begin{cases} x_1 = X_1 + ktX_2 \\ x_2 = X_2 \\ x_3 = X_3 \end{cases} .$$

Calculate:

- (a) the strain tensor  $\mathbf{E} = \ln \mathbf{U}$ ,  
 (b) the strain-rate tensor  $\dot{\mathbf{E}}$ , and  
 (c) the co-rotational strain-rate tensor  $\dot{\mathbf{E}}^{\circ}$ :

$$\dot{\mathbf{E}}^{\circ} = \dot{\mathbf{E}} - \mathbf{W}\mathbf{E} - \mathbf{E}\mathbf{W}^{\top} .$$

13. In simple shear defined by

$$\mathbf{F} = \begin{bmatrix} 1 & \gamma & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} ,$$

show that  $\mathbf{U}$  is given by:

$$\mathbf{U} = \frac{1}{\sqrt{4 + \gamma^2}} \begin{bmatrix} 2 & \gamma & 0 \\ \gamma & 2 + \gamma^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} ,$$

show that  $\mathbf{v}$  is given by:

$$\mathbf{v} = \frac{1}{\sqrt{4 + \gamma^2}} \begin{bmatrix} 2 + \gamma^2 & \gamma & 0 \\ \gamma & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} ,$$

show that  $\mathbf{R}$  is given by:

$$\mathbf{R} = \frac{1}{\sqrt{4 + \gamma^2}} \begin{bmatrix} 2 & \gamma & 0 \\ -\gamma & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} ,$$

and finally show that the engineering shear strain  $\varepsilon_{12}$  (that is used by most FE software) is given by

$$\varepsilon_{12} = \frac{\gamma}{\sqrt{1 + \gamma^2/4}} .$$

14. Show that the largest engineering shear strain NE12 that Abaqus will predict for simple shear is 2.0.

15. Two symmetric tensors are *coaxial* if they have the same principal axes. Prove that Cauchy stress tensor  $\boldsymbol{\sigma}$  and the left Cauchy-Green tensor  $\mathbf{b} = \mathbf{F}\mathbf{F}^T$  are coaxial.
16. How can you check if the first law of thermodynamic is satisfied in an FE simulation?
17. What deformation rate measure is work conjugate to the Kirchhoff stress?
18. What deformation rate measure is work conjugate to the Cauchy stress?
19. How many governing field equations can be formulated from the conservation laws? How many unknown field variables are there? and What are the missing equations?

## References

- [1] M.E. Gurtin, *An Introduction to Continuum Mechanics*, Academic Press, Inc., New York, NY, 1981.
- [2] G.A. Holzapfel, *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*, John Wiley & Sons, New York, 2000.
- [3] M. Šilhavý, *The Mechanics and Thermodynamics of Continuous Media*, Springer-Verlag, Berlin, 1997.
- [4] R.W. Ogden, *Non-Linear Elastic Deformations*, Dover Publications, New York, 1997.
- [5] A.J.M. Spencer, *Continuum Mechanics*, Dover Publications, New York, 2004.
- [6] J.E. Marsden, T.J.R. Hughes, *Mathematical Foundations of Elasticity*, Dover Publications, New York, 1994.
- [7] C. Truesdell, W. Noll, S.S. Antman, *The Non-Linear Field Theories of Mechanics*, Springer-Verlag, Berlin, 1965.
- [8] J.N. Reddy, *An Introduction to Continuum Mechanics*, Cambridge University Press, Cambridge, 2013.
- [9] L.E. Malvern, *Introduction to the Mechanics of a Continuous Medium*, Prentice-Hall, Englewood Cliffs, NJ, 1969.
- [10] P. Chadwick, *Continuum Mechanics: Concise Theory and Problems*, Dover Publications, New York, 1999.
- [11] C.F. Van Loan, G.H. Golub, *Matrix Computations*, The John Hopkins University Press, Baltimore, MD, 1996.

- [12] G. Strange, Introduction to Linear Algebra, Wellesley Cambridge Press, Wellesley, MA, 2003.
- [13] Matlab, <http://www.mathworks.com/>.
- [14] Mathematica, <http://www.wolfram.com/>.
- [15] Python, <http://www.python.org/>.
- [16] J.A. Fay, Introduction to Fluid Mechanics, The MIT Press, Cambridge, MA, 1994.
- [17] J.H. Williams, Fundamentals of Applied Dynamics, Wiley, New York, 1995.
- [18] J.H. Lienhard V, J.H. Lienhard IV, A Heat Transfer Textbook, Plagiston Press, Cambridge, MA, 2003.
- [19] K. Denbigh, The Principles of Chemical Equilibrium, third ed., Cambridge University Press, Cambridge, 1971.
- [20] F. Reif, Fundamentals of Statistical and Thermal Physics, McGraw-Hill, New York, 1965.
- [21] M.J. Moran, H.N. Shapiro, Fundamentals of Engineering Thermodynamics, Wiley, New York, 2003.

# 5 Elasticity/Hyperelasticity

---

## Chapter Outline

5.1	Introduction	210
5.2	Linear Elasticity	211
5.2.1	Isotropic Elasticity	211
5.2.2	Anisotropic Elasticity	215
	Orthotropic Elasticity	216
5.2.3	Transversely Isotropic Elasticity	217
5.3	Isotropic Hyperelasticity	218
5.3.1	Continuum Mechanics Foundations	219
5.3.2	Similarity Between Uniaxial Compression and Biaxial Tension	225
5.3.3	Similarity Between Pure Shear and Planar Tension	226
5.3.4	Dependence of Stored Energy on $I_1$ and $I_2$	229
5.3.5	Freely Jointed Chain Model	232
5.3.6	Neo-Hookean Model	236
5.3.7	Mooney-Rivlin Model	243
5.3.8	Yeoh Model	245
5.3.9	Polynomial in $I_1$ and $I_2$ Model	250
5.3.10	Eight-Chain Model	250
5.3.11	Ogden Model	259
5.3.12	Gent Model	263
5.3.13	Horgan and Saccomandi Model	265
5.3.14	Knowles Model	268
5.3.15	Response Function Hyperelasticity	270
5.3.16	Extended Tube Model	273
5.3.17	BAM Model	275
5.4	Summary of Predictive Capabilities of Isotropic Hyperelastic Models	277
5.5	Anisotropic Hyperelasticity	281
5.5.1	Generalized Fung Model	282
5.5.2	Invariant Based Anisotropy	282

5.5.3	Bischoff Anisotropic Eight-Chain Model	283
5.5.4	Bergstrom Anisotropic Eight-Chain Model	285
5.5.5	Holzappel-Gasser-Ogden Model	285
5.6	Hyperelastic Foam Models	287
5.6.1	Blatz-Ko Foam Model	289
5.6.2	Hyperfoam Model	290
5.7	Mullins Effect Models	292
5.7.1	Ogden-Roxburgh Mullins Effect Model	293
5.7.2	Qi-Boyce Mullins Effect Model	295
5.8	Use of Hyperelasticity in Polymer Modeling	295
5.8.1	Experimental Testing	296
5.8.2	Drucker Stability	297
5.8.3	Determination of Material Parameters	298
5.8.4	Limitations of Hyperelasticity	298
5.9	Hyperelastic Code Examples	299
5.10	Exercises	303
	References	304

## 5.1 Introduction

Linear elasticity and hyperelasticity are two classes of constitutive models that are both easy to use and computationally efficient. Here, a constitutive model is considered to be easy to use if it requires little effort to calibrate and if it provides robust predictions for other loading conditions than what was used for the calibration. Both of these classes of models can provide useful predictions of the mechanical behavior of different polymers. The following sections discuss the theory behind these two classes of models, and how and when they can be applied to solve polymer mechanics problems.

Note that although the hyperelastic models that are discussed in this chapter are most useful for rubber-like material, they are also a critical component/building block for both linear and non-linear viscoelastic and viscoplastic models that will be discussed in detail in the following chapters. Hence, the theory of hyperelasticity is of significant importance and will be discussed in detail in this chapter.

The last few sections of this chapter introduce extensions of traditional hyperelastic models that allow for predictions of anisotropic non-linear hyperelastic behaviors, and predictions of the Mullins [1–3] effect that is commonly observed for elastomers, thermo-plastic elastomers (TPE), and other elastomer-like materials.

## 5.2 Linear Elasticity

Linear elasticity is the most direct way to represent the small strain mechanical behavior of solid polymers. The theory of linear elasticity is covered in numerous introductory textbooks [4–6] and in advanced textbooks [7–12]. The advanced theory of elasticity is very elegant but requires more sophisticated mathematics than what is covered in this introductory text. For our purposes it is sufficient to consider simple forms of isotropic and anisotropic elasticity, which are discussed in the following two sections.

### 5.2.1 Isotropic Elasticity

The most elementary form of the theory of elasticity is *isotropic* elasticity. In this form of the theory the stress is proportional to the applied strain and independent of the orientation of the material body. The constitutive equation for an elastic material is often called *Hooke's law* and can be written in different equivalent but alternative forms. One common representation is given by the following set of equations that determine the strain for a given stress state:

$$\varepsilon_{11} = \frac{1}{E} [\sigma_{11} - \nu (\sigma_{22} + \sigma_{33})], \quad (5.1)$$

$$\varepsilon_{22} = \frac{1}{E} [\sigma_{22} - \nu (\sigma_{33} + \sigma_{11})], \quad (5.2)$$

$$\varepsilon_{33} = \frac{1}{E} [\sigma_{33} - \nu (\sigma_{11} + \sigma_{22})], \quad (5.3)$$

$$\varepsilon_{12} = \frac{\sigma_{12}}{2\mu}, \quad (5.4)$$

$$\varepsilon_{23} = \frac{\sigma_{23}}{2\mu}, \quad (5.5)$$

$$\varepsilon_{31} = \frac{\sigma_{31}}{2\mu}. \quad (5.6)$$

In these equations,  $E$  is the Young's modulus (also called the modulus of elasticity),  $\mu$  is the shear modulus, and  $\nu$  is the Poisson's ratio. This set of equations can also be written in a more condensed form as follows:

$$\varepsilon_{ij} = \frac{1 + \nu}{E} \sigma_{ij} - \frac{\nu}{E} \sigma_{kk} \delta_{ij}, \quad (5.7)$$

where the indices  $i$  and  $j$  take the values 1, 2, and 3, and  $\delta_{ij}$  is the Kronecker delta function defined by:

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \quad (5.8)$$

Another common and useful way to write Hooke's law is to instead solve for the stresses in terms of the applied strains:

$$\sigma_{ij} = 2\mu\varepsilon_{ij} + \lambda\varepsilon_{kk}\delta_{ij}, \quad (5.9)$$

where  $\mu$  is the shear modulus and  $\lambda$  is Lamé's constant.

The constitutive theory for linear elasticity requires two material parameters that need to be specified from experimental data. There are different, equally valid, pairs of material parameters that can be chosen. The most common material parameters and conversion formulas between the different parameters are summarized in [Table 5.1](#). This table shows that if we know two of the elastic constants then we can calculate any of the other constants, e.g. if we know the shear modulus ( $\mu$ ) and the bulk modulus ( $\kappa$ ), then the Poisson's ratio can be calculated from:

$$\nu = \frac{3\kappa - 2\mu}{6\kappa + 2\mu}. \quad (5.10)$$

The typical approach to calibrate a linear elasticity model involves a uniaxial tension experiment where the stress-strain response determines the Young's modulus, and the Poisson's ratio is obtained from the transverse contraction in the tension experiment. After the two material parameters have been determined,



**Table 5.1 Conversion Equations Between Different Common Linear Elastic Material Parameters**

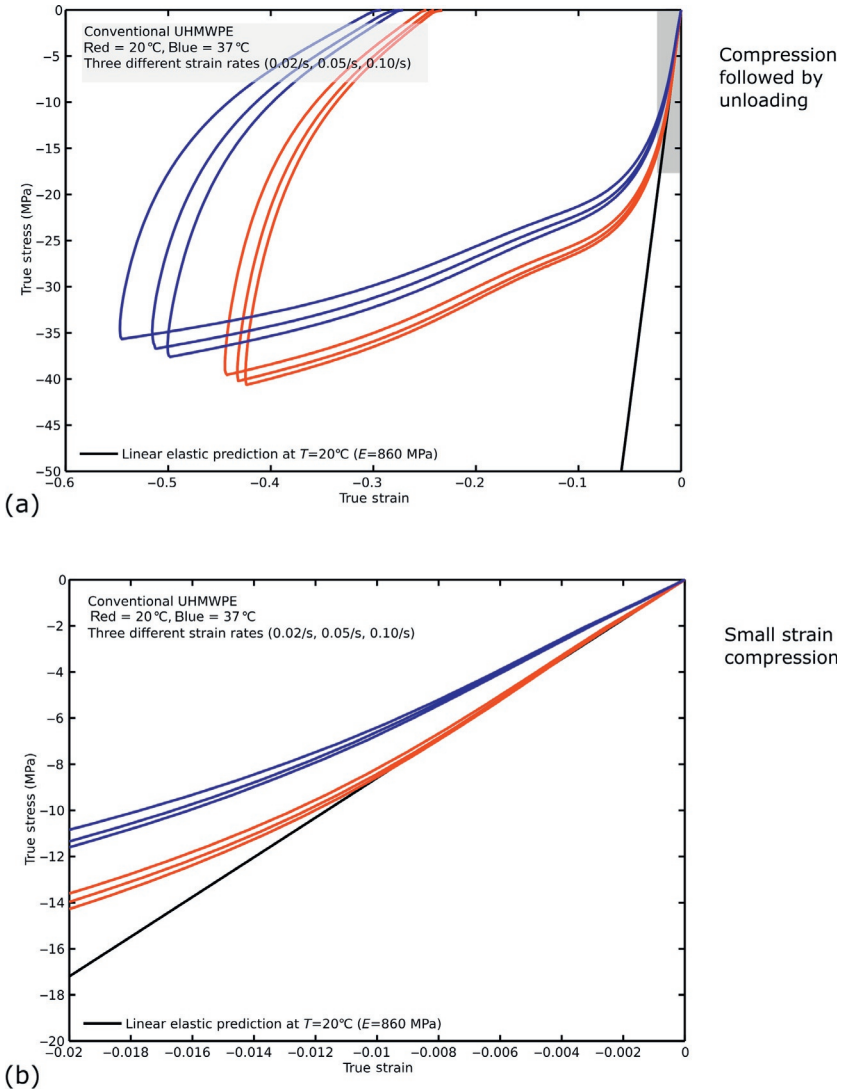
Known Elastic Constants	$E$	$\nu$	$\mu$	$\kappa$	$\lambda$
Shear modulus $\mu$ , Bulk modulus $\kappa$	$\frac{9\kappa\mu}{3\kappa+\mu}$	$\frac{3\kappa-2\mu}{6\kappa+2\mu}$	$\mu$	$\kappa$	$\frac{3\kappa-2\mu}{3}$
Young's modulus $E$ , Poisson's ratio $\nu$	$E$	$\nu$	$\frac{E}{2(1+\nu)}$	$\frac{E}{3(1-2\nu)}$	$\frac{E\nu}{(1+\nu)(1-2\nu)}$
Young's modulus $E$ , Shear modulus $\mu$	$E$	$\frac{E-2\mu}{2\mu}$	$\mu$	$\frac{E\mu}{3(3\mu-E)}$	$\frac{\mu(E-2\mu)}{3\mu-E}$
Young's modulus $E$ , Bulk modulus $\kappa$	$E$	$\frac{3\kappa-E}{6\kappa}$	$\frac{3\kappa E}{9\kappa-E}$	$\kappa$	$\frac{3\kappa(3\kappa-E)}{9\kappa-E}$
Shear modulus $\mu$ , Lame's constant $\lambda$	$\frac{\mu(3\lambda+2\mu)}{\lambda+\mu}$	$\frac{\lambda}{2(\lambda+\mu)}$	$\mu$	$\frac{3\lambda+2\mu}{3}$	$\lambda$

the calibrated linear elasticity model can be used to predict the behavior of the material in any deformation state in a finite element (FE) analysis (as long as the material behavior is linear elastic).

The main limitation of using a linear elastic model for predicting the mechanical behavior of polymer materials is that polymers behave linearly only for very small strains<sup>1</sup> and a restricted range of strain-rates and temperatures. When performing FE simulations it is sometimes possible to capture the response at different temperatures by specifying the temperature dependence of the elastic constants.

One example illustrating the use and limitations of linear elasticity is shown in [Figure 5.1](#). This figure compares experimental data for ultra high molecular weight polyethylene (UHMWPE) at three different strain rates and two temperatures with predictions from linear elasticity theory. It is clear that the linear elasticity model predictions are only accurate at one temperature and for

<sup>1</sup>Note that some polymers, for example low density polyethylene, often do not exhibit a linear elastic response for any strains at room temperature.



**Figure 5.1** Comparison between experimental data for UHMWPE and predictions from linear elasticity. (a) Stress-strain response from a compression followed by an unloading cycle. (b) Small strain response.

strains smaller than about 1%. The figure also shows that the strain range within the response is linear elastic decreases with increasing temperature.

Linear elasticity is mostly useful for predicting the small-strain behavior of thermoplastics below the glass transition

temperature, and for thermosets. For elastomer-like materials it is recommended to use a hyperelastic model or one of the more advanced models discussed in later chapters.

### 5.2.2 Anisotropic Elasticity

The theory for linear elasticity can be extended to include anisotropic behavior. Many polymers, e.g. fiber reinforced composites, drawn polymer films, or other polymers with a preferred molecular orientation, including many biopolymers, are anisotropic to various degrees. For these materials it is sometimes useful to represent their behavior using anisotropic elasticity.

For an anisotropic elastic material the Hooke's law can be written:

$$\varepsilon_{ij} = S_{ijkl}\sigma_{kl}, \quad (5.11)$$

or when expressed as a function of the strain state

$$\sigma_{ij} = C_{ijkl}\varepsilon_{kl}. \quad (5.12)$$

In these equations  $S_{ijkl}$  is the compliance tensor, and  $C_{ijkl}$  is the stiffness tensor. These equations show that the stress and strain tensors are linearly dependent on each other by a linear stiffness or compliance tensor.<sup>2</sup>

The theory for anisotropic elasticity is covered in detail in numerous textbooks [4–6], here we will simply summarize some of the more important and useful results for engineering analysis. As is shown in Equations (5.11) and (5.12), the stiffness and compliance tensors have  $3 \times 3 \times 3 \times 3 = 81$  components. Fortunately, since both the strain and stress tensors are symmetric, the number of independent components of  $S$  and  $C$  can be reduced to 36 by using symmetry arguments. Hence, it is common to arrange the stiffness and compliance tensors into  $6 \times 6$  tensors as shown in the following equations:

---

<sup>2</sup>The stiffness and compliance tensors are fourth-order tensors that are discussed in more detail in various textbooks [4–6, 12].

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{23} \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} & S_{15} & S_{16} \\ S_{21} & S_{22} & S_{23} & S_{24} & S_{25} & S_{26} \\ S_{31} & S_{32} & S_{33} & S_{34} & S_{35} & S_{36} \\ S_{41} & S_{42} & S_{43} & S_{44} & S_{45} & S_{46} \\ S_{51} & S_{52} & S_{53} & S_{54} & S_{55} & S_{56} \\ S_{61} & S_{62} & S_{63} & S_{64} & S_{65} & S_{66} \end{bmatrix} \times \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix}, \quad (5.13)$$

and

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{bmatrix} \times \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{23} \end{bmatrix}. \quad (5.14)$$

Depending on the degree of anisotropy these  $6 \times 6$  matrices can often be simplified further. The following subsections present common special cases.

### **Orthotropic Elasticity**

If the mechanical response of a material contain three orthogonal symmetry planes, as illustrated in the example in [Figure 5.2](#), then the material is said to be orthotropic.

One common case of an orthotropic material is a material which contains fibers in three orthogonal directions. In this case the material contains three-fold symmetry and the constitutive equations are given by:

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{23} \end{bmatrix} = \begin{bmatrix} 1/E_1 & -\nu_{21}/E_2 - \nu_{31}/E_3 & 0 & 0 & 0 \\ -\nu_{12}/E_1 & 1/E_2 & -\nu_{32}/E_3 & 0 & 0 & 0 \\ -\nu_{13}/E_1 & -\nu_{23}/E_2 & 1/E_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/(2G_{12}) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/(2G_{13}) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/(2G_{23}) \end{bmatrix} \times \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix}, \quad (5.15)$$

where  $E_1, E_2, E_3, G_{12}, G_{13}, G_{23}, \nu_{12}, \nu_{13}, \nu_{21}, \nu_{23}, \nu_{31}$ , and  $\nu_{32}$  are material parameters. Here, the Poisson's ratios  $\nu_{ij}$  is defined



**Figure 5.2** The microstructure of wood is typically orthotropic.

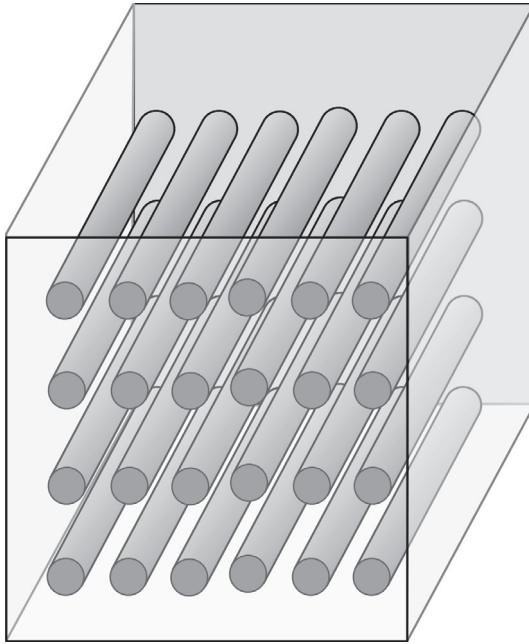
by the transverse strain in the  $j$ -direction when the material is stressed in the  $i$ -direction. In general, the Poisson's ratio terms are not symmetric, i.e.  $\nu_{ij} \neq \nu_{ji}$ .

### 5.2.3 Transversely Isotropic Elasticity

A special case of orthotropic materials are those that have isotropic properties in one plane (here taken to be the 1-2 plane), and different properties in the direction normal to this plane. This type of material symmetry is called transverse isotropic and is described by 5 material parameters ( $E_p$ ,  $E_t$ ,  $G_t$ ,  $\nu_p$ ,  $\nu_{tp}$ , and  $\nu_{pt} = \nu_{tp}E_p/E_t$ ) as shown in the following equations

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{23} \end{bmatrix} = \begin{bmatrix} 1/E_p & -\nu_p/E_p & -\nu_{tp}/E_t & 0 & 0 & 0 \\ -\nu_p/E_p & 1/E_p & -\nu_{tp}/E_t & 0 & 0 & 0 \\ -\nu_{pt}/E_p & -\nu_{pt}/E_p & 1/E_t & 0 & 0 & 0 \\ 0 & 0 & 0 & (1 + \nu_p)/E_p & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/G_t & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/G_t \end{bmatrix} \times \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix}. \quad (5.16)$$

An example of material that is transversely isotropic is a fiber-reinforced composite with all fibers oriented in one direction, see [Figure 5.3](#).



**Figure 5.3** Fiber-reinforced composite having a transverse isotropic behavior.

**Note.** *The number of experimental tests required to fully determine the material parameters for these anisotropic elastic material models can be large. The exact number and type of required tests depends on the type of anisotropic response, typically a combination of tension (or compression) tests and shear tests are required.*

### 5.3 Isotropic Hyperelasticity

Hyperelasticity is a generalization of linear elasticity that is non-linear and suitable for large strain predictions. Throughout the years there have been an extensive amount of work done developing different hyperelasticity theories, and many of these models are available in commercial FE software. One of the most important and interesting aspects of hyperelasticity is that it is the simplest model representation that in some cases can be connected to the micromechanisms that drive the deformation

behavior of polymers. In the following sections we will present some of the most common and useful theories of hyperelasticity.

The continuum mechanics foundation of the hyperelastic theories was derived in Section 4.11.1. In the next section we will extend the general expressions for the stress in terms of the Helmholtz free energy into different and easily applied representations. This section focuses on the different theories, and the strengths and limitations of the different theories to predict the behavior of different polymeric materials.

### 5.3.1 Continuum Mechanics Foundations

As was shown in Section 4.11.1, the Helmholtz free energy per unit reference volume<sup>3</sup>  $\Psi(\cdot)$ , the first Piola-Kirchhoff stress  $\mathbf{P}(\cdot)$ , the entropy per unit reference volume  $\eta_0(\cdot)$ , and the heat flux per unit reference surface area  $\mathbf{Q}(\cdot)$  of a *thermoelastic material* only depend on the deformation gradient  $\mathbf{F}$ , the temperature  $\theta_0$ , and the gradient of temperature  $\text{Grad } \theta_0$ .

To satisfy the Clausius-Duhem inequality Cauchy stress for a thermoelastic material has to be given by:

$$\boldsymbol{\sigma}(\mathbf{F}, \theta_0) = \frac{1}{J} \frac{\partial \Psi(\mathbf{F}, \theta_0)}{\partial \mathbf{F}} \mathbf{F}^\top. \quad (4.235\text{-repeat})$$

To satisfy objectivity (see Section 4.12), a change of observer implies that the stress and the Helmholtz free energy have to have the following functional forms<sup>4</sup>:

$$\boldsymbol{\sigma}(\mathbf{F}) = \mathbf{R}\boldsymbol{\sigma}(\mathbf{U})\mathbf{R}^\top = \mathbf{R}\boldsymbol{\sigma}(\mathbf{C})\mathbf{R}^\top, \quad (5.17)$$

$$\Psi(\mathbf{F}, \theta_0) = \Psi(\mathbf{U}, \theta_0) = \Psi(\mathbf{C}, \theta_0), \quad (5.18)$$

which when inserted into (4.235) gives

$$\boldsymbol{\sigma}(\mathbf{U}, \theta_0) = \mathbf{R} \left[ \frac{1}{J} \frac{\partial \Psi(\mathbf{U}, \theta_0)}{\partial \mathbf{U}} \mathbf{U}^\top \right] \mathbf{R}^\top. \quad (5.19)$$

<sup>3</sup>The notation  $\Psi(\cdot)$ ,  $\mathbf{P}(\cdot)$ , etc. indicates that these quantities are functions of arguments that are not listed for notational simplicity.

<sup>4</sup>Here, and in the following, the tensorial functionals are not explicitly showing the inherent arguments. That is, instead of writing  $\hat{\boldsymbol{\sigma}}(\mathbf{F}) = \mathbf{R}\hat{\boldsymbol{\sigma}}(\mathbf{U})\mathbf{R}^\top = \mathbf{R}\hat{\boldsymbol{\sigma}}(\mathbf{C})\mathbf{R}^\top$ , we have for notational simplicity skipped the superscripts giving:  $\boldsymbol{\sigma}(\mathbf{F}) = \mathbf{R}\boldsymbol{\sigma}(\mathbf{U})\mathbf{R}^\top = \mathbf{R}\boldsymbol{\sigma}(\mathbf{C})\mathbf{R}^\top$ .

The partial derivative of the Helmholtz free energy with respect to the right stretch tensor ( $\mathbf{U}$ ) can be converted to a derivative with respect to the right Cauchy-Green tensor ( $\mathbf{C}$ ) by the following theorem.

**Theorem.** *The partial derivative  $\partial\Psi/\partial\mathbf{U}$  can be written as follows:*

$$\frac{\partial\Psi(\mathbf{U})}{\partial\mathbf{U}} = 2\mathbf{U}\frac{\partial\Psi(\mathbf{C})}{\partial\mathbf{C}}. \quad (5.20)$$

*Proof.* This theorem can be proven in different ways. Here, we will start by converting the left-hand-side of Equation (5.20) to index notation:

$$\left(\frac{\partial\Psi}{\partial\mathbf{U}}\right)_{ij} = \frac{\partial\Psi}{\partial U_{ij}} \quad (5.21)$$

$$= \frac{\partial\Psi}{\partial C_{kl}} \frac{\partial C_{kl}}{\partial U_{ij}} \quad (5.22)$$

$$= \frac{\partial\Psi}{\partial C_{kl}} \frac{\partial(U_{km}U_{ml})}{\partial U_{ij}}. \quad (5.23)$$

Then by expanding terms and taking the partial derivative, this expression becomes:

$$\left(\frac{\partial\Psi}{\partial\mathbf{U}}\right)_{ij} = 2U_{ik} \frac{\partial\Psi}{\partial C_{kj}} \quad (5.24)$$

$$= \left(2\mathbf{U}\frac{\partial\Psi(\mathbf{C})}{\partial\mathbf{C}}\right)_{ij}, \quad (5.25)$$

which completes the proof.  $\square$

By inserting (5.20) into (5.19), the expression for the Cauchy stress automatically satisfies objectivity if:

$$\boldsymbol{\sigma}(\mathbf{C}, \theta_0) = \frac{2}{J} \mathbf{F} \frac{\partial\Psi(\mathbf{C}, \theta_0)}{\partial\mathbf{C}} \mathbf{F}^\top. \quad (5.26)$$

When working with isotropic hyperelastic models it is often convenient to express the Helmholtz free energy in terms of invariants of the right Cauchy-Green tensor instead of the actual tensor ( $\mathbf{C}$ ). It is common to use the invariants from (4.65) to (4.67):  $I_1(\mathbf{C})$ ,  $I_2(\mathbf{C})$ , and  $I_3(\mathbf{C})$ :



$$I_1(\mathbf{C}) = \lambda_1^2 + \lambda_2^2 + \lambda_3^2, \quad (5.27)$$

$$I_2(\mathbf{C}) = \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_3^2\lambda_1^2, \quad (5.28)$$

$$I_3(\mathbf{C}) = \lambda_1^2\lambda_2^2\lambda_3^2. \quad (5.29)$$

With these invariants the Helmholtz free energy can be written  $\Psi(I_1, I_2, I_3, \theta_0)$  which when inserted in (5.26) gives:

$$\boldsymbol{\sigma}(I_1, I_2, I_3, \theta_0) = \frac{2}{J} \mathbf{F} \left[ \frac{\partial \Psi}{\partial I_1} \frac{\partial I_1}{\partial \mathbf{C}} + \frac{\partial \Psi}{\partial I_2} \frac{\partial I_2}{\partial \mathbf{C}} + \frac{\partial \Psi}{\partial I_3} \frac{\partial I_3}{\partial \mathbf{C}} \right] \mathbf{F}^\top. \quad (5.30)$$

This equation can be simplified further by the following theorem.

**Theorem.** *The partial derivative of  $I_1$ ,  $I_2$ , and  $I_3$  with respect to  $\mathbf{C}$  can be written:*

$$\frac{\partial I_1(\mathbf{C})}{\partial \mathbf{C}} = \mathbf{I}, \quad (5.31)$$

$$\frac{\partial I_2(\mathbf{C})}{\partial \mathbf{C}} = I_1 \mathbf{I} - \mathbf{C}, \quad (5.32)$$

$$\frac{\partial I_3(\mathbf{C})}{\partial \mathbf{C}} = I_3 \mathbf{C}^{-1} = I_3 \mathbf{F}^{-1} \mathbf{F}^{-\top}. \quad (5.33)$$

*Proof.* The proof of this theorem can be obtained through tensor manipulations as discussed in [13, 14]. Since the proof is somewhat lengthy it is here left as an exercise.  $\square$

Inserting the partial derivatives of  $I_1$ ,  $I_2$ , and  $I_3$  into (5.30) give the following expression for the Cauchy stress:

$$\boldsymbol{\sigma}(I_1, I_2, I_3, \theta_0) = \frac{2}{J} \left( \frac{\partial \Psi}{\partial I_1} + I_1 \frac{\partial \Psi}{\partial I_2} \right) \mathbf{b} - \frac{2}{J} \frac{\partial \Psi}{\partial I_2} \mathbf{b}^2 + \frac{2}{J} I_3 \frac{\partial \Psi}{\partial I_3} \mathbf{I}. \quad (5.34)$$

It is often more common to use the invariant  $I_3(\mathbf{F}) = J$  instead of  $I_3(\mathbf{C}) = J^2$ . By inserting this substitution into Equation (5.34) and using the chain rule we get:

$$\boldsymbol{\sigma}(I_1, I_2, J, \theta_0) = \frac{2}{J} \left( \frac{\partial \Psi}{\partial I_1} + I_1 \frac{\partial \Psi}{\partial I_2} \right) \mathbf{b} - \frac{2}{J} \frac{\partial \Psi}{\partial I_2} \mathbf{b}^2 + \frac{\partial \Psi}{\partial J} \mathbf{I}. \quad (5.35)$$

The hyperelasticity models that are available in most FE programs are based on a decomposition of the deformation gradient into dilatational (volume change) and distortional<sup>5</sup> (no volume change) components (see Equations (4.38)–(4.40)):

$$\mathbf{F} = J^{1/3} \mathbf{F}^*, \quad (5.36)$$

$$\mathbf{C} = \mathbf{F}^\top \mathbf{F} = J^{2/3} \mathbf{C}^*. \quad (5.37)$$

Similarly, the invariants can also be divided into dilatational and distortional parts:

$$I_1(\mathbf{C}) = \text{tr}(\mathbf{C}) = \text{tr}(J^{2/3} \mathbf{C}^*) = J^{2/3} \text{tr}(\mathbf{C}^*) = J^{2/3} I_1^*, \quad (5.38)$$

$$I_2(\mathbf{C}) = \frac{1}{2} \left[ (\text{tr} \mathbf{C})^2 - \text{tr}(\mathbf{C}^2) \right] = J^{4/3} I_2^*. \quad (5.39)$$

Using these invariants, the Helmholtz free energy can be written  $\Psi(I_1^*(I_1, J), I_2^*(I_2, J), J)$ . Inserting this functional form into Equation (5.35) and using the chain rule gives:

$$\begin{aligned} \boldsymbol{\sigma}(I_1^*, I_2^*, J, \theta_0) = & \frac{J}{2} \left( \frac{\partial \Psi}{\partial I_1^*} \frac{\partial I_1^*}{\partial I_1} + J^{2/3} I_1^* \frac{\partial \Psi}{\partial I_2^*} \frac{\partial I_2^*}{\partial I_2} \right) \mathbf{b} - \frac{2}{J} \frac{\partial \Psi}{\partial I_2^*} \frac{\partial I_2^*}{\partial I_2} \mathbf{b}^2 \\ & + \left( \frac{\partial \Psi}{\partial I_1^*} \frac{\partial I_1^*}{\partial J} + \frac{\partial \Psi}{\partial I_2^*} \frac{\partial I_2^*}{\partial J} + \frac{\partial \Psi}{\partial J} \right) \mathbf{I}. \end{aligned} \quad (5.40)$$

This equation can be simplified by inserting the invariants from Equations (5.38) and (5.39) giving the Cauchy stress as:

$$\boldsymbol{\sigma} = \frac{2}{J} \left[ \frac{\partial \Psi}{\partial I_1^*} + \frac{\partial \Psi}{\partial I_2^*} I_1^* \right] \mathbf{b}^* - \frac{2}{J} \frac{\partial \Psi}{\partial I_2^*} (\mathbf{b}^*)^2 + \left[ \frac{\partial \Psi}{\partial J} - \frac{2I_1^*}{3J} \frac{\partial \Psi}{\partial I_1^*} - \frac{4I_2^*}{3J} \frac{\partial \Psi}{\partial I_2^*} \right] \mathbf{I}. \quad (5.41)$$

If there is no dependence on  $I_2^*$ , then the Cauchy stress can be written

$$\boldsymbol{\sigma} = \frac{2}{J} \frac{\partial \Psi}{\partial I_1^*} \text{dev}[\mathbf{b}^*] + \frac{\partial \Psi}{\partial J} \mathbf{I}. \quad (5.42)$$

*Equations (5.41) and (5.42) are very useful and will be used extensively in the remainder of this chapter.*

<sup>5</sup>Here, and in the following, distortional scalars and tensors are denoted by a superscript \*.

**Example**

For an incompressible material, Equation (5.41) be simplified to:

$$\boldsymbol{\sigma} = 2 \left[ \frac{\partial \Psi}{\partial I_1} + \frac{\partial \Psi}{\partial I_2} I_1 \right] \mathbf{b} - 2 \frac{\partial \Psi}{\partial I_2} (\mathbf{b})^2 + p \mathbf{I}, \quad (5.43)$$

where  $p$  is a pressure term that is determined from the boundary conditions. Hence, if an incompressible hyperelastic material is exposed to uniaxial loading, then the Cauchy stresses in the loading and transverse directions are given by:

$$\sigma_{\text{uniax}} = 2 \left[ \frac{\partial \Psi}{\partial I_1} + \frac{\partial \Psi}{\partial I_2} I_1 \right] \lambda^2 - 2 \frac{\partial \Psi}{\partial I_2} \lambda^2 + p, \quad (5.44)$$

$$\sigma_{\text{transverse}} = 2 \left[ \frac{\partial \Psi}{\partial I_1} + \frac{\partial \Psi}{\partial I_2} I_1 \right] \frac{1}{\lambda} - 2 \frac{\partial \Psi}{\partial I_2} \frac{1}{\lambda^2} + p \equiv 0 \quad (5.45)$$

giving

$$\sigma_{\text{uniax}} = 2 \left[ \frac{\partial \Psi}{\partial I_1} + \frac{\partial \Psi}{\partial I_2} I_1 \right] \left( \lambda^2 - \frac{1}{\lambda} \right) - 2 \frac{\partial \Psi}{\partial I_2} \left( \lambda^4 - \frac{1}{\lambda^2} \right). \quad (5.46)$$

Furthermore, if the material has no dependence on  $I_2$ , and is exposed to incompressible uniaxial loading, then the Cauchy stress is given by

$$\sigma_{\text{uniax}} = 2 \frac{\partial \Psi}{\partial I_1} \left( \lambda^2 - \frac{1}{\lambda} \right). \quad (5.47)$$

If the hyperelastic material has no dependence on  $I_2$ , the Cauchy stress in incompressible planar loading is given by

$$\sigma_{\text{planar}} = 2 \frac{\partial \Psi}{\partial I_1} \left( \lambda^2 - \frac{1}{\lambda^2} \right), \quad (5.48)$$

and if exposed to incompressible biaxial loading, the Cauchy stress is given by

$$\sigma_{\text{biaxial}} = 2 \frac{\partial \Psi}{\partial I_1} \left( \lambda^2 - \frac{1}{\lambda^4} \right). \quad (5.49)$$

Instead of using the invariants  $I_1$ ,  $I_2$ , and  $I_3$ , it is sometimes useful to express the Helmholtz free energy in terms of the

principal stretches  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$ :  $\Psi(\lambda_1, \lambda_2, \lambda_3, \theta_0)$ . In this case the principal Cauchy stresses can be obtained from Equation (5.26) as follows:

$$\sigma_i(\lambda_1, \lambda_2, \lambda_3, \theta_0) = \sum_{j=1}^3 \frac{2}{J} \lambda_i^2 \frac{\partial \Psi}{\partial \lambda_j} \frac{\partial \lambda_j}{\partial C_{ii}}. \quad (5.50)$$

From the definition of  $\mathbf{C}$  we get that  $\partial \lambda_i / \partial C_{ii} = 1/(2\lambda_i)$ , which when inserted in Equation (5.50) gives:

$$\sigma_i(\lambda_1, \lambda_2, \lambda_3, \theta_0) = \frac{\lambda_i}{J} \frac{\partial \Psi}{\partial \lambda_i} \quad (\text{no summation}). \quad (5.51)$$

The complete Cauchy stress tensor can then be obtained from:

$$\boldsymbol{\sigma} = \sum_{i=1}^3 \frac{\lambda_i}{J} \frac{\partial \Psi(\lambda_1, \lambda_2, \lambda_3, \theta)}{\partial \lambda_i} \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i, \quad (5.52)$$

where  $\hat{\mathbf{n}}_i$  are the principal directions of the left Cauchy-Green tensor  $\mathbf{b} = \mathbf{F}\mathbf{F}^T$ , see (4.106). Using the same approach we can similarly show that the principal values of the first Piola-Kirchhoff stress can be obtained from:

$$P_i(\lambda_1, \lambda_2, \lambda_3, \theta_0) = \frac{\partial \Psi}{\partial \lambda_i}, \quad (5.53)$$

giving the Piola-Kirchhoff stress tensor

$$\mathbf{P} = \sum_{i=1}^3 \frac{\partial \Psi(\lambda_1, \lambda_2, \lambda_3, \theta_0)}{\partial \lambda_i} \hat{\mathbf{n}}_i \otimes \hat{\mathbf{N}}_i. \quad (5.54)$$

Similarly, if the Helmholtz free energy is expressed in terms of the distortional stretches:  $\lambda_i^* = J^{-1/3} \lambda_i$ , then the Cauchy stress is given by:

$$\boldsymbol{\sigma} = \sum_{i=1}^3 \frac{\lambda_i^*}{J} \frac{\partial \Psi}{\partial \lambda_i^*} \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i + \frac{\partial \Psi}{\partial J} \mathbf{I}. \quad (5.55)$$

### 5.3.2 Similarity Between Uniaxial Compression and Biaxial Tension

The deformation gradient in incompressible uniaxial compression is

$$\mathbf{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & 1/\sqrt{\lambda} & 0 \\ 0 & 0 & 1/\sqrt{\lambda} \end{bmatrix}, \quad (5.56)$$

and the deformation gradient in incompressible biaxial tension is given by

$$\mathbf{F} = \begin{bmatrix} 1/\lambda^2 & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}. \quad (5.57)$$

At first sight these two deformation gradients look quite different but if we select  $\lambda = 0.5$  in uniaxial compression, and  $\lambda = 2$  in biaxial tension, then the deformation gradient in both cases becomes

$$\mathbf{F} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}. \quad (5.58)$$

This means that for an incompressible material uniaxial compression and biaxial tension impose the same deformation state on the material, and hence are equivalent. The two loading modes, however, give different stresses due to the difference in required pressure to satisfy the stress boundary conditions.

Since no real material is fully incompressible, this similarity between uniaxial compression and biaxial tension is not strictly valid. To explore the difference in the predicted stresses due to the material compressibility consider a compressible Neo-Hookean (NH) material model with a shear modulus of  $\mu = 1$  MPa, and different values for the bulk modulus  $\kappa$ . To get concrete stress values consider a compressible stretch state of  $\lambda = 0.5$ , and a biaxial tensile stretch of  $1/\sqrt{0.5}$ . Under these conditions the calculated true stress values are listed in [Table 5.2](#).

**Table 5.2 Summary of Calculated Stress Values in Uniaxial Compression and the Corresponding Biaxial Tension Deformation State**

$\kappa$	$\nu$	Stress Uniaxial	Stress Biaxial	Error
500	0.499	-1.7511 MPa	1.7421 MPa	0.5%
100	0.495	-1.7554 MPa	1.7116 MPa	2.6%
50	0.49	-1.7610 MPa	1.6761 MPa	5.1%
10	0.45	-1.8097 MPa	1.4570 MPa	24.2%

The shear modulus was  $\mu = 1$  MPa. The applied uniaxial stretch was  $\lambda = 0.5$ , and the applied biaxial stretch was  $\lambda = 1/\sqrt{0.5}$ .

The table shows that the bulk modulus needs to very high compared to the shear modulus in order for there to be a similarity between the predicted stress in uniaxial compression and biaxial tension.

### 5.3.3 Similarity Between Pure Shear and Planar Tension

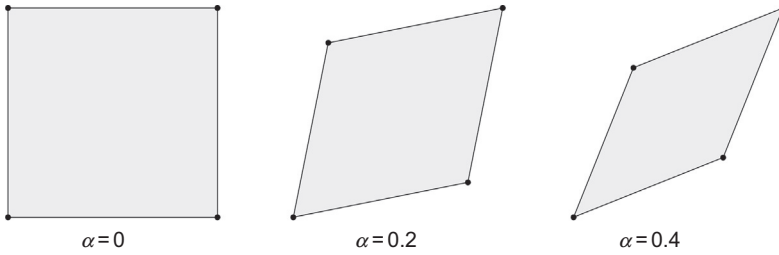
The results from a planar tension test can under certain conditions be converted to equivalent pure shear data.<sup>6</sup> On occasion, this can be useful if shear data is required, but the experimental equipment for performing shear tests is not available.

To show this similarity between pure shear and planar tension, start with the deformation gradient of an incompressible material in pure shear:

$$\mathbf{F} = \begin{bmatrix} 1/(1 - \alpha^2) & 0 & 0 \\ 0 & 1 & \alpha \\ 0 & \alpha & 1 \end{bmatrix}, \quad (5.59)$$

where  $\alpha$  is the shear displacement. The term  $F_{11}$  is taken to be  $1/(1 - \alpha^2)$  in order to conserve volume, that is to get  $\det(\mathbf{F}) = 1$ . Figure 5.4 shows how a square is deformed in pure shear for different  $\alpha$  values.

<sup>6</sup>Note that pure shear does not impose the same deformation as simple shear. Both loading modes apply shear, but in different ways.



**Figure 5.4** Deformation of a square in pure shear with different  $\alpha$  values. Note that the deformation of a square becomes a rhombus (with all equal sides). This is different than simple shear of a square that becomes a parallelogram.

Now consider the deformation gradient in incompressible plane strain tension:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 + \alpha & 0 \\ 0 & 0 & 1/(1 + \alpha) \end{bmatrix}, \quad (5.60)$$

where  $\alpha$  is the same displacement value as in (5.59). With this deformation gradient the stretching is in the 2-direction, and the 1-direction has no strain.

If we rotate this deformation state by 45% around the 1-direction then the rotation matrix becomes:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 45\% & \sin 45\% \\ 0 & -\sin 45\% & \cos 45\% \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 0 & 1 & 1 \\ 0 & -1 & 1 \end{bmatrix}, \quad (5.61)$$

and the rotated deformation gradient becomes:

$$\mathbf{F} = \mathbf{Q}^\top \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 + \alpha & 0 \\ 0 & 0 & 1/(1 + \alpha) \end{bmatrix} \cdot \mathbf{Q}. \quad (5.62)$$

For small applied strains this equation can be simplified to:

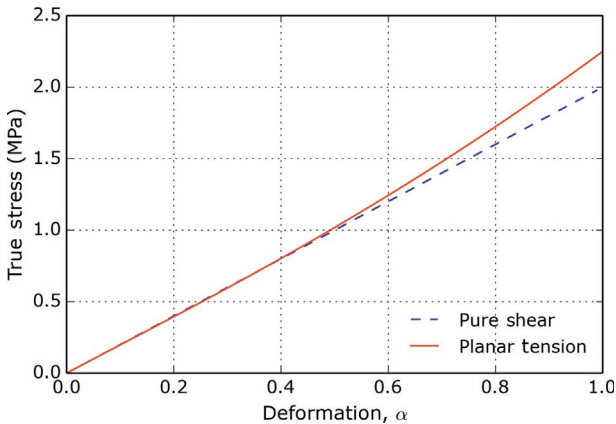
$$\mathbf{F} = \begin{bmatrix} 1 + \Delta & 0 & 0 \\ 0 & (1 + \alpha^2/2 + \mathcal{O}(\alpha^3)) & (\alpha - \alpha^2/2 + \mathcal{O}(\alpha^3)) \\ 0 & (\alpha - \alpha^2/2 + \mathcal{O}(\alpha^3)) & (1 + \alpha^2/2 + \mathcal{O}(\alpha^3)) \end{bmatrix}, \quad (5.63)$$

where  $\Delta$  is a small value that is needed in order to enforce incompressibility,  $\mathcal{O}(\alpha^3)$  is big O notation indicating the term is of order  $\alpha^3$ . If  $\alpha \ll 1$  then  $\alpha^2$  and higher terms can be ignored giving

$$\mathbf{F} = \begin{bmatrix} 1 + \Delta & 0 & 0 \\ 0 & 1 & \alpha \\ 0 & \alpha & 1 \end{bmatrix}, \quad (5.64)$$

which is the same deformation gradient as in pure shear. This shows that for incompressible materials, at small strains the pure shear and planar tension behaviors are the same.

The difference between pure shear and planar tension increases with the applied deformation. Figure 5.5 shows the true stress as a function of the deformation  $\alpha$  for the idealized case of incompressible loading.<sup>7</sup> In this figure the stress is  $\sigma_{23}$  for pure shear and  $\sigma_{22}$  for planar tension. The stress was calculated using a Neo-Hookean (NH) material model with a shear modulus of 1 MPa. The figure shows that the two deformation modes start to diverge significantly for deformations  $\alpha > 0.5$ .



**Figure 5.5** Comparison between predicted stress in pure shear and the corresponding planar tension for different applied deformations.

<sup>7</sup>The difference between the two loading modes will be even larger if the material is compressible.

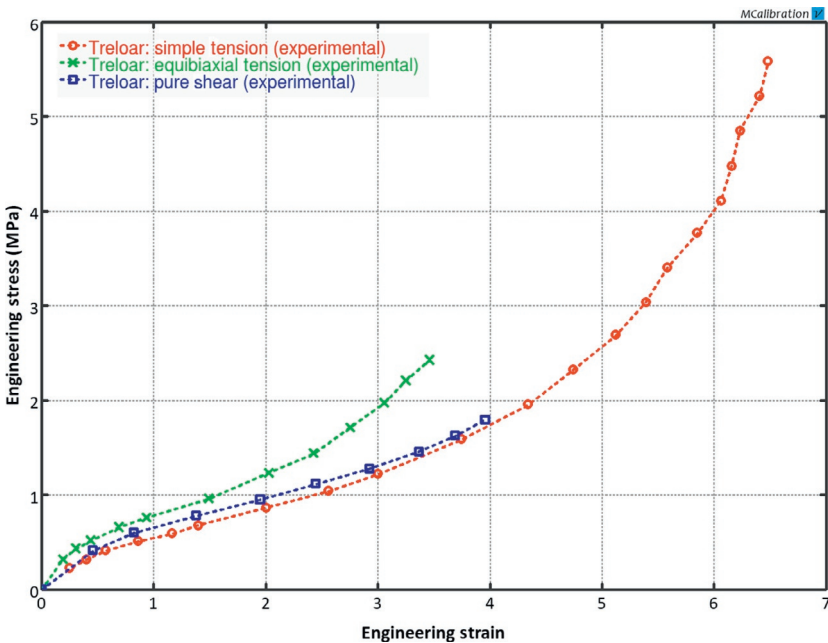


### 5.3.4 Dependence of Stored Energy on $I_1$ and $I_2$

The Helmholtz free energy that is stored in a solid polymer<sup>8</sup> as it is being deformed depends on the applied deformation invariants  $I_1$ ,  $I_2$ , and  $J$ . That is, a hyperelastic theory is uniquely defined by how the stored energy depends on these invariants.

To better illustrate these dependencies it is useful to consider real experimental data for an elastomer. Figure 5.6 shows experimental data from Treloar [15, 16] for a natural rubber tested in uniaxial tension, biaxial tension, and pure shear.

In this case the rubber material was deformed to very large engineering strains in excess of 300%, and for each applied strain value the equilibrium stress was measured after the material was allowed to relax. This figure shows, as expected, that the stress-



**Figure 5.6** Experimental data in uniaxial tension, biaxial tension, and pure shear [15, 16].

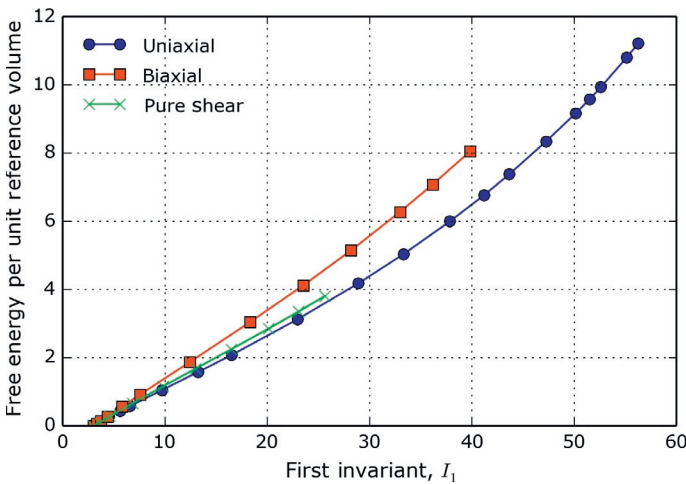
<sup>8</sup>This is only true for materials that are non-linear elastic and follow the theory of hyperelasticity. In other cases some of the applied energy is dissipated and some is stored.

strain response for all three loading modes is non-linear, and that the biaxial stress is the highest and that the uniaxial stress is the lowest.

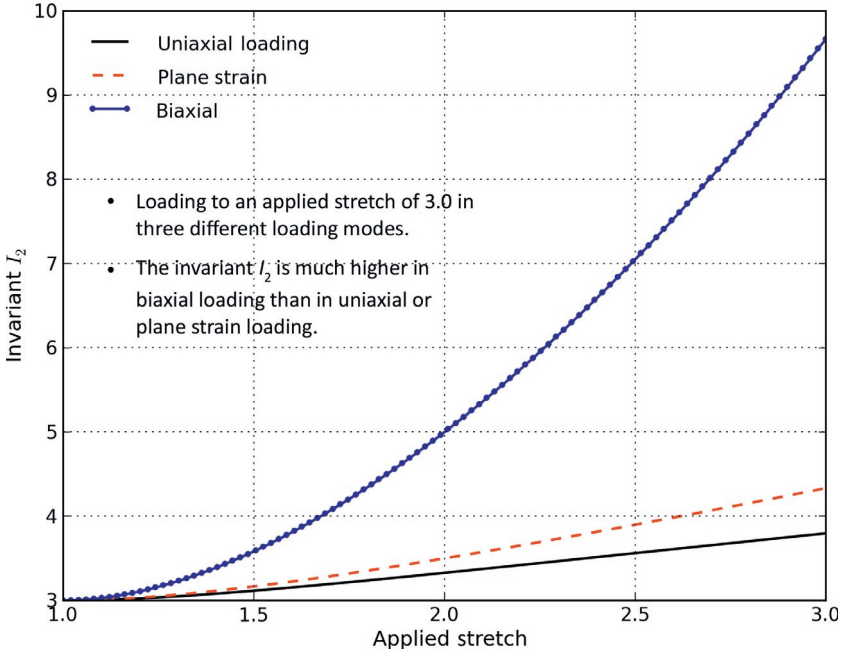
From this data it is straightforward to calculate the stored energy in the material as a function of the applied deformation in each loading mode. Figure 5.7 shows how the free energy depends on the first invariant  $I_1$ . As mentioned, in general the internal energy does not only depend on the first invariant  $I_1$  but also on  $I_2$  and  $J$ . This is clearly shown in the figure since the energy curves for the three loading modes do not perfectly overlap.

The influence of  $I_1$  and  $I_2$  on the free energy can be studied by calculating how  $I_1$  and  $I_2$  depend on the applied stretch in different loading modes. For example, directly inserting the deformation gradients from Equations (5.78)–(5.80) into the definitions for  $I_1$  and  $I_2$  (see Equations (4.65) and (4.66)) gives the results shown in Figures 5.8 and 5.9. The figures show that of the three loading modes, biaxial loading exposes the material to much higher  $I_2$  values than in uniaxial loading or plane strain loading.

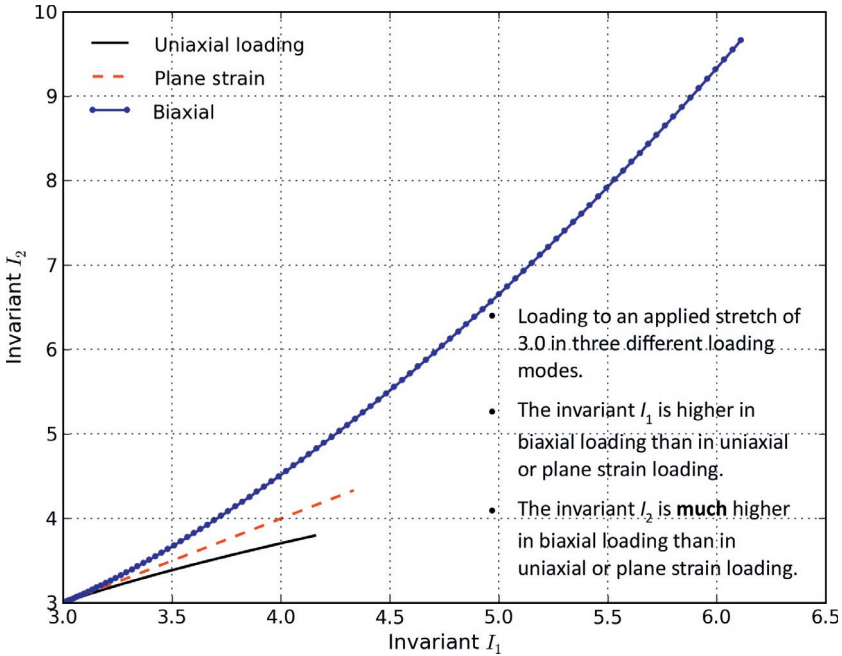
Another way to plot the data is look at the ratio  $I_2/I_1$  as a function of the applied stretch  $\lambda$ , see Figure 5.10. For any arbitrary deformation state there are three invariants, and if the material is



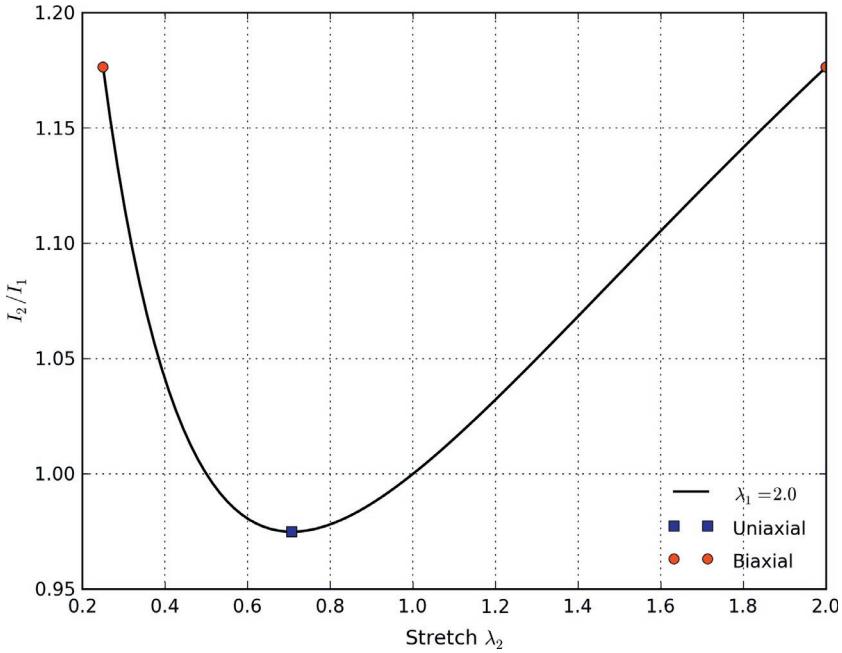
**Figure 5.7** Calculated Helmholtz free energy as a function of the first invariant  $I_1$  for data from Treloar [15, 16].



**Figure 5.8** Comparison between the  $I_2$  value in uniaxial, plane strain, and biaxial loading, as a function of the applied stretch  $\lambda_1$ .



**Figure 5.9** Comparison between the  $I_2$  value in uniaxial, plane strain, and biaxial loading, as a function of the applied  $I_1$ .



**Figure 5.10** Normalized  $I_2/I_1$  value for all loading modes for isotropic materials.

incompressible there are only two:  $I_1$  and  $I_2$ . To make the example concrete consider a deformation state where  $\lambda_1 = 2$ , and then vary  $\lambda_2$ . The  $I_1$  and  $I_2$  values can be directly calculated from the  $\lambda_1$  and  $\lambda_2$  values, and the ratio  $I_2/I_1$  established. The results from that calculation is shown in [Figure 5.10](#).

*Figure 5.10 shows that  $I_2$  is the lowest in uniaxial loading, and the highest in biaxial loading.*

These results help understand the experimental Helmholtz energy that was shown in [Figure 5.7](#). In that case the free energy in biaxial loading was higher than the energies in uniaxial loading or pure shear. One of the reasons for this is likely that the value of  $I_2$  is higher in biaxial loading, and hence plotting the energy only as a function of  $I_1$  will underestimate the free energy.

### 5.3.5 Freely Jointed Chain Model

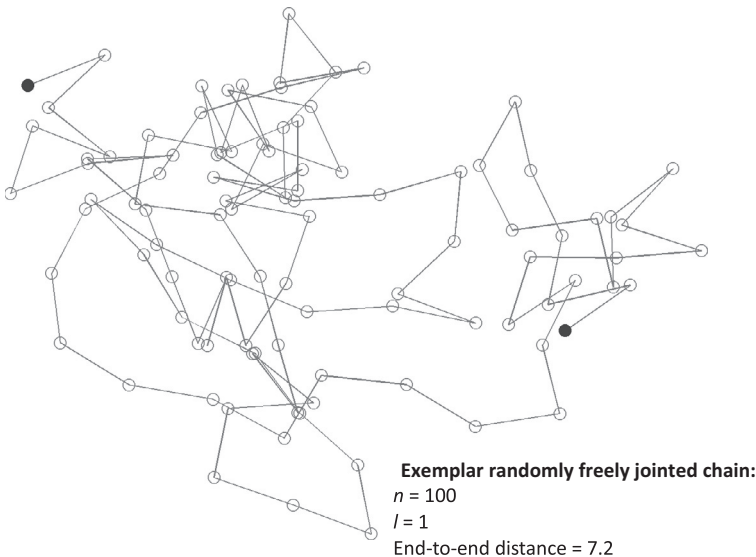
The presentation in this chapter has so far focused on deriving suitable continuum mechanics expressions for the stress response

for a given deformation history. Specifically, it has been shown that the stress can be directly calculated if the expression for the Helmholtz free energy is known. The Helmholtz free energy can either be taken from a phenomenological (often parametrized) expression for  $\Psi(\cdot)$ , or from a micromechanism inspired representation based on the deformation behavior of the microstructure.

This subsection presents a simple micromechanism inspired molecular chain model called the *Freely Jointed Chain* (FJC) model. Other micromechanical chain models, like the Kratky-Porod chain model [17] and the worm-like chain mode [18, 19] have been developed in the literature. These models are not discussed in this text.

In the FJC model the macromolecules in the polymer microstructure are represented as  $n$  “rigid links” (called Kuhn segments), each of length  $l$ , for a total chain contour length of  $L = nl$ , see Figure 5.11. This model is also called the random walk model since the molecular chains are assumed to not interact and to have a distribution that corresponds to a random walk.

The Helmholtz free energy of this molecular chain model is given by the sum of the entropies of the individual molecular



**Figure 5.11** Randomly generated freely jointed chain with  $n = 100$ , and  $l = 1$ .

chains, which can be obtained from a statistical survey of the possible molecular configurations using statistical mechanics.

As a macromolecule is stretched, the number of available configurations which can accommodate the extended chain end-to-end distance decrease, thus giving a decrease in configurational entropy,  $\Delta\eta$ . By assuming the molecules to be freely jointed with a fixed bond length, the entropy can be determined from the statistical mechanics relation [20]:

$$\eta_0 = Nk_B \ln \Omega(r) + c, \quad (5.65)$$

where  $\Omega(r)$  is the probability distribution of the end-to-end distance of the molecular chain,  $N$  is the number of chains per reference unit volume, and  $k_B = 1.38 \times 10^{-23}$  J/K is Boltzmann's constant. Flory [21] showed that the probability distribution under these conditions can be written:

$$\Omega(r) = \frac{1}{2\pi^2 r} \int_0^\infty q \sin(qr) \left[ \frac{\sin ql}{ql} \right]^n dq. \quad (5.66)$$

One approximation of the integral (5.66) that is good for  $n \gg 1$  and  $r \approx n$  is the Langevin expression attributed to Kuhn and Gr $\ddot{u}$  n [22]:

$$\Omega(r) = \frac{A'}{\beta^3} \left[ \frac{\sinh \beta}{\beta} \right]^n \exp \left[ \frac{-\beta r}{l} \right], \quad (5.67)$$

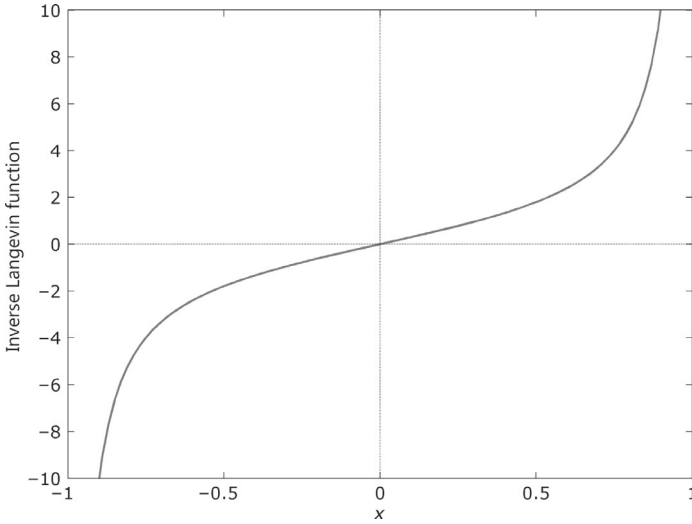
where  $\beta = \mathcal{L}^{-1}(r/(nl))$  and  $\mathcal{L}(x) = \coth(x) - 1/x$  is the Langevin function. The inverse Langevin function  $\mathcal{L}^{-1}(x)$  cannot be expressed in elementary functions but can be approximated from the following expression [23] that has a max relative error of  $6.4 \times 10^{-4}$ :

$$\mathcal{L}^{-1}(x) \approx \begin{cases} 1.31446 \tan(1.58986x) + 0.91209x, & \text{if } |x| < 0.84136, \\ 1/(\text{sign}(x) - x), & \text{if } 0.84136 \leq |x| < 1. \end{cases} \quad (5.68)$$

The inverse Langevin function is plotted in [Figure 5.12](#).

Note that in the limit as  $n \rightarrow \infty$ , Equation (5.67) becomes a Gaussian distribution:

$$\Omega(r) = \left[ \frac{3}{2\pi nl^3} \right]^{3/2} \exp \left[ \frac{-3r^2}{2nl^2} \right]. \quad (5.69)$$



**Figure 5.12** Inverse Langevin function  $\mathcal{L}(x)$ . The value of the function is only defined for  $x$ -values between  $-1$  and  $+1$ .

The change in entropy with chain length for the Langevin expression therefore can be written:

$$\frac{\partial \eta_0}{\partial r} = \frac{\partial [Nk_B \ln \Omega(r)]}{\partial r} = \frac{-Nk_B}{l} \mathcal{L}^{-1} \left( \frac{r}{nl} \right), \quad (5.70)$$

and that the corresponding expression for Gaussian chains becomes:

$$\frac{\partial \eta_0}{\partial r} = \frac{\partial [Nk_B \ln \Omega(r)]}{\partial r} = -\frac{3Nk_B r}{nl^2}, \quad (5.71)$$

where  $N$  is the number of chains per reference unit volume,  $k_B$  is Boltzmann's constant, and  $r$  is the end-to-end distance of the molecular chains.

The definition of Helmholtz free energy per unit reference volume:

$$\Psi = e_0 - \theta_0 \eta_0, \quad (4.219\text{-repeat})$$

together with the lack of storage of internal energy ( $e_0 \equiv 0$ ) since each link in the molecular chain model is rigid, allows us to directly obtain the force required to stretch a molecular chain to a given end-to-end distance:

$$f_c = \frac{\partial \Psi}{\partial r} = \frac{Nk_B\theta_0}{l} \mathcal{L}^{-1} \left( \frac{r}{nl} \right). \quad (5.72)$$

This type of micromechanics inspired model can be used to gain a better understanding the behavior of the mechanical response of polymers. This approach is also directly used in some commonly used hyperelastic material models, like the Arruda-Boyce Eight-Chain model (see [Section 5.3.10](#)).

The next few subsections discusses some of the more commonly used hyperelastic models that are available in FE programs. The purpose of this review is to illustrate some of the similarities between these models, how they work, and under what conditions they can provide accurate predictions of real polymer materials.

### 5.3.6 Neo-Hookean Model

The NH model is a simple hyperelastic model that is based on two material parameters: a shear modulus  $\mu$  and a bulk modulus  $\kappa$ . Here, as is often done, the theory for the NH model will be presented both for compressible and incompressible deformations.<sup>9</sup> The incompressible form is often easier to use in theoretical derivations, but for practical studies, including FE simulations, the compressible version is often more useful and will be emphasized in the following. Hyperelasticity models, such as the NH model, are mostly used for solid, rubber-like materials. These materials are characterized by an almost incompressible response and the actual value of the bulk modulus therefore typically has very little influence on the response of the rubber-like component. The only exception is in applications where the component is highly confined, e.g. o-rings that are compressed in a confined space (see Chapter 2, Section 2.2.9).

Like all hyperelastic material models, the NH model is specified by the expression for its Helmholtz free energy per unit reference volume. Here, the Helmholtz free energy is assumed to be independent of the temperature, and is hence identical to the

---

<sup>9</sup>The compressible version of the model can be made incompressible by studying the case when  $\kappa \rightarrow \infty$ . The incompressible version of the NH model consequently only contain one material parameter, the shear modulus  $\mu$ .



internal energy per unit reference volume  $e_0$ . For the NH model, the Helmholtz free energy per unit reference volume is given by:

$$\Psi(I_1^*, J) = \frac{\mu}{2}(I_1^* - 3) + \frac{\kappa}{2}(J - 1)^2. \quad (5.73)$$

**Note 1:** The equation for the free energy is linear in  $I_1^*$ , and can therefore not accurately capture the large-strain non-linear response of many elastomers.

**Note 2:** The equation for the free energy is not including any dependence on the second invariant  $I_2^*$ . As shown in the example below, this may cause the stress predictions to be too low in situations where the loading is mainly biaxial.

**Note 3:** The energy from volumetric deformations is quadratic in  $(J - 1)$ , giving a volumetric stress that is linear in  $(J - 1)$ , see Equation (5.74).

By using Equations (5.41), (4.66), and (4.36) it is easy to show that the compressible NH model has the following expression for the Cauchy stress for an arbitrary deformation state:

$$\boldsymbol{\sigma} = \frac{\mu}{J} \operatorname{dev} [\mathbf{b}^*] + \kappa(J - 1)\mathbf{I}, \quad (5.74)$$

where  $\mu$  is the shear modulus, and  $\kappa$  is the bulk modulus.

For the incompressible NH model, the Cauchy stress is given by Equations (5.47)–(5.49) giving the following expressions for uniaxial, planar, and biaxial deformations:

$$\sigma_{\text{uniax}} = \mu \left( \lambda^2 - \frac{1}{\lambda} \right), \quad (5.75)$$

$$\sigma_{\text{planar}} = \mu \left( \lambda^2 - \frac{1}{\lambda^2} \right), \quad (5.76)$$

$$\sigma_{\text{biaxial}} = \mu \left( \lambda^2 - \frac{1}{\lambda^4} \right). \quad (5.77)$$

Here, and in the remainder of this text, incompressible uniaxial deformation is defined by

$$\mathbf{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & 1/\sqrt{\lambda} & 0 \\ 0 & 0 & 1/\sqrt{\lambda} \end{bmatrix}, \quad (5.78)$$

incompressible planar deformation is defined by

$$\mathbf{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/\lambda \end{bmatrix}, \quad (5.79)$$

and incompressible biaxial deformation is defined by

$$\mathbf{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & 1/\lambda^2 \end{bmatrix}. \quad (5.80)$$

For incompressible uniaxial loading the NH material model can be implemented into Matlab code as follows:

Matlab Code: NH\_incompressible\_uniaxial.m

```
function [stress] = NH(time, strain, params)
%NH Neo-Hookean hyperelastic model
%Incompressible uniaxial loading
%This function is using true stress and strain
mu = params(1);
lambda = exp(strain);
stress = mu * (lambda.^2 - 1./lambda);
end
```

And as Python code as follows:

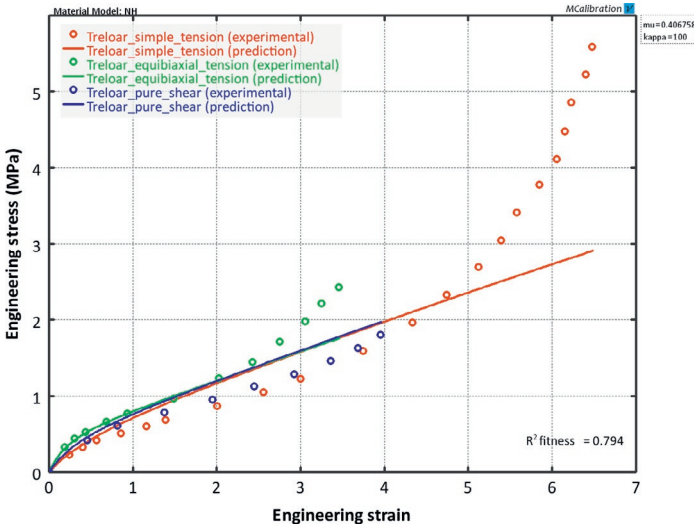
Python Code: "NH\_incompressible\_uniaxial.py"

```
from pylab import *

def NH(strain, params):
    """Neo-Hookean hyperelastic model.
    Incompressible uniaxial loading.
    This function is using true stress and strain"""
    mu = params[0]
    lam = exp(strain)
    return mu * (lam*lam - 1/lam)

strain = linspace(0, 0.8)
params = [1.0]
stress = NH(strain, params)
plot(strain, stress, 'r-')
```

The response of the NH model in this case is only controlled by the shear modulus  $\mu$ . A direct comparison between predictions



**Figure 5.13** Comparison between experimental data from Treloar [15, 16] and predictions from the Neo-Hookean material model.

from the NH model and classical experimental data for vulcanized natural rubber from Treloar [15, 16] is shown in Figure 5.13. This figure shows experimental data in simple tension, equibiaxial tension, and planar tension. As illustrated in the figure, the NH model does not capture the large strain response very well. This limitation of the model is caused by its inability to capture the limiting molecular chain stretch of the material as it becomes highly deformed.

As shown in the following code and figure, it is also easy to implement the NH model for compressible loading. This code numerically searches for the transverse strain that gives  $\sigma_{22} = 0$  for each value of the applied strain  $\varepsilon_{11}$ .

The value of the NH model lies in its simplicity—if the shear modulus is known, the response in any loading mode can be determined in a robust and computationally efficient way. The main limitation of the NH model is its limited range of conditions that give accurate predictions.

### Example: Shear Modulus in the NH Model.

In this example we will demonstrate that the material parameter  $\mu$  in Equation (5.73) is indeed the shear modulus of the material.

## Python Code: "Polymer\_Mechanics\_Chap05.py"

```

from pylab import *
import scipy.optimize

def uniaxial_stress(model, trueStrainVec, params):
    """Compressible uniaxial loading. Returns true stress."""
    stress = zeros(len(trueStrainVec))
    for i in range(len(trueStrainVec)):
        lam1 = exp(trueStrainVec[i])
        calcS22Abs = lambda x: abs(model([lam1,x,x],params)[1,1])
        # search for transverse stretch that gives S22=0
        lam2 = scipy.optimize.fmin(calcS22Abs, x0=1/sqrt(lam1),
                                  xtol=1e-9, ftol=1e-9, disp=False)
        stress[i] = model([lam1,lam2,lam2], params)[0,0]
    return stress

def NH_3D(stretch, param):
    """Neo-Hookean. 3D loading specified by stretches.
    param[0]=mu, param[1]=kappa"""
    F = array([[stretch[0],0,0], [0,stretch[1],0], [0,0,stretch[2]]])
    J = det(F)
    Fstar = J**(-1/3) * F
    bstar = dot(Fstar, Fstar.T)
    dev_bstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/J * dev_bstar + param[1]*(J-1) * eye(3)

```

## Python Code: "NH\_compressible\_uniaxial.py"

```

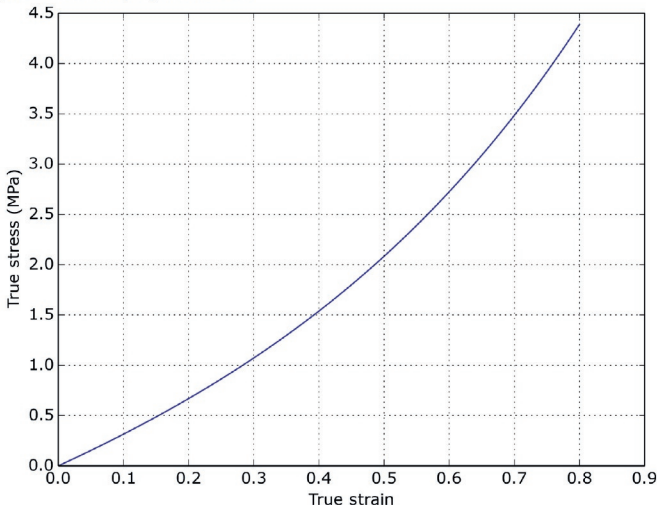
from pylab import *
from Polymer_Mechanics_Chap05 import *

trueStrain = linspace(0, 0.8, 100)
trueStress = uniaxial_stress(NH_3D, trueStrain, [1.0, 100])

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()

```

Figure Created by Python Code:



One way to show this is to consider a case of simple shear defined by:

$$\mathbf{F} = \begin{bmatrix} 1 & \gamma & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (5.81)$$

Inserting this deformation gradient into Equation (5.74) gives the shear stress  $\sigma_{12} = \mu\gamma$ , which together with Equation (5.4) demonstrates that  $\mu$  is the shear modulus.

Another way to show that  $\mu$  in Equation (5.73) is the shear modulus is to consider the derivative of  $\sigma_{\text{uniax}}$  in Equation (5.75) with respect to  $\lambda$  in the limit when  $\lambda \rightarrow 1$ :

$$\lim_{\lambda \rightarrow 1} \frac{d\sigma_{\text{uniax}}}{d\lambda} = \lim_{\varepsilon \rightarrow 0} \frac{d\sigma_{\text{uniax}}}{d\varepsilon} = 3\mu, \quad (5.82)$$

where  $\varepsilon$  is the uniaxial small strain. The derivative in Equation (5.82) is also, by definition, equal to the Young's modulus, and for an incompressible material the Young's modulus is equal to 3 times the shear modulus, hence  $\mu$  is indeed the shear modulus of the material.

### **Example: Bulk Modulus in the NH Model.**

This example will demonstrate that the material parameter  $\kappa$  in Equation (5.73) is the bulk modulus of the material. One way to show this is to consider a case of triaxial deformation:

$$\mathbf{F} = \begin{bmatrix} \varepsilon & 0 & 0 \\ 0 & \varepsilon & 0 \\ 0 & 0 & \varepsilon \end{bmatrix}. \quad (5.83)$$

Inserting this into Equation (5.74) gives  $\sigma_{11} = \sigma_{22} = \sigma_{33} = \sigma = \kappa(\lambda^3 - 1)$ . Taking the stretch derivative in the limit of small deformations gives

$$\lim_{\lambda \rightarrow 1} \frac{d\sigma}{d\lambda} = \lim_{\varepsilon \rightarrow 0} \frac{d\sigma}{d\varepsilon} = 3\kappa. \quad (5.84)$$

From Equation (5.9) and Table 5.1 we also know that  $d\sigma/d\varepsilon$  in a triaxial deformation is equal to  $3\kappa$ , hence the parameter  $\kappa$  in Equation (5.73) is the initial bulk modulus of the material.

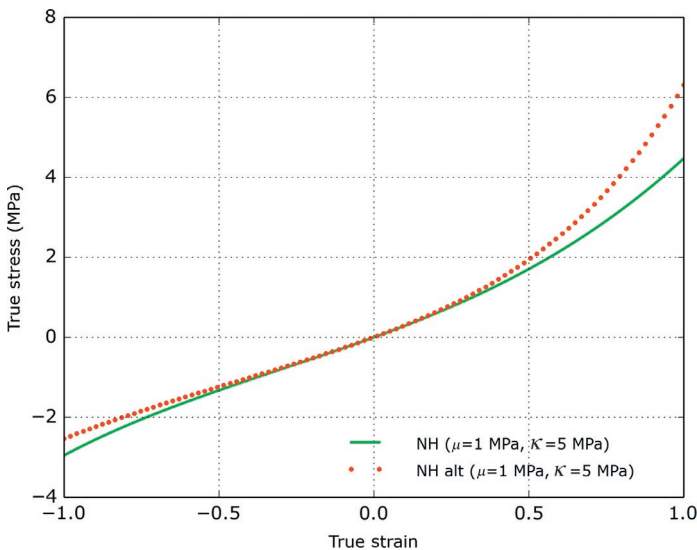
**Example: Alternative Formulation of the NH Model.**

The NH model can also be written using the following alternative equation:

$$\boldsymbol{\sigma} = \frac{\mu}{J} (\mathbf{b} - \mathbf{I}) + \kappa(J - 1)\mathbf{I}. \quad (5.85)$$

That is, the stress is not strictly divided into deviatoric and volumetric parts. For cases when the bulk modulus  $\kappa$  is large the standard NH model and this alternative NH model give similar predictions, but the smaller the bulk modulus the more different the two models become.

Figure 5.14 shows the stress-strain predictions from the standard NH model and the alternative NH model. In this case the shear modulus was taken to be  $\mu = 1$  MPa, and the bulk modulus was taken as  $\kappa = 5$  MPa.



**Figure 5.14** Comparison between predictions from the standard and an alternative NH model formulations.

### 5.3.7 Mooney-Rivlin Model

The Mooney-Rivlin (MR) model is an extension of the NH model that attempts to improve the accuracy by including a linear dependence on  $I_2^*$  in the Helmholtz free energy per unit reference volume:

$$\Psi(C_{10}, C_{01}, \kappa) = C_{10} (I_1^* - 3) + C_{01} (I_2^* - 3) + \frac{\kappa}{2} (J - 1)^2. \quad (5.86)$$

As illustrated in this equation, the compressible version of the MR model requires three material parameters:  $C_{10}$ ,  $C_{01}$ , and  $\kappa$ . Using Equation (5.41) it can be shown that the Cauchy stress for the Mooney-Rivlin model is given by:

$$\boldsymbol{\sigma} = \frac{2}{J} (C_{10} + C_{01} I_1^*) \mathbf{b}^* - \frac{2C_{01}}{J} (\mathbf{b}^*)^2 + \left[ \kappa (J - 1) - \frac{2I_1^* C_{10}}{3J} - \frac{4I_2^* C_{01}}{3J} \right] \mathbf{I}. \quad (5.87)$$

For the incompressible version of the MR model ( $\kappa \rightarrow \infty$ ), the Cauchy stresses in uniaxial, planar, and equibiaxial deformations are given by the following expressions:

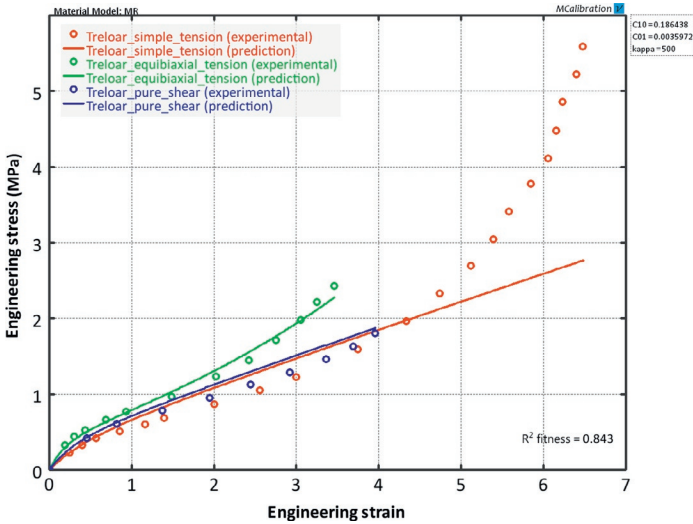
$$\sigma_{\text{uniax}} = 2 \left( \lambda^2 - \frac{1}{\lambda} \right) \left[ C_{10} + \frac{C_{01}}{\lambda} \right], \quad (5.88)$$

$$\sigma_{\text{planar}} = 2 \left( \lambda^2 - \frac{1}{\lambda^2} \right) [C_{10} + C_{01}], \quad (5.89)$$

$$\sigma_{\text{biaxial}} = 2C_{10} \left( \lambda^2 - \frac{1}{\lambda^4} \right) + 2C_{01} \left( \lambda^4 - \frac{1}{\lambda^2} \right). \quad (5.90)$$

The accuracy of the Mooney-Rivlin model to predict the behavior of elastomers is demonstrated in [Figure 5.15](#) by comparison to vulcanized natural rubber data of Treloar [16].

This figure shows that the Mooney-Rivlin model can improve the predictions of the NH model. However, the improvements can come with a price—having a negative  $C_{01}$  term, although often improving the predictions in one loading mode, can at the same time make the model unstable at finite deformations in a different



**Figure 5.15** Comparison between experimental data from Treloar [16] and predictions from the incompressible Mooney-Rivlin material model.

loading mode (see Section 5.8.2). One example of this is shown in Figure 5.16.

For incompressible uniaxial loading the Mooney-Rivlin material model can be implemented in Python using the following code:

Python Code: “MR\_incompressible\_uniaxial.py”

```
from pylab import *

def MR(strain, params):
    """Mooney-Rivlin hyperelastic model.
    Incompressible uniaxial loading.
    This function is using true stress and strain"""
    C10 = params[0]
    C01 = params[1]
    lam = exp(strain)
    return 2 * (lam*lam - 1/lam) * (C10 + C01/lam)

strain = linspace(0, 0.8)
params = [1.0, 0.1]
stress = MR(strain, params)
plot(strain, stress, 'r-')
show()
```

The following code example shows one way to implement the Mooney-Rivlin material model for *compressible* uniaxial loading.



**Additional Python Code to “Polymer\_Mechanics\_Chap05.py”**

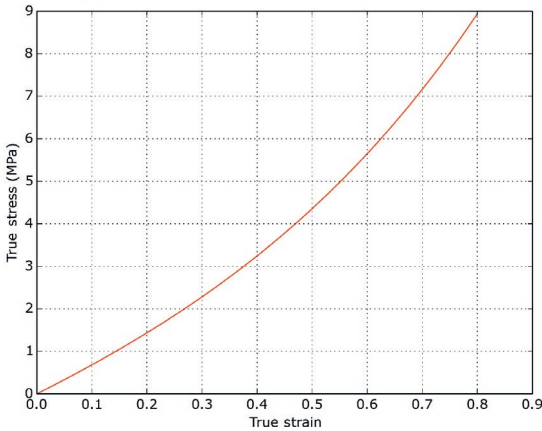
```
def MR_3D(stretch, param):
    """Mooney-Rivlin. 3D loading specified by stretches.
    param: [C10, C01, kappa]"""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    bstar2 = dot(bstar, bstar)
    I1s = trace(bstar)
    I2s = 0.5 * (I1s**2 - trace(bstar2))
    C10 = param[0]
    C01 = param[1]
    kappa = param[2]
    return 2/J*(C10+C01*I1s)*bstar - 2*C01/J*bstar2 + \
        (kappa*(J-1) - 2*I1s*C10/(3*J) - 4*I2s*C01/(3*J))*eye(3)
```

**Python Code: “MR\_compressible\_uniaxial.py”**

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

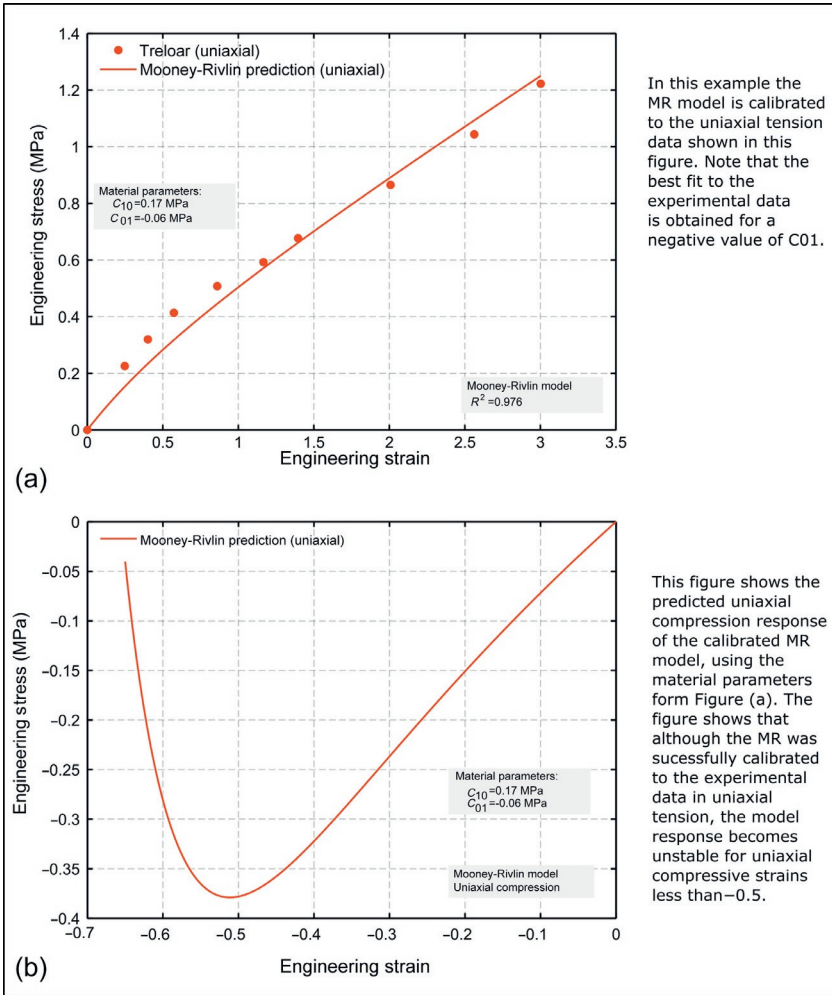
trueStrain = linspace(0, 0.8, 100)
trueStress = uniaxial_stress(MR_3D, trueStrain, [1.0, 0.1, 100])

plot(trueStrain, trueStress, 'r-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

**Figure Created by Python Code:**

### 5.3.8 Yeoh Model

The Yeoh model [24] is based on a Helmholtz free energy that is a third-order polynomial in  $I_1^*$ , with no dependence on  $I_2^*$ . Due to the higher order  $I_1^*$  terms, this model will enable more accurate predictions than the NH model, and at the same time potentially



**Figure 5.16** Example illustrating stability problems with the Mooney-Rivlin model.

avoid some of the stability issues of the Mooney-Rivlin model. One convenient way to write the Helmholtz free energy per unit reference volume for a compressible version of the Yeoh model is as follows:

$$\Psi(C_{10}, C_{20}, C_{30}, \kappa) = C_{10} (I_1^* - 3) + C_{20} (I_1^* - 3)^2 + C_{30} (I_1^* - 3)^3 + \frac{\kappa}{2} (J - 1)^2. \quad (5.91)$$

The model requires four material parameters:  $C_{10}$ ,  $C_{20}$ ,  $C_{30}$ , and  $\kappa$ . Note that some FE programs also use higher order terms to represent the energy from volumetric deformations. As was discussed in Section 2.2.9, these higher order terms rarely play a role in the accuracy of the model predictions.

One of the main motivations for the Yeoh model is that for most elastomers the Helmholtz free energy is much weaker dependent on the invariant  $I_2^*$  than the first invariant  $I_1^*$  [24–26]. Also, it turns out to be difficult to experimentally determine the dependence of the Helmholtz free energy on the  $I_2^*$  term. Based on these arguments, Yeoh [24] suggested that it is reasonable to neglect the dependence on  $I_2^*$  altogether. It has also been shown that by neglecting the  $I_2^*$  dependence it becomes easier to ensure that a hyperelastic model is Drucker stable [24] (see Section 5.8.2).

Using Equation (5.41), it can be shown that the Cauchy stress for the compressible Yeoh model is given by:

$$\boldsymbol{\sigma} = \frac{2}{J} \left\{ C_{10} + 2C_{20} (I_1^* - 3) + 3C_{30} (I_1^* - 3)^2 \right\} \text{dev}[\mathbf{b}^*] + \kappa (J - 1) \mathbf{I}. \quad (5.92)$$

For the incompressible version of the Yeoh model, the Cauchy stresses in uniaxial, planar, and equibiaxial deformations are given by the following expressions:

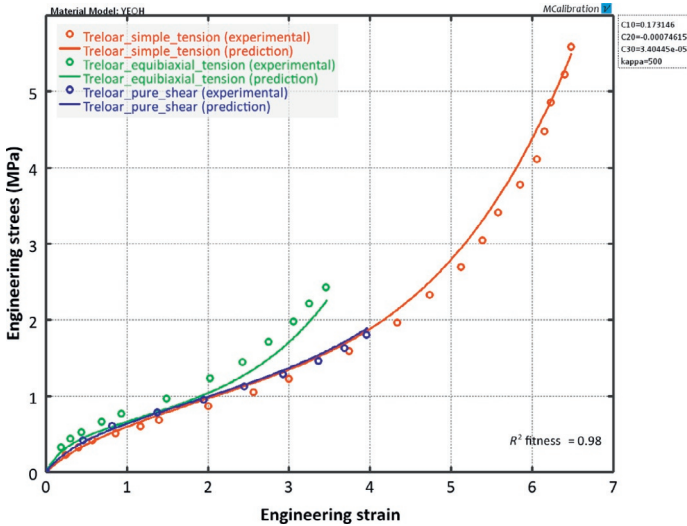
$$\sigma_{\text{uniax}} = 2 \left[ C_{10} + 2C_{20}(I_1 - 3) + 3C_{30}(I_1 - 3)^2 \right] \left( \lambda^2 - \frac{1}{\lambda} \right), \quad (5.93)$$

$$\sigma_{\text{planar}} = 2 \left[ C_{10} + 2C_{20}(I_1 - 3) + 3C_{30}(I_1 - 3)^2 \right] \left( \lambda^2 - \frac{1}{\lambda^2} \right), \quad (5.94)$$

$$\sigma_{\text{biax}} = 2 \left[ C_{10} + 2C_{20}(I_1 - 3) + 3C_{30}(I_1 - 3)^2 \right] \left( \lambda^2 - \frac{1}{\lambda^4} \right). \quad (5.95)$$

The accuracy of the Yeoh model to predict the behavior of elastomers is demonstrated in Figure 5.17.

This figure shows that the Yeoh model can improve the predictions of the NH model for the different loading modes,



**Figure 5.17** Comparison between experimental data from Treloar [16] and predictions from the incompressible Yeoh material model.

particularly at large deformations. For many elastomeric materials it has been shown [27] that a useful rule-of-thumb rule for the material parameters is to select  $C_{10}$  to be positive and  $C_{20} \approx -0.01C_{10}$ , and  $C_{30} \approx -0.01C_{20}$ .

For incompressible uniaxial loading the Yeoh material model can be implemented in Python using the following code:

Python Code: "Yeoh\_incompressible\_uniaxial.py"

```
from pylab import *

def Yeoh(strain, params):
    """Yeoh hyperelastic model.
    Incompressible uniaxial loading.
    This function is using true stress and strain"""
    C10 = params[0]
    C20 = params[1]
    C30 = params[2]
    lam = exp(strain)
    I1 = lam**2 + 2/lam
    return 2 * (lam*lam - 1/lam) * (C10 + 2*C20*(I1-3) + 3*C30*(I1-3)**2)

strain = linspace(0, 0.8)
params = [1.0, -0.1, 0.01]
stress = Yeoh(strain, params)
plot(strain, stress, 'r-')
show()
```

The following code example shows one way to implement the Yeoh material model for *compressible* uniaxial loading.

**Additional Code to “Polymer\_Mechanics\_Chap05.py”:**

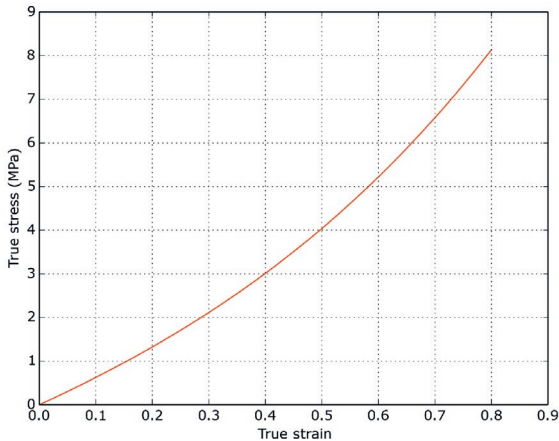
```
def Yeoh_3D(stretch, param):
    """Yeoh. 3D loading specified by stretches.
    param: [C10, C20, C30, kappa]. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    I1s = trace(bstar)
    return 2/J*(param[0] + 2*param[1]*(I1s-3) + \
        3*param[2]*(I1s-3)**2)*devbstar + param[3]*(J-1) * eye(3)
```

**Python Code: “Yeoh\_compressible\_uniaxial.py”**

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

trueStrain = linspace(0, 0.8, 100)
trueStress = uniaxial_stress(Yeoh_3D, trueStrain, \
    [1.0, -0.01, 1e-4, 100])

plot(trueStrain, trueStress, 'r-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

**Figure Created by Python Code:**

### 5.3.9 Polynomial in $I_1$ and $I_2$ Model

A generalization of the NH, Mooney-Rivlin, and the Yeoh models can be obtained by taking the Helmholtz free energy to be a polynomial expansion in terms of  $I_1^*$  and  $I_2^*$ . One common way to express this series expansion is as follows<sup>10</sup>:

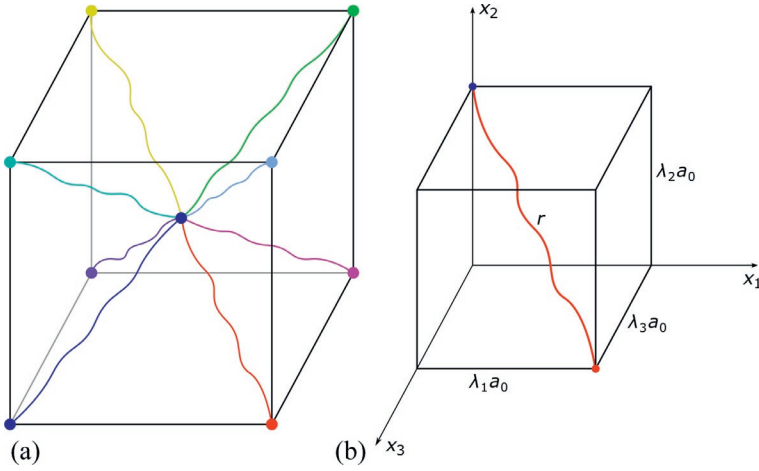
$$\Psi(C_{10}, C_{01}, \dots; D_1, D_2, \dots) = \sum_{i+j=1}^N C_{ij} (I_1^* - 3)^i (I_2^* - 3)^j + \sum_{i=1}^N \frac{1}{D_i} (J - 1)^{2i}. \quad (5.96)$$

This expression for the Helmholtz free energy is quite general and encompasses the NH, Mooney-Rivlin, and Yeoh models, but its generality also makes it difficult to determine the best set of material parameters that give accurate and robust predictions in multiaxial loading cases. This polynomial representation is therefore not as commonly used as the simpler models with a fixed number of terms.

### 5.3.10 Eight-Chain Model

The eight-chain (EC) model by Arruda and Boyce [28] is a hyperelasticity model that is motivated by the deformation behavior of the microstructure of elastomers. The basic assumption of the EC model is that the macromolecules (also called chain molecules) on average are located along the diagonals of a unit cell located in principal stretch space as illustrated in Figure 5.18. The side lengths of the unit cell in the reference state are denoted by  $a_0$ , and the undeformed chain length by  $r_0$ . It then directly follows that  $r_0 = a_0\sqrt{3}$ . Further, the macromolecules are taken to be freely jointed with  $n$  rigid links each of length  $l$ . For this FJC model the average end-to-end distance in the absence of an external force field is  $l\sqrt{n}$  [21]. By defining  $\lambda_1^*$ ,  $\lambda_2^*$ , and  $\lambda_3^*$  to be the applied principal distortional stretches, the effective distortional chain length becomes  $r = a_0[(\lambda_1^*)^2 + (\lambda_2^*)^2 + (\lambda_3^*)^2]^{1/2}$ , giving the effective distortional chain stretch

<sup>10</sup>This is the expression that is used, for example, by the FE program Abaqus [27].



**Figure 5.18** (a) Eight chain molecules are located in the unit cell. (b) Volume element for one chain molecule.

$$\bar{\lambda}^* = \left[ \frac{(\lambda_1^*)^2 + (\lambda_2^*)^2 + (\lambda_3^*)^2}{3} \right]^{1/2} = \sqrt{\frac{\text{tr } \mathbf{C}^*}{3}} = \sqrt{\frac{\text{tr } \mathbf{b}^*}{3}} = \sqrt{\frac{I_1^*}{3}}, \quad (5.97)$$

where  $\mathbf{b}^* = (J)^{-2/3} \mathbf{b}$ . This shows that the distortional chain stretch is in fact a function only of the first invariant  $I_1^*$ .

Based on this physically motivated model, an *eight-chain material* is defined as an isotropic thermoelastic material whose Helmholtz free energy per unit reference volume,  $\Psi$ , only depends on the two deformation invariants  $\bar{\lambda}^*(\mathbf{b}^*) = [\text{tr}(\mathbf{b}^*)/3]^{1/2}$  and  $J = \det(\mathbf{F})$ , and the temperature  $\theta_0$ .

By noting that the effective chain stretch is related to the first invariant of  $\mathbf{b}^*$  through  $\bar{\lambda}^* = [I_1(\mathbf{b}^*)/3]^{1/2}$ , it follows that the Helmholtz free energy per unit reference volume can be written  $\Psi(\bar{\lambda}^*, J, \theta)$ , or alternatively  $\Psi(I_1^*, J, \theta_0)$  where  $I_1^* = I_1(\mathbf{b}^*) = \text{tr}(\mathbf{b}^*)$ . The Cauchy stress for an EC material can then be obtained from the continuum mechanics Equation (5.41) which in this case with no dependence on  $I_2^*$  can be simplified to

$$\boldsymbol{\sigma} = \frac{2}{J} \frac{\partial \Psi}{\partial I_1^*} \text{dev}[\mathbf{b}^*] + \frac{\partial \Psi}{\partial J} \mathbf{I}, \quad (5.98)$$

or when expressed in terms of the effective chain stretch  $\bar{\lambda}^*$

$$\boldsymbol{\sigma} = \frac{1}{J} \frac{1}{3\bar{\lambda}^*} \frac{\partial \Psi}{\partial \bar{\lambda}^*} \text{dev}[\mathbf{b}^*] + \frac{\partial \Psi}{\partial J} \mathbf{I}. \quad (5.99)$$

The Helmholtz free energy per unit reference volume ( $\Psi$ ) can be determined by first using the experimental observation that for elastomers the internal energy is typically not a function of the applied distortional stretch [16], i.e.  $e_0(J, \theta_0)$ . Hence, the functional form of the Helmholtz free energy has the form:

$$\Psi(\bar{\lambda}^*, J, \theta_0) = e_0(J) - \theta_0 \eta_0(\bar{\lambda}^*). \quad (5.100)$$

Note that the dependence on  $J$  in  $\eta_0(\bar{\lambda}^*, J)$  has been neglected due to the assumption of small volume change. The assumption of small volumetric deformations also enable the relationship between the pressure component of the Cauchy stress  $\boldsymbol{\sigma}$  and the volumetric deformation  $J$  to be taken as linear:

$$\boldsymbol{\sigma} : \mathbf{I} = \frac{\partial \Psi(\bar{\lambda}^*, J, \theta_0)}{\partial J} = \kappa(J - 1), \quad (5.101)$$

giving the internal energy  $e_0(J) = \kappa J(J/2 - 1)$ . The Cauchy stress can then be calculated from Equation (5.99) giving

$$\boldsymbol{\sigma} = \frac{-\theta_0}{3J\bar{\lambda}^*} \frac{\partial \eta_0(\bar{\lambda}^*)}{\partial \bar{\lambda}^*} \text{dev}[\mathbf{b}^*] + \kappa[J - 1] \mathbf{I}. \quad (5.102)$$

To completely specify the constitutive relationship it now only remains to determine how the entropy depends on the effective chain stretch. From the chain rule we get:

$$\frac{\partial \eta_0(r(\bar{\lambda}^*))}{\partial \bar{\lambda}^*} = \frac{\partial \eta_0}{\partial r} \frac{\partial r}{\partial \bar{\lambda}^*} = \frac{\partial \eta_0}{\partial r} \frac{1}{\frac{\partial}{\partial r} [rl\sqrt{n}]} = l\sqrt{n} \frac{\partial \eta_0}{\partial r}. \quad (5.103)$$

Thus it is sufficient to determine how the entropy of a single macromolecule depends on its end-to-end distance in order to finalize the constitutive equation. This derivation was discussed in detail in [Section 5.3.5](#).

By using Equation (5.70), the equation for the Cauchy stress (5.102) can now be written

$$\boldsymbol{\sigma} = \frac{Nk_B\theta}{3J} \frac{\lambda^{\text{lock}}}{\bar{\lambda}^*} \mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right) \text{dev}[\mathbf{b}^*] + \kappa[J - 1] \mathbf{I}, \quad (5.104)$$



where  $\lambda^{\text{lock}} \equiv nl$  is the maximum (fully extended) stretch that a molecule can be exposed to.

For the special case of incompressible uniaxial deformation Equation (5.104) simplifies to

$$\sigma_{\text{uniax}} = \frac{Nk_B\theta}{3J} \frac{\lambda^{\text{lock}}}{\bar{\lambda}^*} \mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right) \left[ \lambda^2 - \frac{1}{\lambda} \right], \quad (5.105)$$

and for simple shear defined by  $\mathbf{F} = \mathbf{I} + \gamma \mathbf{e}_1 \otimes \mathbf{e}_2$  the shear stress is given by

$$\sigma_{\text{shear}} = \frac{Nk_B\theta_0}{3J} \frac{\lambda^{\text{lock}}}{\bar{\lambda}^*} \mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right) \gamma, \quad (5.106)$$

where  $\bar{\lambda}^* = \sqrt{1 + \gamma^2/3}$ .

The initial shear modulus of the material is given by  $\mu = \left. \frac{\partial \sigma_{12}}{\partial \gamma} \right|_{\gamma=0}$  giving

$$\mu = \frac{Nk_B\theta_0}{3} \lambda^{\text{lock}} \mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right), \quad (5.107)$$

which when inserted in (5.104) gives the Cauchy stress for the EC model as

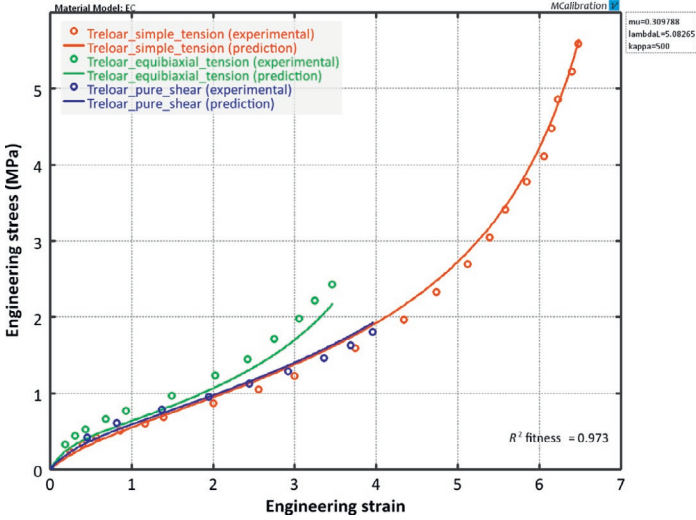
$$\sigma = \frac{\mu}{J\bar{\lambda}^*} \frac{\mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right)}{\mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right)} \text{dev}[\mathbf{b}^*] + \kappa[J - 1] \mathbf{I}. \quad (5.108)$$

In this equation  $[\mu, \lambda^{\text{lock}}, \kappa]$  are the material parameters.

For the incompressible version of the EC model, the Cauchy stresses in uniaxial, planar, and equibiaxial deformations are given by the following expressions:

$$\sigma_{\text{uniax}} = \frac{\mu}{\bar{\lambda}^*} \frac{\mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right)}{\mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right)} \left[ \lambda^2 - \frac{1}{\lambda} \right], \quad (5.109)$$

$$\sigma_{\text{planar}} = \frac{\mu}{\bar{\lambda}^*} \frac{\mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right)}{\mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right)} \left[ \lambda^2 - \frac{1}{\lambda^2} \right], \quad (5.110)$$



**Figure 5.19** Comparison between experimental data from Treloar [16] and predictions from the eight-chain material model.

$$\sigma_{\text{biax}} = \frac{\mu}{\lambda^*} \frac{\mathcal{L}^{-1}\left(\frac{\bar{\lambda}^*}{\lambda_{\text{lock}}}\right)}{\mathcal{L}^{-1}\left(\frac{1}{\lambda_{\text{lock}}}\right)} \left[ \lambda^2 - \frac{1}{\lambda^4} \right]. \quad (5.111)$$

The compressible EC model contains three material parameters: a shear modulus  $\mu$ , a limiting chain stretch  $\lambda_{\text{lock}}$ , and a bulk modulus  $\kappa$ . The EC model has no dependence on  $I_2$  and give predictions that are similar in accuracy to the Gent model [29]. The accuracy of the EC model to predict the behavior of elastomers is demonstrated in Figure 5.19 by comparison to vulcanized natural rubber data of Treloar [16].

The figure shows that the EC model in this case is more accurate than the NH model and the Mooney-Rivlin model, and almost as accurate as the Yeoh model. The figure also shows that the EC model slightly underpredicts the biaxial response. This can be expected since the model is  $I_1$ -based, and does not include any dependence on  $I_2$ .

For incompressible uniaxial loading the EC material model can be implemented into Matlab using the following code:

**Matlab Code:**

```
function [stress] = mat_EC(time, strain, params)
%mat_EC Arruda-Boyce Eight-Chain hyperelastic model
%Incompressible uniaxial loading
%This function is using true stress and strain
mu = params(1);
lambdaL = params(2);
lambda = exp(strain);
lambdaChain = sqrt((lambda.^2 + 2./lambda) / 3);
stress = mu./lambdaChain .* invLangevin(lambdaChain/lambdaL) ...
    ./ invLangevin(1/lambdaL) .* (lambda.^2 - 1./lambda);
end

function [res] = invLangevin(x)
%INVLANGEVIN implementation of the inverse langevin function
x(find(x>=1)) = 1-eps;
x(find(x<=-1)) = -1+eps;
res = zeros(size(x));
index = find(abs(x) < 0.839);
res(index) = 1.31435 * tan(1.59*x(index)) + 0.911249*x(index);
index = find(abs(x) >= 0.839);
res(index) = 1 ./ (sign(x(index)) - x(index));
end
```

For incompressible uniaxial loading the NH material model can be implemented in Python using the following code:

**Additional Code to “Polymer\_Mechanics\_Chap05.py”:**

```
def invLangevin(x):
    EPS = spacing(1)
    if type(x) == float: # x is a scalar
        if x >= 1-EPS: x = 1 - EPS
        if x <= -1+EPS: x = -1 + EPS
        if abs(x) < 0.839:
            return 1.31435 * tan(1.59*x) + 0.911249*x
        return 1.0 / (sign(x) - x)
    # x is an array
    x[x >= 1-EPS] = 1 - EPS
    x[x <= -1+EPS] = -1 + EPS
    res = zeros(size(x))
    index = abs(x) < 0.839
    res[index] = 1.31435 * tan(1.59*x[index]) + 0.911249*x[index]
    index = abs(x) >= 0.839
    res[index] = 1.0 / (sign(x[index]) - x[index])
    return res
```

**Python Code: “EC\_incompressible\_uniaxial.py”**

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

def EightChain(trueStrain, params):
    """Arruda-Boyce eight-chain model.
    Incompressible uniaxial loading. Returns true stress."""
    mu = params[0]
    lambdaL = params[1]
    lam = exp(trueStrain)
    lamChain = sqrt((lam**2 + 2/lam)/3)
    return mu/lamChain * invLangevin(lamChain/lambdaL)/invLangevin(1/lambdaL) \
        * (lam*lam - 1/lam)

trueStrain = linspace(0, 0.8, 10)
trueStress = EightChain(trueStrain, [1.0, 3.0])

plot(trueStrain, trueStress, 'r-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

The following code example shows one way to implement the EC material model for *compressible* uniaxial loading.

**Additional Code to “Polymer\_Mechanics\_Chap05.py”:**

```
def EC_3D(stretch, param):
    """Eight-Chain. 3D loading specified by stretches.
    param: [mu, lambdaL, kappa]. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    lamChain = sqrt(trace(bstar)/3)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/(J*lamChain) * invLangevin(lamChain/param[1]) / \
        invLangevin(1/param[1]) * devbstar + param[2]*(J-1) * eye(3)
```

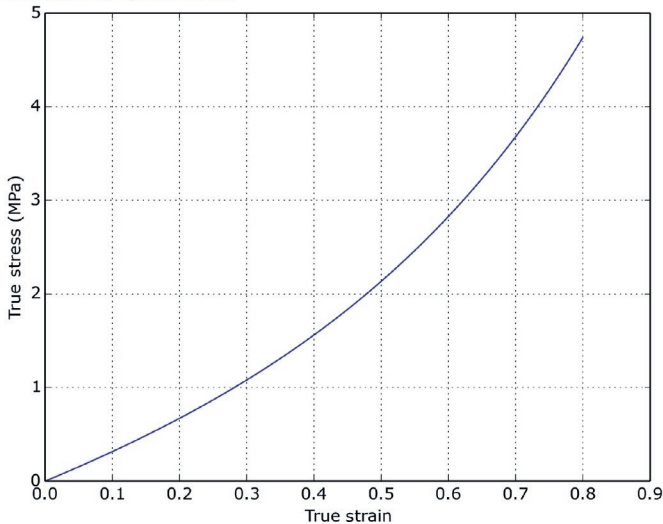
**Python Code: “EC\_compressible\_uniaxial.py”**

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

trueStrain = linspace(0, 0.8, 100)
trueStress = uniaxial_stress(EC_3D, trueStrain, [1.0, 3.0, 100])

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

**Figure Created by Python Code:**



### Example: Different Implementations for the Inverse Langevin Function.

The Langevin function is defined by:

$$\mathcal{L}(x) = \coth(x) - 1/x. \quad (5.112)$$

The inverse of this function is called the inverse Langevin function  $\mathcal{L}^{-1}(x)$  and cannot be expressed in elementary functions. As shown above, to numerically calculate the stress for the EC model it is necessary to evaluate the inverse Langevin function. This can be done by solving for  $x$  in Equation (5.112) using a numerical equation solver, such as the Newton method. This approach, however, is computationally expensive and multiple alternative faster approximations have been developed.

One method that was developed by Arruda and Boyce [28] is to express the inverse Langevin function as a series expansion:

$$\mathcal{L}(x) \approx 3x + \frac{9}{5}x^3 + \frac{297}{175}x^5 + \frac{1539}{875}x^7. \quad (5.113)$$

The function can also be approximated using a Padé approach [30]:

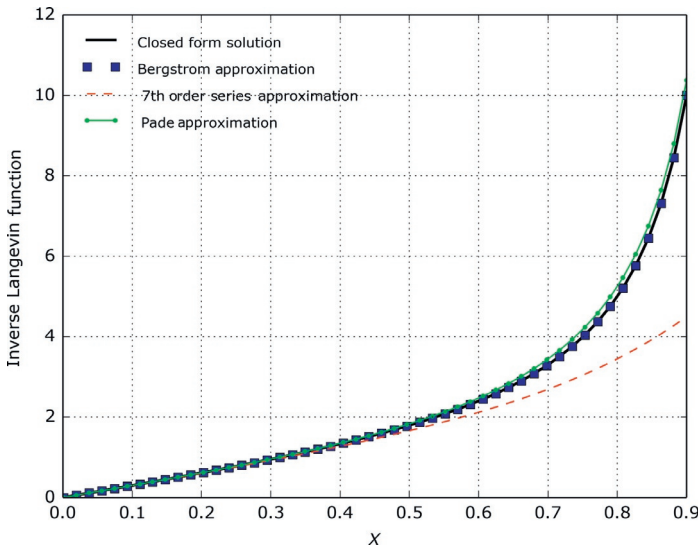
$$\mathcal{L}(x) \approx \frac{x(3 - x^2)}{1 - x^2}. \quad (5.114)$$

Yet another approach was developed by Bergstrom [23] who derived the functional response into two regions. For small  $x$  values the function was represented using a curve fit based on a tan-function, and for large  $x$  values the function was represented by its asymptotic form:

$$\mathcal{L}^{-1}(x) \approx \begin{cases} 1.31446 \tan(1.58986x) + 0.91209x, & \text{if } |x| < 0.84136, \\ 1/(\text{sign}(x) - x), & \text{if } 0.84136 \leq |x| < 1. \end{cases} \quad (5.115)$$

These different approximations are compared in Figure 5.20, and a comparison between the accuracy of the approximations is given in Table 5.3.

The difference between these numerical approximations may seem more like an academic issue than an important practical concern, but the issue becomes important since different FE codes have decided to implement the inverse Langevin function in different ways. For example, Abaqus uses a series expansion, and the PolyUMod library [31] uses the more accurate Bergstrom solution approach.



**Figure 5.20** Comparison between different approaches for calculating the inverse Langevin function.

**Table 5.3** Comparison in Errors in the Different Approaches for Calculating the Inverse Langevin Function

Approximation Method	Relative Error in Prediction at $x = 0.7$
Bergstrom Approximation	0.06%
5-term Series Expansion	18.5%
Padé Approximation	4.3%

In other words, it is difficult to translate the parameters for the EC model from one FE solver to another. It may be necessary to recalibrate the model for each specific FE package.

### Example: Gaussian Chains.

The constitutive relationship for the EC model for the case of Gaussian chains (instead of Langevin chains) can be directly obtained by replacing the inverse Langevin function by the first term in its series expansion:

$$\mathcal{L}^{-1}(x) = 3x + \frac{9}{5}x^3 + \frac{297}{175}x^5 + \mathcal{O}(x^7),$$

giving for uniaxial tension

$$\boldsymbol{\sigma} = \frac{\mu_0}{J} \operatorname{dev}[\mathbf{b}^*] + \kappa[J - 1]\mathbf{I}, \quad (5.116)$$

which is identical to the NH model (Equation (5.74)).

### 5.3.11 Ogden Model

The Ogden model [12] is a very general hyperelasticity model with a Helmholtz free energy per reference volume that is expressed in terms of the applied principal stretches. The Helmholtz free energy for the Ogden model can be written in different ways. One common compressible representation [27] is given in Equation (5.117). In this equation the volumetric response is written in terms of  $D_i$  parameters instead of the bulk modulus terms.

$$\begin{aligned} \Psi(\lambda_1^*, \lambda_2^*, \lambda_3^*) &= \sum_{k=1}^N \frac{2\mu_k}{\alpha_k^2} \left( (\lambda_1^*)^{\alpha_k} + (\lambda_2^*)^{\alpha_k} + (\lambda_3^*)^{\alpha_k} - 3 \right) \\ &\quad + \sum_{k=1}^N \frac{1}{D_k} (J - 1)^{2k}. \end{aligned} \quad (5.117)$$

This general form of the Helmholtz free energy makes the model powerful but can also complicate the selection of an appropriate set of material parameters that give stable predictions of general deformation states.

The principal stresses  $\sigma_i$ ,  $i \in [1, 2, 3]$ , for the Ogden model, are given by:

$$\begin{aligned} \sigma_i &= \frac{2}{J} \sum_{k=1}^N \frac{\mu_k}{\alpha_k} \left( (\lambda_i^*)^{\alpha_k} - \frac{1}{3} [(\lambda_1^*)^{\alpha_k} + (\lambda_2^*)^{\alpha_k} + (\lambda_3^*)^{\alpha_k}] \right) \\ &\quad + \sum_{k=1}^N \frac{2k}{D_k} (J - 1)^{2k-1}. \end{aligned} \quad (5.118)$$

The stresses from the incompressible Ogden model in uniaxial loading, planar loading, and biaxial loading are given by:

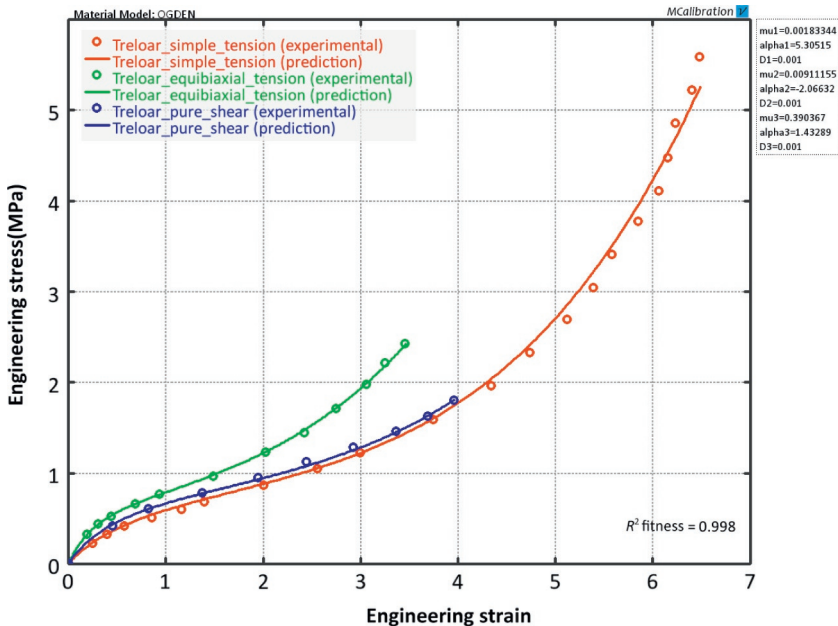
$$\sigma_{\text{uniax}} = \sum_{k=1}^N \frac{2\mu_k}{\alpha_k} \left[ \lambda^{\alpha_k} - \left( \frac{1}{\sqrt{\lambda}} \right)^{\alpha_k} \right], \quad (5.119)$$

$$\sigma_{\text{planar}} = \sum_{k=1}^N \frac{2\mu_k}{\alpha_k} \left[ \lambda^{\alpha_k} - \left( \frac{1}{\lambda} \right)^{\alpha_k} \right], \quad (5.120)$$

$$\sigma_{\text{biax}} = \sum_{k=1}^N \frac{2\mu_k}{\alpha_k} \left[ \lambda^{\alpha_k} - \left( \frac{1}{\lambda^2} \right)^{\alpha_k} \right]. \quad (5.121)$$

It is interesting to note that if  $N = 1$  and  $\alpha_2 = 1$ , then the Ogden model becomes equal to the NH model.

The accuracy of the Ogden model to predict the behavior of elastomers is demonstrated in [Figure 5.21](#) by comparison to



**Figure 5.21** Comparison between experimental data from Treloar [16] and predictions from a three-term Ogden model.



vulcanized natural rubber data of Treloar [16]. The figure shows that a 3-term Ogden model in this case is more accurate than the NH model and the Mooney-Rivlin model, but not as accurate as the Yeoh model or the EC model.

For incompressible uniaxial loading the Ogden material model can be implemented into Matlab using the following code:

**Matlab Code: "Ogden\_incompressible\_uniaxial.m"**

```
function [stress] = mat_Ogden(time, strain, params)
%mat_Ogden Ogden hyperelastic model
%Incompressible uniaxial loading
%This function is using true stress and strain
mu = params(1:2:end);
alpha = params(2:2:end);
lambda = exp(strain);
for i = 1 : length(lambda)
    stress(i) = sum(2*mu./alpha .* (lambda(i).^alpha - ...
        1./(lambda(i).^(alpha/2))));
end
end
```

For incompressible uniaxial loading the Ogden material model can also be implemented in Python using the following code:

**Python Code: "Ogden\_incompressible\_uniaxial.py"**

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

def Ogden(trueStrain, muVec, alphaVec):
    """Ogden model. Incompressible uniaxial loading.
    Returns true stress."""
    lam = exp(trueStrain)
    res = 0
    for i in range(len(muVec)):
        mu = muVec[i]
        alpha = alphaVec[i]
        res = res + 2*mu/alpha * (lam**alpha - (1/sqrt(lam))**alpha)
    return res

trueStrain = linspace(0, 0.8, 100)
trueStress = Ogden(trueStrain, [1.0, 3.0], [3.0, 0.4])

plot(trueStrain, trueStress, 'r-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

The following code example shows one way to implement the Ogden material model for *compressible* uniaxial loading.

Additional Code to “Polymer\_Mechanics\_Chap05.py”:

```
def Ogden_3D(stretch, param):
    """Ogden model. 3D loading specified by stretches.
    param: [mu1, mu2, ..., alpha1, alpha2, kappa].
    Returns true stress."""
    J = stretch[0] * stretch[1] * stretch[2]
    lam = J**(-1/3) * stretch
    N = round((len(param)-1)/2)
    mu = param[0:N]
    alpha = param[N:2*N]
    kappa = param[-1]
    Stress = kappa*(J-1)*eye(3)
    for i in range(N):
        fac = (2/J) * mu[i] / alpha[i]
        tmp = (lam[0]**alpha[i]+lam[1]**alpha[i]+lam[2]**alpha[i])/3
        Stress[0,0] = Stress[0,0] + fac * (lam[0]**alpha[i] - tmp)
        Stress[1,1] = Stress[1,1] + fac * (lam[1]**alpha[i] - tmp)
        Stress[2,2] = Stress[2,2] + fac * (lam[2]**alpha[i] - tmp)
    return Stress
```

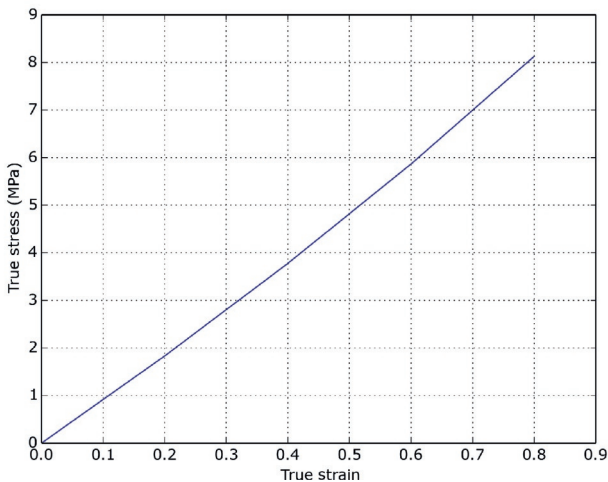
Python Code: “Ogden\_compressible\_uniaxial.py”

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

param = [1.0, 2.0, 1.1, 0.4, 100.0]
trueStrain = linspace(0, 0.8, 5)
trueStress = uniaxial_stress(Ogden_3D, trueStrain, param)

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

Figure Created by Python Code:



### 5.3.12 Gent Model

The Gent model [32] is an extension of the NH model that aims at better characterizing the response of elastomer-like materials at large deformations. The Helmholtz free energy per unit reference volume for the Gent model is given by<sup>11</sup> [32]:

$$\Psi(I_1, J) = -\frac{\mu}{2} J_m \ln \left( 1 - \frac{I_1^* - 3}{J_m} \right) + \frac{\kappa}{2} [J - 1]^2. \quad (5.122)$$

This expression contains three material parameters: the shear modulus  $\mu$ , a dimensionless parameter  $J_m$  that controls the limited chain extensibility at large applied strains, and the bulk modulus  $\kappa$ . The Cauchy stress for the Gent model is given by

$$\boldsymbol{\sigma} = \frac{\mu}{J} \cdot \frac{1}{1 - \frac{I_1^* - 3}{J_m}} \text{dev}[\mathbf{b}^*] + \kappa [J - 1] \mathbf{I}. \quad (5.123)$$

In the Gent model the first invariant  $I_1^*$  is always less than  $J_m + 3$ , and in the limit as  $J_m \rightarrow \infty$ , the model becomes identical to the NH model.

The Cauchy stresses for the incompressible Gent model in uniaxial loading, planar loading, and biaxial loading are given by:

$$\sigma_{\text{uniax}} = \mu \left( \lambda^2 - \frac{1}{\lambda} \right) \frac{J_m}{J_m - (\lambda^2 + 2/\lambda - 3)}, \quad (5.124)$$

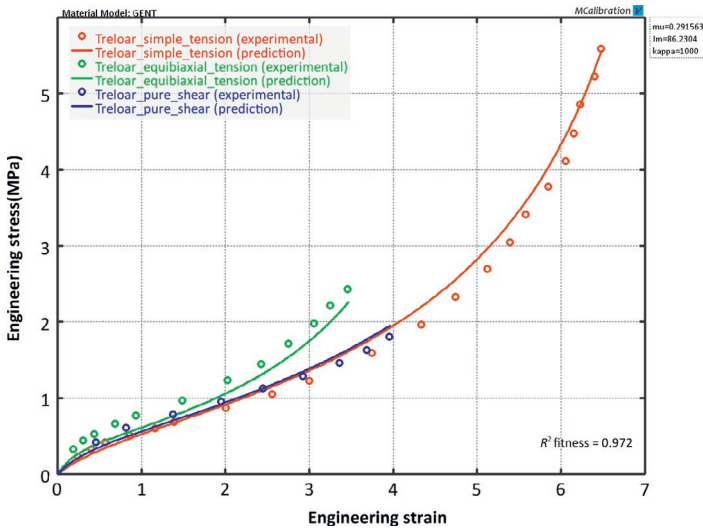
$$\sigma_{\text{planar}} = \mu \left( \lambda^2 - \frac{1}{\lambda^2} \right) \frac{J_m}{J_m - (\lambda^2 + 2/\lambda - 3)}, \quad (5.125)$$

$$\sigma_{\text{biax}} = \mu \left( \lambda^2 - \frac{1}{\lambda^4} \right) \frac{J_m}{J_m - (\lambda^2 + 2/\lambda - 3)}. \quad (5.126)$$

The accuracy of the Gent model to predict the behavior of elastomers is demonstrated in Figure 5.22 by comparison to vulcanized natural rubber data of Treloar [16].

For incompressible uniaxial loading the Gent material model can be implemented into Matlab using the following code:

<sup>11</sup>The original version of the Gent model was incompressible. Here we have added a simple compressibility term. Other forms of the compressibility can be chosen if desired.



**Figure 5.22** Comparison between experimental data from Treloar [16] and predictions from the Gent model.

Matlab Code: “Gent\_incompressible\_uniaxial.m”

```
function [stress] = mat_Gent(time, strain, params)
%mat_Gent Gent hyperelastic model
%Incompressible uniaxial loading
%This function is using true stress and strain
mu = params(1);
Jm = params(2);
lambda = exp(strain);
stress = Jm*mu * (lambda.^2 - 1./lambda) ./ \
    (Jm - (lambda.^2 + 2./lambda - 3));
end
```

For incompressible uniaxial loading the Gent material model can also be implemented in Python using the following code:

Python Code: “Gent\_incompressible\_uniaxial.py”

```
from pylab import *

def Gent(trueStrain, params):
    """Gent model. Incompressible uniaxial loading. Returns true stress."""
    mu = params[0]
    Jm = params[1]
    lam = exp(trueStrain)
    return mu*Jm / (Jm - lam*lam - 2/lam + 3) * (lam*lam - 1/lam)

trueStrain = linspace(0, 0.8, 100)
trueStress = Gent(trueStrain, [0.29, 86.0])

plot(trueStrain, trueStress, 'r-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

The following code example shows one way to implement the Gent material model for *compressible* uniaxial loading.

**Additional Code to “Polymer\_Mechanics\_Chap05.py”:**

```
def Gent_3D(stretch, param):
    """Gent. 3D loading specified by stretches.
    param: [mu, Jm, kappa]. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    I1s = trace(bstar)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/ J / (1 - (I1s-3)/param[1]) * devbstar + \
        param[2]*(J-1) * eye(3)
```

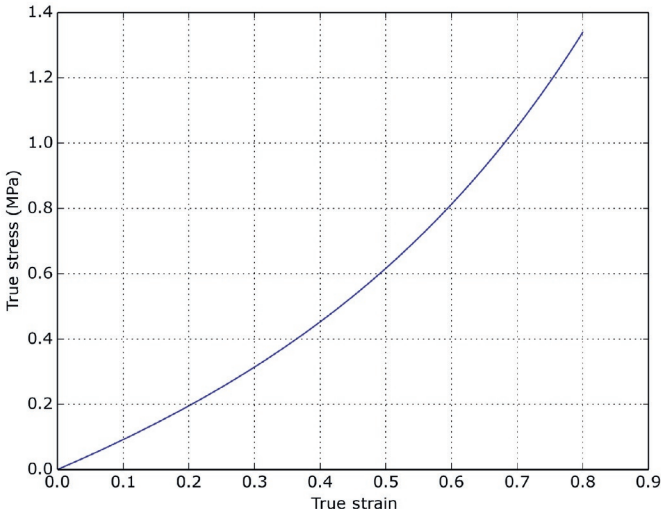
**Python Code: “Gent\_compressible\_uniaxial.py”**

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

trueStrain = linspace(0, 0.8, 100)
trueStress = uniaxial_stress(Gent_3D, trueStrain, [0.29, 86.0, 100])

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

**Figure Created by Python Code:**



### 5.3.13 Horgan and Saccomandi Model

Horgan and Saccomandi have developed a model [33, 34] that is an extension of the Gent model. This model, henceforth

called the HS model, also involves three material parameters. The main difference between the Gent and the HS models is that the latter depends on both the first and second strain-invariants and thus might in some cases agree more closely with experimental data. In addition, the limiting chain parameter in the HS model is related directly to the maximum allowable stretch and so is more physically accessible than its counterpart for the Gent model that depends on a maximum value of the first invariant [33]. The Helmholtz free energy per unit reference volume of a compressible version of the HS is given by:

$$\Psi(I_1^*, I_2^*, J) = -\frac{\mu}{2} \lambda_{\max}^2 \ln \left[ \frac{\lambda_{\max}^6 - \lambda_{\max}^4 I_1^* + \lambda_{\max}^2 I_2^* - 1}{(\lambda_{\max}^2 - 1)^3} \right] + \frac{\kappa}{2} [J - 1]^2, \quad (5.127)$$

where  $\mu$  is the shear modulus,  $\lambda_{\max}$  is the limiting chain stretch, and  $\kappa$  is the bulk modulus.

The Cauchy stress for the compressible HS model is given by

$$\boldsymbol{\sigma} = \frac{\mu \lambda_{\max}^4}{J} \frac{[\lambda_{\max}^2 - I_1^*] \mathbf{b}^* + (\mathbf{b}^*)^2 - \frac{1}{3} [\lambda_{\max}^2 I_1^* - 2I_2^*] \mathbf{I}}{\lambda_{\max}^6 - \lambda_{\max}^4 I_1^* + \lambda_{\max}^2 I_2^* - 1} + \kappa [J - 1] \mathbf{I}. \quad (5.128)$$

In this model  $\max(\lambda_1, \lambda_2, \lambda_3) < \lambda_{\max}$ . In the limit as  $\lambda_{\max} \rightarrow \infty$ , the model becomes identical to the NH model. The Cauchy stresses of the incompressible version of the HS model in uniaxial, planar, and equibiaxial deformations are given by the following expressions:

$$\sigma_{\text{uniax}} = \mu \lambda_{\max}^4 \frac{\lambda^3 - 1}{(\lambda \lambda_{\max}^2 - 1)(\lambda_{\max}^2 - \lambda^2)}, \quad (5.129)$$

$$\sigma_{\text{planar}} = \mu \lambda_{\max}^4 \frac{\lambda^4 - 1}{(\lambda^2 \lambda_{\max}^2 - 1)(\lambda_{\max}^2 - \lambda^2)}, \quad (5.130)$$

$$\sigma_{\text{biax}} = \mu \lambda_{\max}^4 \frac{\lambda^6 - 1}{(\lambda^4 \lambda_{\max}^2 - 1)(\lambda_{\max}^2 - \lambda^2)}. \quad (5.131)$$

The accuracy of the HS model to predict the behavior of elastomers is demonstrated in [Figure 5.23](#).

One nice feature of the HS model is that the model is unconditionally stable if  $\mu > 0$  and  $\lambda_{\max} > 1$ .

The following code example shows one way to implement the HS material model for *compressible* uniaxial loading.

Additional Code to “Polymer\_Mechanics\_Chap05.py”:

```
def HS_3D(stretch, param):
    """Horgan-Saccomandi. 3D loading specified by stretches.
    param: mu, lamMax, kappa. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    bstar2 = dot(bstar, bstar)
    I1s = trace(bstar)
    I2s = 0.5 * (I1s**2 - trace(bstar2))
    mu = param[0]
    lamM = param[1]
    kappa = param[2]
    fac = mu * lamM**4 / J
    den = lamM**6 - lamM**4 * I1s + lamM**2 * I2s - 1
    return fac/den * ((lamM**2 - I1s)*bstar + bstar2 - \
        (lamM**2*I1s-2*I2s)/3*eye(3)) + kappa*(J-1) * eye(3)
```

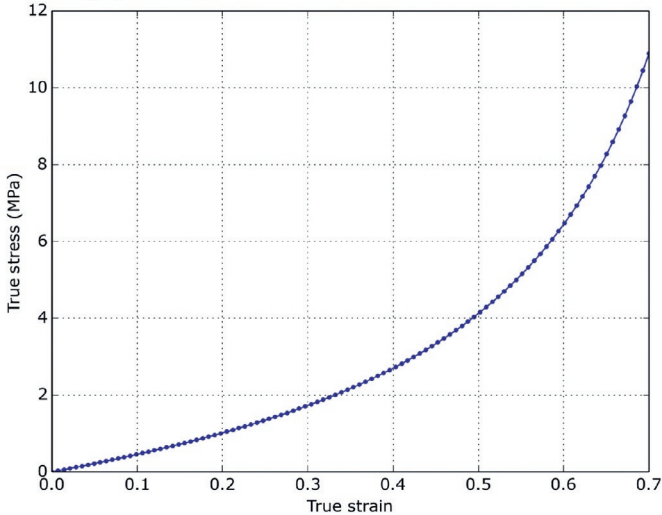
Python Code: “HS\_compressible\_uniaxial.py”

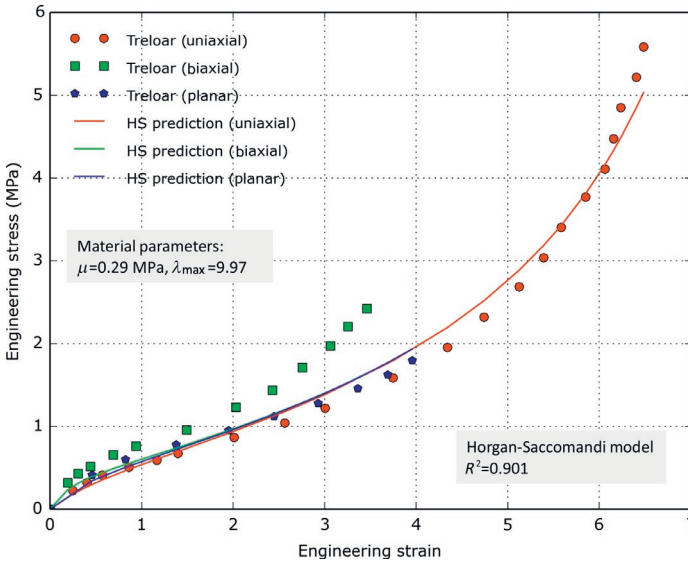
```
from pylab import *
from Polymer_Mechanics_Chap05 import *

trueStrain = linspace(0, 0.7, 100)
trueStress = uniaxial_stress(HS_3D, trueStrain, [1.0, 2.5, 1000])

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

Figure Created by Python Code:





**Figure 5.23** Comparison between experimental data from Treloar [16] and predictions from the Horgan-Saccomandi model.

### 5.3.14 Knowles Model

The Knowles hyperelastic model [35] is a simple and versatile hyperelastic material model that is an extension of the NH model, but is not a built-in native feature of most FE software. The Helmholtz free energy per unit reference volume of a compressible version of the Knowles model [36] is given by:

$$\Psi(I_1^*, J) = -\frac{\mu}{2b} \left[ \left( 1 + \frac{b}{n} (I_1^* - 3) \right)^n - 1 \right] + \frac{\kappa}{2} [J - 1]^2, \quad (5.132)$$

where  $\mu$  is the shear modulus,  $n$  is a hardening parameter,  $b$  is a shape parameter, and  $\kappa$  is the bulk modulus. From this equation it is clear that if  $n = 1$  then the Knowles model becomes equal to the NH model. The Knowles model was originally developed for predicting the crack tip response of incompressible elastic solids [35], but it has recently also been used to study the degradation response of poly-L-lactic acid (PLLA) materials [37].



The Cauchy stress for the compressible Knowles model is:

$$\boldsymbol{\sigma} = \frac{\mu}{J} \left( 1 + \frac{b}{n} (I_1^* - 3) \right)^{n-1} \text{dev}[\mathbf{b}^*] + \kappa (J - 1) \mathbf{I}. \quad (5.133)$$

The Cauchy stresses for the incompressible Knowles model in uniaxial loading, planar loading, and biaxial loading are given by:

$$\sigma_{\text{uniax}} = \mu \left( 1 + \frac{b}{n} (\lambda^2 + 2/\lambda - 3) \right)^{n-1} \cdot \left[ \lambda^2 - \frac{1}{\lambda} \right], \quad (5.134)$$

$$\sigma_{\text{planar}} = \mu \left( 1 + \frac{b}{n} (\lambda^2 + 1/\lambda^2 - 2) \right)^{n-1} \cdot \left[ \lambda^2 - \frac{1}{\lambda^2} \right], \quad (5.135)$$

$$\sigma_{\text{biax}} = \mu \left( 1 + \frac{b}{n} (2\lambda^2 + 1/\lambda^4 - 3) \right)^{n-1} \cdot \left[ \lambda^2 - \frac{1}{\lambda^4} \right]. \quad (5.136)$$

The accuracy of the Knowles model to predict the behavior of elastomers is demonstrated in [Figure 5.24](#).

The following code example shows one way to implement the Knowles material model for *compressible* uniaxial loading.

Additional Code to “Polymer\_Mechanics\_Chap05.py”:

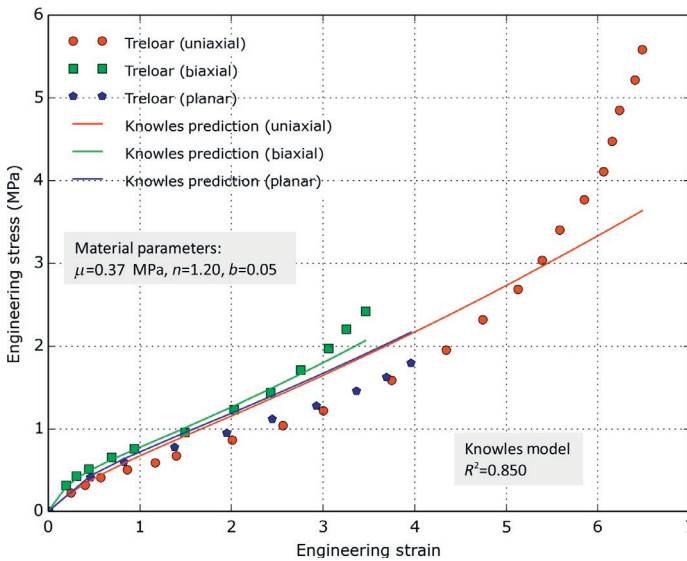
```
def Knowles_3D(stretch, param):
    """Knowles. 3D loading specified by stretches.
    param: mu, n, b, kappa. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    I1s = trace(bstar)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/J * (1+param[2]/param[1]*(I1s-3))**(param[1]-1) \
        * devbstar + param[3]*(J-1) * eye(3)
```

Python Code: “Knowles\_compressible\_uniaxial.py”

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

trueStrain = linspace(0, 0.7, 100)
trueStress = uniaxial_stress(Knowles_3D, trueStrain, \
    [1.0, 0.5, 0.5, 100])

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```



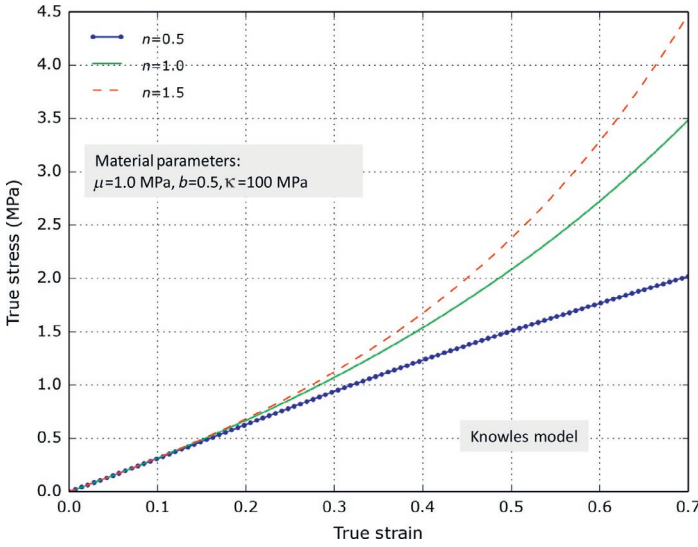
**Figure 5.24** Comparison between experimental data from Treloar [16] and predictions from the incompressible Knowles material model.

The influence of the parameters  $n$  and  $b$  on the predicted stress-strain response of the Knowles model are shown in Figures 5.25 and 5.26. These figures show that the  $n$  and  $b$  parameters strongly influence the predicted stress-strain response.

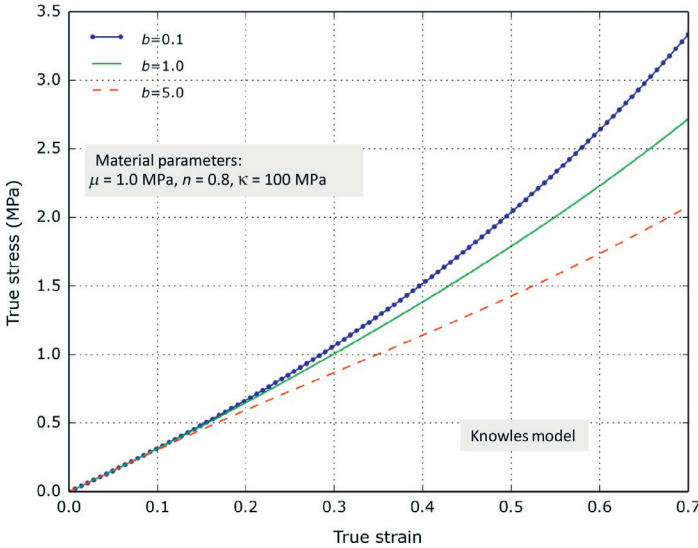
### 5.3.15 Response Function Hyperelasticity

All hyperelastic models that have been discussed so far in this chapter have been based on specific equations for the Helmholtz free energy per unit reference volume, where the energy equation contains material constants that specify the behavior of a specific material. Another approach is to specify the free energy directly based on experimental data in uniaxial loading [27, 38]. One advantage of this approach is that there are no adjustable material parameters that need to be found, instead the energy function, and as a result the stress calculations, are directly obtained from the provided experimental uniaxial test data.

Most major FE programs have a hyperelastic model of this type, for example, in Abaqus the model is called the



**Figure 5.25** Influence of parameter  $n$  on the stress-strain response for the Knowles model.

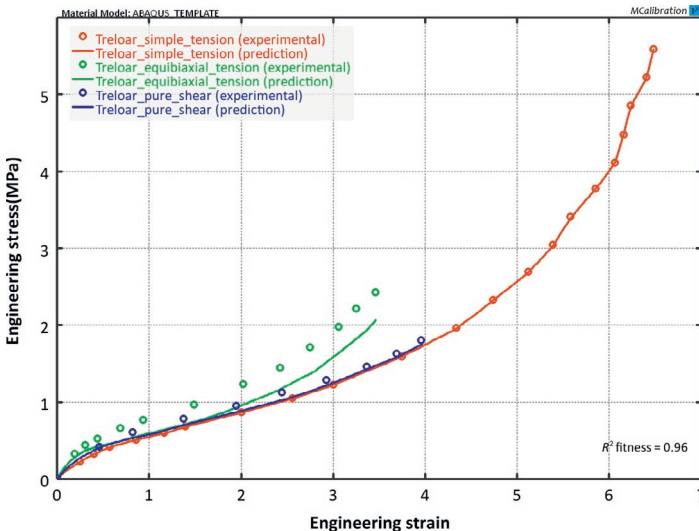


**Figure 5.26** Influence of parameter  $b$  on the stress-strain response for the Knowles model.

Marlow model. This type of model implementation is based on the assumption that the free energy is only dependent on the first invariant  $I_1$ , which is then determined from the experimental data. A consequence of this is that although this model can predict the uniaxial tension data with almost perfect accuracy, the predictions of other loading modes are typically far from perfect.

The accuracy of the response function model to predict the behavior of elastomers is demonstrated in Figure 5.27 by comparison to vulcanized natural rubber data of Treloar [16].

This figure shows that as expected the model predictions agrees with the provided uniaxial tension data, but the predicted stress in biaxial loading is too low. This is typical since accurately predicting the response in biaxial loading often requires an energy function that includes  $I_2$  dependence, which is not the case for response function hyperelasticity.



**Figure 5.27** Comparison between experimental data from Treloar [16] and predictions from the response function model as implemented by Abaqus.

### 5.3.16 Extended Tube Model

The Extended Tube (ET) model is a statistical mechanics and micromechanics inspired material model that considers the network constraints from surrounding molecular chains and the limited chain extensibility in the derivation of the Helmholtz free energy expression [39].

The energy function consists of three parts: energy from the cross-linking of the network ( $\Psi_c$ ), energy from confining tube constrains ( $\Psi_e$ ), and energy from volumetric deformations ( $\Psi_v$ ):  $\Psi = \Psi_c + \Psi_e + \Psi_v$ , where:

$$\Psi_c = \frac{G_c}{2} \left[ \frac{(1 - \delta^2)(I_1^* - 3)}{1 - \delta^2(I_1^* - 3)} + \ln \left( 1 - \delta^2(I_1^* - 3) \right) \right], \quad (5.137)$$

$$\Psi_e = \frac{2G_e}{\beta^2} \sum_{i=1}^3 \left( (\lambda_i^*)^{-\beta} - 1 \right), \quad (5.138)$$

$$\Psi_v = \kappa (J - 1)^2. \quad (5.139)$$

In this equation  $[G_e, G_c, \delta, \beta, \kappa]$  are material parameters,  $I_1^* = J^{-2/3} \text{tr}[\mathbf{C}]$ ,  $\lambda_i^* = J^{-1/3} \lambda_i$ , and  $J = \det[\mathbf{F}]$ . The energy term  $\Psi_c$  is a simple  $I_1^*$  based expression,  $\Psi_e$  is a one-term Ogden model with a negative exponent, and  $\Psi_v$  gives the volumetric response.

The Cauchy stress for the ET model can be calculated from Equations (5.42) and (5.55) giving:

$$\begin{aligned} \boldsymbol{\sigma} = & \frac{G_c}{J} \left\{ \frac{1 + (1 + (I_1^*)^2 - 4I_1^*) \delta^2 + (5I_1^* - (I_1^*)^2 - 6) \delta^4}{[1 - (I_1^* - 3) \delta^2]^2} \right\} \text{dev}[\mathbf{b}^*] \\ & - \frac{2G_e}{J\beta} \sum_{i=1}^3 \left[ (\lambda_i^*)^{-\beta} - \frac{1}{3} \left( (\lambda_1^*)^{-\beta} + (\lambda_2^*)^{-\beta} + (\lambda_3^*)^{-\beta} \right) \right] \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i + \kappa (J - 1) \mathbf{I}. \end{aligned} \quad (5.140)$$

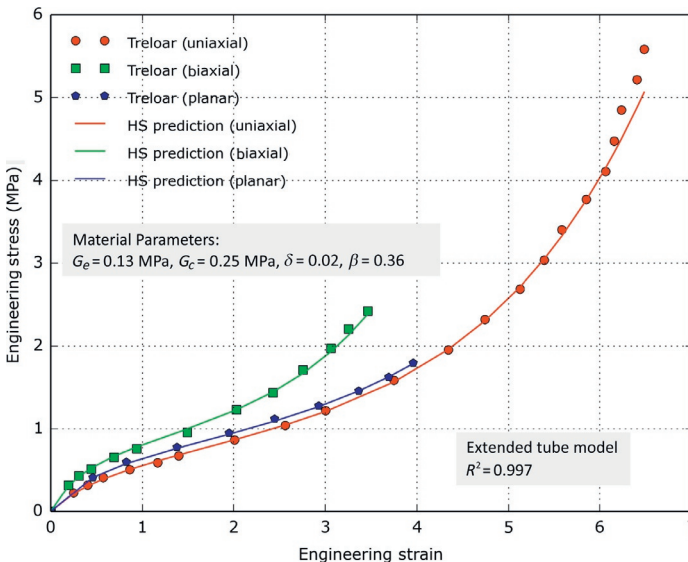
The Cauchy stresses for the incompressible ET model in uniaxial loading, planar loading, and biaxial loading are given by:

$$\sigma_{\text{uniax}} = G_c \left\{ \frac{1 + (1 + (I_1^*)^2 - 4I_1^*) \delta^2 + (5I_1^* - (I_1^*)^2 - 6) \delta^4}{[1 - (I_1^* - 3)\delta^2]^2} \right\} \cdot \left( \lambda^2 - \frac{1}{\lambda} \right) - \frac{2G_c}{\beta} \left[ \lambda^{-\beta} - \left( \frac{1}{\sqrt{\lambda}} \right)^{-\beta} \right], \quad (5.141)$$

$$\sigma_{\text{planar}} = G_c \left\{ \frac{1 + (1 + (I_1^*)^2 - 4I_1^*) \delta^2 + (5I_1^* - (I_1^*)^2 - 6) \delta^4}{[1 - (I_1^* - 3)\delta^2]^2} \right\} \cdot \left( \lambda^2 - \frac{1}{\lambda^2} \right) - \frac{2G_c}{\beta} \left[ \lambda^{-\beta} - \left( \frac{1}{\lambda} \right)^{-\beta} \right], \quad (5.142)$$

$$\sigma_{\text{biaxial}} = G_c \left\{ \frac{1 + (1 + (I_1^*)^2 - 4I_1^*) \delta^2 + (5I_1^* - (I_1^*)^2 - 6) \delta^4}{[1 - (I_1^* - 3)\delta^2]^2} \right\} \cdot \left( \lambda^2 - \frac{1}{\lambda^4} \right) - \frac{2G_c}{\beta} \left[ \lambda^{-\beta} - \left( \frac{1}{\lambda^2} \right)^{-\beta} \right]. \quad (5.143)$$

The accuracy of the ET model to predict the behavior of elastomers is demonstrated in [Figure 5.28](#) by comparison to vulcanized natural rubber data of Treloar [16]. The figure shows that ET model in this case is more accurate than any of the other hyperelastic models that have been examined so far.



**Figure 5.28** Comparison between experimental data from Treloar [16] and predictions from the Extended Tube model.

The following code example shows one way to implement the ET model for *compressible* uniaxial loading.

Additional Code to “Polymer\_Mechanics\_Chap05.py”:

```
def ETube_3D(stretch, param):
    """Extended Tube model. 3D loading specified by stretches.
    Param: Ge, Gc, delta, beta, kappa"""
    Ge = param[0]
    Gc = param[1]
    delta = param[2]
    beta = param[3]
    kappa = param[4]
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    lam = J**(-1/3) * array(stretch)
    I1s = trace(bstar)
    fac1 = (1+(1+I1s**2-4*I1s)*delta**2 + \
            (5*I1s-I1s**2-6)*delta**4) / \
            (1 - (I1s-3)*delta**2)**2
    stressC = Gc/J * fac1 * devbstar
    fac2 = -2*Ge/(J*beta)
    tmp = (lam[0]**(-beta) + lam[1]**(-beta) + lam[2]**(-beta)) / 3
    stressE = zeros((3,3))
    stressE[0,0] = fac2 * (lam[0]**(-beta) - tmp)
    stressE[1,1] = fac2 * (lam[1]**(-beta) - tmp)
    stressE[2,2] = fac2 * (lam[2]**(-beta) - tmp)
    stressV = kappa*(J-1)*eye(3)
    return stressC + stressE + stressV
```

Python Code: “ETube\_compressible\_uniaxial.py”

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

# [Ge, Gc, delta, beta, kappa]
param = [1.0, 1.0, 0.1, 0.1, 1000.0]

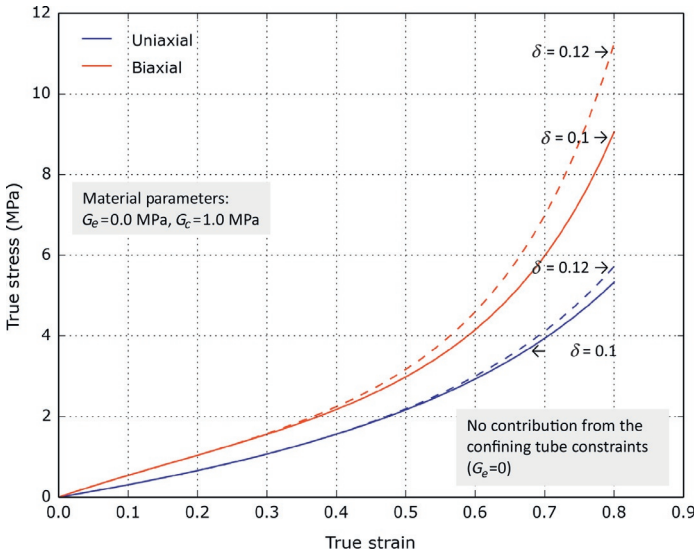
trueStrain = linspace(0, 0.7, 100)
trueStress = uniaxial_stress(ETube_3D, trueStrain, param)

plot(trueStrain, trueStress, 'b-')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

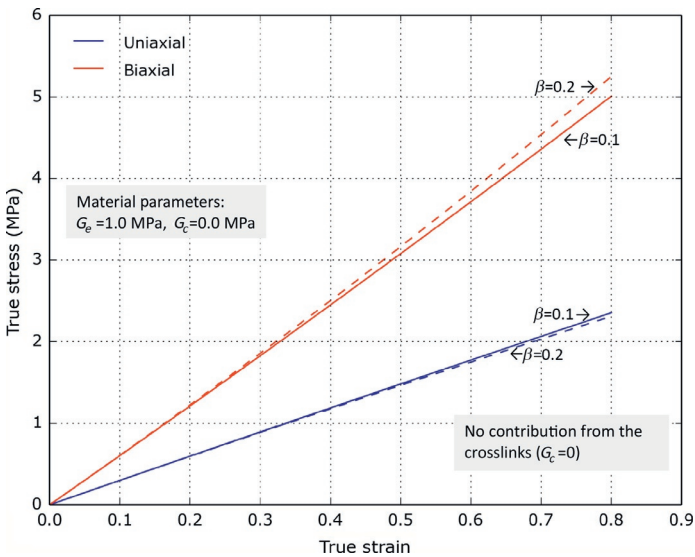
The influence of the parameters  $\delta$  and  $\beta$  on the predicted stress-strain response of the ET model are shown in [Figures 5.29](#) and [5.30](#). These figures show that the  $\delta$  parameter strongly influence the predicted stress-strain response.

### 5.3.17 BAM Model

The BAM model is a new hyperelastic material model that I developed specifically for this book. The purpose of this exercise is to show one way to improve the accuracy of the Arruda-Boyce EC model. The EC model, which was discussed in [Section 5.3.10](#), is an excellent model for predicting the multiaxial response of



**Figure 5.29** Influence of parameter  $\delta$  on the stress-strain response for the Extended Tube model.



**Figure 5.30** Influence of parameter  $\beta$  on the stress-strain response for the Extended Tube model.



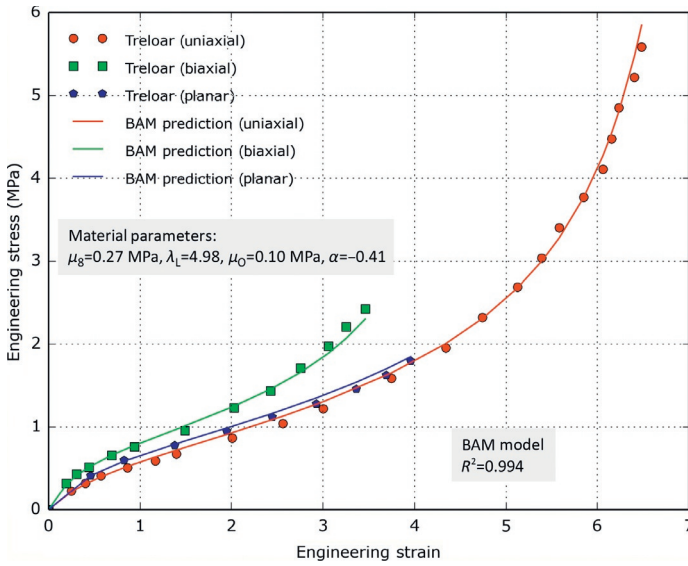
elastomers. One of the main limitations of the EC model is that it underpredicts the stress in biaxial loading since its Helmholtz free energy does not depend on the second invariant. One way to remedy that limitation is to add another contribution to the Helmholtz free energy that specifically helps in biaxial loading. One simple way to do that is to follow the approach developed by the extended tube model, where a first order Ogden model with a negative  $\alpha$  coefficient is used. As was shown in [Figure 5.30](#), a first order Ogden model with a negative  $\alpha$  parameter is exactly what is needed.

The new hyperelastic model can then be obtained by quickly combining the energy functions from the EC model (Equation (5.100)) and the first-order Ogden model (Equation (5.117)), and bam, a new material model is obtained. We will simply call this new model the BAM model.

The predictions from the BAM model when compared to the classical data from Treloar is shown in [Figure 5.31](#). The figure shows that the BAM model is as accurate as the 3-term Ogden model and the extended tube model, which were the most accurate hyperelastic models that have been examined so far for the Treloar data.

## 5.4 Summary of Predictive Capabilities of Isotropic Hyperelastic Models

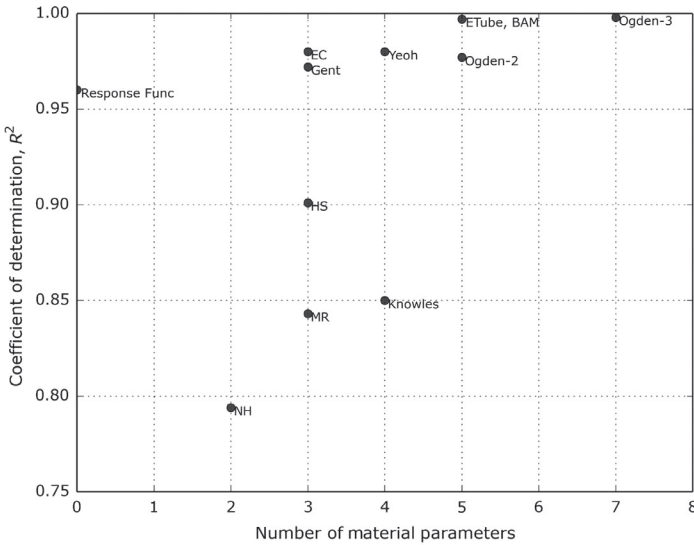
A summary of the predictive capabilities of the various hyperelastic models discussed in the previous sections are summarized in [Table 5.4](#). In this table the accuracy of the different material models is quantified by the coefficient of determination ( $R^2$ ). The table shows that the Ogden (3-term) model and the ET model are the most accurate. Another way to evaluate the usefulness of the different models is to compare the accuracy of the model predictions to the number of required material parameters, see [Figure 5.32](#). In this case one can argue that a material model with fewer material parameters should be easier to calibrate and therefore should be preferable to a material model with many material parameters. The figure shows that response function



**Figure 5.31** Comparison between experimental data from Treloar [16] and predictions from the BAM model.

**Table 5.4** Comparison Between the Predictive Capabilities of Different Isotropic Hyperelasticity Models

Hyperelastic Model	$R^2$ -Prediction
Neo-Hookean	0.794
Mooney-Rivlin	0.843
Yeoh	0.980
Eight-Chain	0.973
Ogden (2-term)	0.977
Ogden (3-term)	0.998
Gent	0.972
Horgan-Saccomandi	0.901
Knowles	0.850
Response Function	0.960
Extended Tube	0.997
BAM	0.994

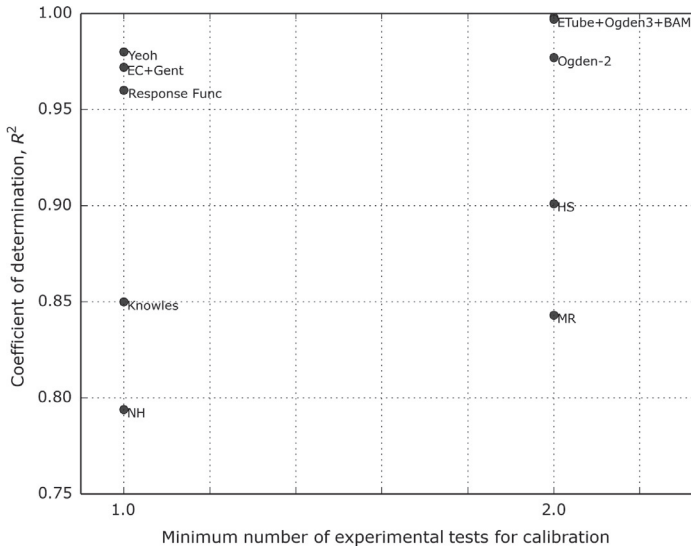


**Figure 5.32** Comparison between the different hyperelastic material model with respect to the accuracy and the number of material parameters.

model, the Arruda-Boyce EC model, the Yeoh model, and the ET model are the most promising from this condition.

Another way to evaluate the usefulness of the different candidate hyperelastic models is to compare the predictive accuracy to the minimum number of experimental tests that are required for the model calibration, see [Figure 5.33](#). In this case, material models that are only based on the first invariant  $I_1$  are assigned a minimum number of experimental tests of one (1), and material model that are based on both the first and second invariants ( $I_1$  and  $I_2$ ) or the principal stretches are assigned a minimum number of experimental tests of two (2). This figure suggests that the Yeoh model, the Arruda-Boyce EC model, the Gent model, and the response function models are most useful since they only require one experimental test.

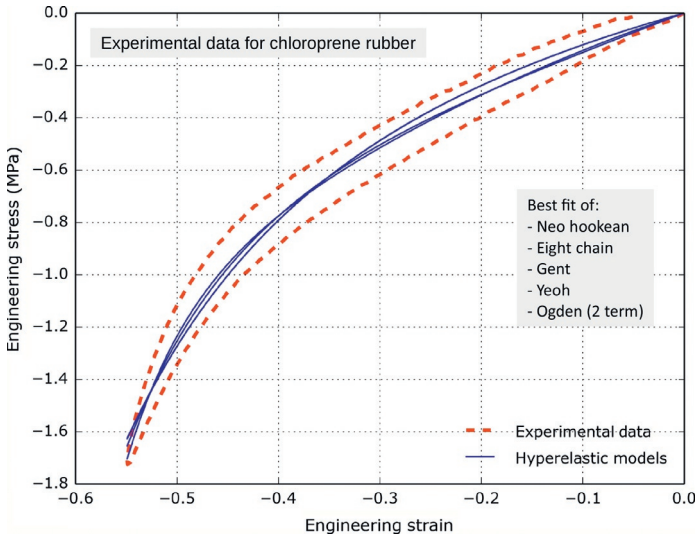
It is important to note that the results presented in this table were obtained for the classical Treloar [16] data for natural rubber in uniaxial tension, planar tension, and equibiaxial tension. It is possible that the relative ranking of the different models would be different if a different experimental data set was used in the study.



**Figure 5.33** Comparison between the different hyperelastic material model with respect to the accuracy and the number of required experimental tests.

The results presented here are included for illustrational purposes, and are not intended to provide a comprehensive review and evaluation of the various hyperelasticity models. Furthermore, as will be discussed in more detail in [Section 5.8.1](#), the mechanical behavior of elastomers in real-world applications is significantly more advanced than the simple data represented in Treloar's data set.

For example, the response of elastomers in real loading cases is characterized not only by a non-linear hyperelastic response but also non-linear viscoelasticity and damage (Mullins effect) during cyclic loading. These additional non-linearities of the material response will make the reported  $R^2$ -values in [Table 5.4](#) higher than the actual values in many real-world applications. Hence, if a hyperelastic model is chosen to represent the behavior of an elastomer-like material, then there will likely be inaccuracies in the model predictions due to the lack of viscoelastic effect that are more significant than the difference between some of the more advanced hyperelastic model discussed above.



**Figure 5.34** Comparison between experimental data for a chloroprene rubber and calibrated material models for the Neo-Hookean, eight-chain, Yeoh, Gent, and Ogden models.

This behavior is depicted in [Figure 5.34](#). This figure shows that all the different hyperelastic material models provide similar predictions, and that all are unable to predict the loading-unloading response of the material.

## 5.5 Anisotropic Hyperelasticity

Anisotropic hyperelasticity is a relatively new topic that enables many new exciting predictive capabilities, both for biopolymers and other intrinsically anisotropic polymers. Anisotropic hyperelasticity is also a building block for more advanced viscoplastic models as will be discussed in following chapters.

The theory for anisotropic hyperelasticity is covered nicely in the textbook by Holzapfel [13]. If the applied strains are small then it is often sufficient to use an anisotropic elastic model. For larger strains there are two commonly used approaches: strain based [40] and invariant based [41] hyperelasticity. This section summarizes the theory of some commonly used anisotropic hyperelastic models.

### 5.5.1 Generalized Fung Model

The energy function for the generalized Fung anisotropic hyperelastic model [40] can be written:

$$\Psi(c, b_{ijkl}, \kappa) = \frac{c}{2} [\exp(Q) - 1] + \frac{\kappa}{2} (J - 1)^2, \quad (5.144)$$

where  $c$ ,  $b_{ijkl}$ , and  $\kappa$  are material parameters. Note that the parameter  $b_{ijkl}$  is a dimensionless symmetric fourth order tensor of material parameters. The scalar variable  $Q$  is given by:

$$Q = \mathbf{E}^* : \mathbf{b} : \mathbf{E}^* = E_{ij}^* b_{ijkl} E_{kl}^*, \quad (5.145)$$

where  $\mathbf{E}^* = [\mathbf{C}^* - \mathbf{I}]/2$  is the Green-Lagrange strain defined in Equation (4.119),  $\mathbf{C}^* = J^{-2/3} \mathbf{F}^\top \mathbf{F}$  is the right Cauchy-Green tensor, and  $J = \det[\mathbf{F}]$ . For a case with cylindrical coordinates, this equation can be written:

$$\begin{aligned} Q = & b_1 E_{\theta\theta}^2 + b_2 E_{zz}^2 + b_3 E_{rr}^2 + 2b_4 E_{\theta\theta} E_{zz} + 2b_5 E_{zz} E_{rr} \\ & + 2b_6 E_{rr} E_{\theta\theta} + b_7 E_{\theta z}^2 + b_8 E_{rz}^2 + b_9 E_{r\theta}^2. \end{aligned} \quad (5.146)$$

The Cauchy stress for the Generalized Fung model is:

$$\boldsymbol{\sigma} = \frac{1}{J} \mathbf{F} \frac{\partial \Psi}{\partial \mathbf{E}^*} \mathbf{F}^\top + \frac{\partial \Psi}{\partial J} \mathbf{I}. \quad (5.147)$$

The Fung anisotropic hyperelastic model is commonly used to model both engineered and native soft tissues used in medical device and surgical applications [42].

### 5.5.2 Invariant Based Anisotropy

Another common approach to incorporate anisotropic effects into a hyperelastic model is to introduce two additional invariants:  $I_4, I_5$  defined by:

$$I_4 = (\mathbf{F}\mathbf{a}_0) \cdot (\mathbf{F}\mathbf{a}_0) = \mathbf{a}_0 \cdot \mathbf{C}\mathbf{a}_0 = \lambda_F^2, \quad (5.148)$$

$$I_5 = \mathbf{a}_0 \cdot \mathbf{C}^2 \mathbf{a}_0, \quad (5.149)$$

where  $I_4$  is the stretch in the provided fiber direction  $\mathbf{a}_0$ , and  $\mathbf{C}$  is the right Cauchy-Green tensor. The total Helmholtz free energy is then given by:

$$\Psi = \Psi_1(I_1^*, I_2^*, J) + \Psi_2(I_4, I_5). \quad (5.150)$$

The Cauchy stress for this type of invariant-based anisotropic hyperelasticity is given by [13]:

$$\begin{aligned} \boldsymbol{\sigma} = & \frac{2}{J} \left[ \frac{\partial \Psi}{\partial I_1^*} + \frac{\partial \Psi}{\partial I_2^*} I_1^* \right] \mathbf{b}^* - \frac{2}{J} \frac{\partial \Psi}{\partial I_2^*} (\mathbf{b}^*)^2 + \left[ \frac{\partial \Psi}{\partial J} - \frac{2I_1^*}{3J} \frac{\partial \Psi}{\partial I_1^*} - \frac{4I_2^*}{3J} \frac{\partial \Psi}{\partial I_2^*} \right] \mathbf{I} \\ & + \frac{2}{J} I_4 \frac{\partial \Psi}{\partial I_4} \mathbf{a} \otimes \mathbf{a} + \frac{2}{J} I_4 \frac{\partial \Psi}{\partial I_5} (\mathbf{a} \otimes \mathbf{b}\mathbf{a} + \mathbf{a}\mathbf{b} \otimes \mathbf{a}). \end{aligned} \quad (5.151)$$

In this equation  $\mathbf{a} = |\mathbf{F}\mathbf{a}_0|$  is the direction of fiber  $\mathbf{a}_0$  in the deformed state.

### Example: Simple Anisotropic Hyperelastic Model.

As an example, consider a NH material with a single family of fibers with the energy function:

$$\Psi_2(I_4) = A \left( I_4^2 - 2I_4 \right) = A \left( \lambda_f^4 - \lambda_f^2 \right). \quad (5.152)$$

For this simple material model the Cauchy stress is given by:

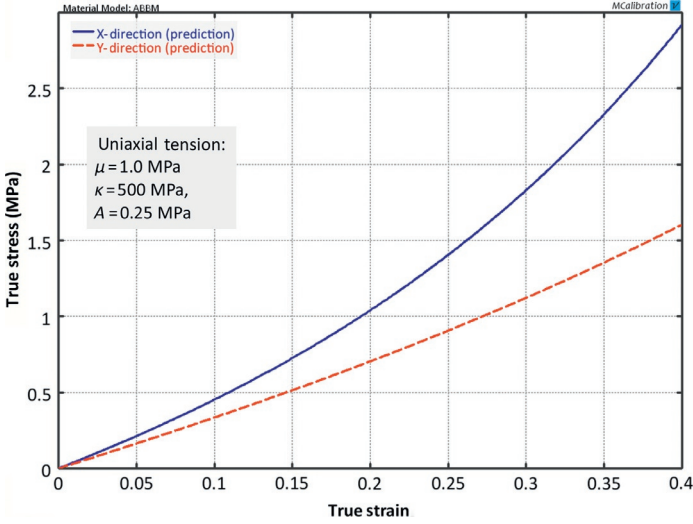
$$\boldsymbol{\sigma} = \frac{\mu}{J} \text{dev}[\mathbf{b}^*] + \kappa(J-1) + \frac{4A}{J} (\lambda_f^2 - 1) (\mathbf{F}\mathbf{a}_0) \otimes (\mathbf{F}\mathbf{a}_0), \quad (5.153)$$

which is plotted in [Figure 5.35](#). This figure shows that adding a single family of fibers can cause a significantly anisotropic response.

### 5.5.3 Bischoff Anisotropic Eight-Chain Model

The Arruda-Boyce EC model (see [Section 5.3.10](#)) is a useful micromechanism inspired model that has been shown to accurately capture the equilibrium response of many isotropic elastomers [43]. Bischoff developed an extension of this model that also works for anisotropic materials [44]. The basic idea of this model framework is that the eight chain unit cell is stretched also its initial unloaded configuration.

The stress predicted by the Bischoff anisotropic EC model is given by:



**Figure 5.35** Stress-strain prediction in uniaxial tension of a simple anisotropic Neo-Hookean material model.

$$\boldsymbol{\sigma} = \sum_{f=1}^4 \frac{3\mu}{4J\lambda_f} \frac{\mathcal{L}^{-1}(\lambda_f/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)} (\mathbf{F}\hat{\mathbf{a}}_f) \otimes (\mathbf{F}\hat{\mathbf{a}}_f) - \sum_{i=1}^3 3\mu \left( \frac{a_i}{\lambda_L} \right)^2 \mathbf{e}_i \otimes \mathbf{e}_i + \kappa (J-1) \mathbf{I}, \quad (5.154)$$

where  $\mu$  is the shear modulus,  $\kappa$  is the bulk modulus,  $[a_1, a_2, a_3]$  are the side lengths of the unit cell in the undeformed configuration,  $J = \det[\mathbf{F}]$ ,  $\mathcal{L}^{-1}(\cdot)$  is the inverse Langevin function, and where:

$$\lambda_L = \sqrt{a_1^2 + a_2^2 + a_3^2}, \quad (5.155)$$

$$\hat{\mathbf{a}}_1 = [+a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{e}_3]/\lambda_L, \quad (5.156)$$

$$\hat{\mathbf{a}}_2 = [+a_1\mathbf{e}_1 + a_2\mathbf{e}_2 - a_3\mathbf{e}_3]/\lambda_L, \quad (5.157)$$

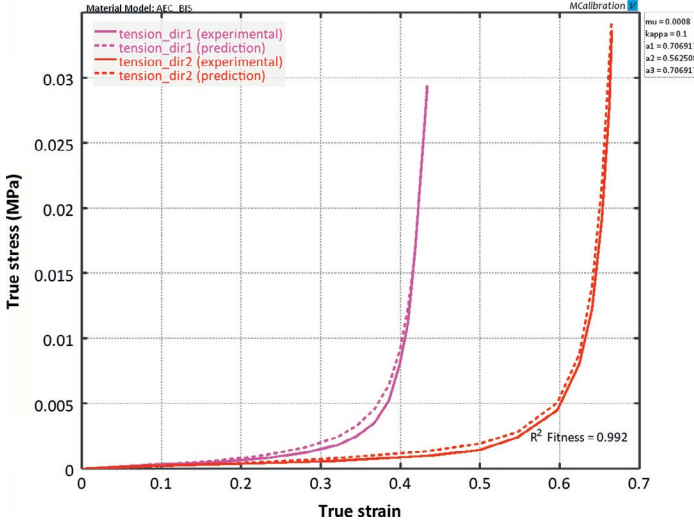
$$\hat{\mathbf{a}}_3 = [+a_1\mathbf{e}_1 - a_2\mathbf{e}_2 + a_3\mathbf{e}_3]/\lambda_L, \quad (5.158)$$

$$\hat{\mathbf{a}}_4 = [+a_1\mathbf{e}_1 - a_2\mathbf{e}_2 - a_3\mathbf{e}_3]/\lambda_L, \quad (5.159)$$

$$\lambda_f = \|\mathbf{F}\hat{\mathbf{a}}_f\| = (\hat{\mathbf{a}}_f \cdot \mathbf{C}\hat{\mathbf{a}}_f)^{1/2}. \quad (5.160)$$

One example of the Bischoff anisotropic EC model is given in [Figure 5.36](#). This figure shows experimental data for rabbit skin [45] and the stress-strain response from the Bischoff model.





**Figure 5.36** Comparison between experimental data for rabbit skin and predictions from the Bischoff model.

### 5.5.4 Bergstrom Anisotropic Eight-Chain Model

Another way to create an anisotropic hyperelastic model is to start with the isotropic Arruda-Boyce EC model and then add one or more families of fibers. Using this approach the Cauchy stress can be written [31]:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_{8chain}(\mathbf{F}) + \left[ A\lambda_f^2 + B\lambda_f - (A + B) \right] \mathbf{a}_f \otimes \mathbf{a}_f, \quad (5.161)$$

where  $\boldsymbol{\sigma}(\mathbf{F})$ ,  $\mathbf{a}_f = \mathbf{F}\mathbf{a}_0$ ,  $\lambda_f = \|\mathbf{a}_f\|$ . One example of the stress-strain response from this model is shown in Figure 5.35.

### 5.5.5 Holzapfel-Gasser-Ogden Model

The Holzapfel-Gasser-Ogden (HGO) model [46, 47] is an anisotropic hyperelastic material model that was developed to predict the response of arterial tissue. The HGO model is a versatile model that is a built-in feature in Abaqus.

The model consists of a NH model representation of the matrix response, and up to three different families of fibers. The initial fiber directions are given by the three vectors  $[a_{1x}, a_{1y}, a_{1z}]$ ,

$[a_{2x}, a_{2y}, a_{2z}]$ , and  $[a_{3x}, a_{3y}, a_{3z}]$ . The three fiber directions do not have to be orthogonal.

The strain energy function for the model is given by the following function:

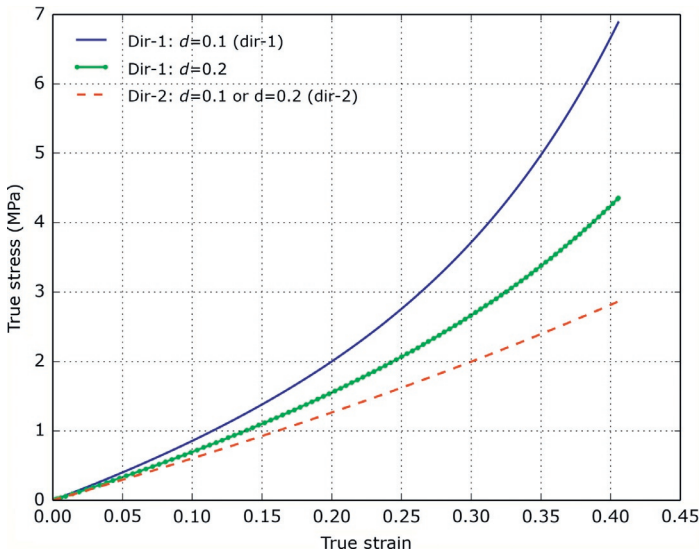
$$\Psi = \frac{\mu}{2}(I_1^* - 3) + \frac{k_1}{2k_2} \sum_{i=1}^3 \left[ e^{k_2 \langle E_i \rangle^2} - 1 \right] + \frac{\kappa}{2}(J - 1)^2, \quad (5.162)$$

where:

- the energy term  $E_i$  is:  $E_i = d(I_1^* - 3) + (1 - 3d)[I_{4i}^* - 1]$ ,
- $d$  is the dispersion,
- $I_{4i}^* = (\mathbf{F}^* \mathbf{a}_i) \cdot (\mathbf{F}^* \mathbf{a}_i)$ ,
- $\langle x \rangle = (x + |x|)/2$  is the ramp function,

If  $d = 0$  then the fibers are perfectly aligned, and if  $d = 1/3$  the fibers are randomly oriented giving an isotropic response.

One example of the stress-strain predictions from the HGO model is presented in Figure 5.37. The prediction is based on the following parameters:  $\mu = 2$  MPa,  $\kappa = 20$  MPa,  $k_1 = 1$  MPa,  $k_2 = 1$ ,  $d = 0.1$  or  $0.2$ , and one family of fibers oriented at  $[1,0,0]$ .



**Figure 5.37** Predicted stress-strain response from the Holzapfel-Gasser-Ogden model.

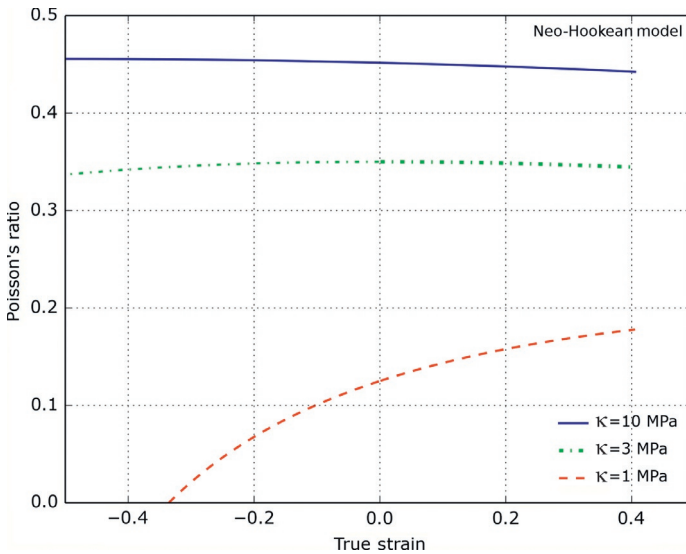
The figure shows that the stress in the 2-direction is independent of the value of  $d$ . This, of course, is caused by the lack of fibers in the 2-direction. The stress in the 1-direction, however, is dependent on the value of  $d$ .

## 5.6 Hyperelastic Foam Models

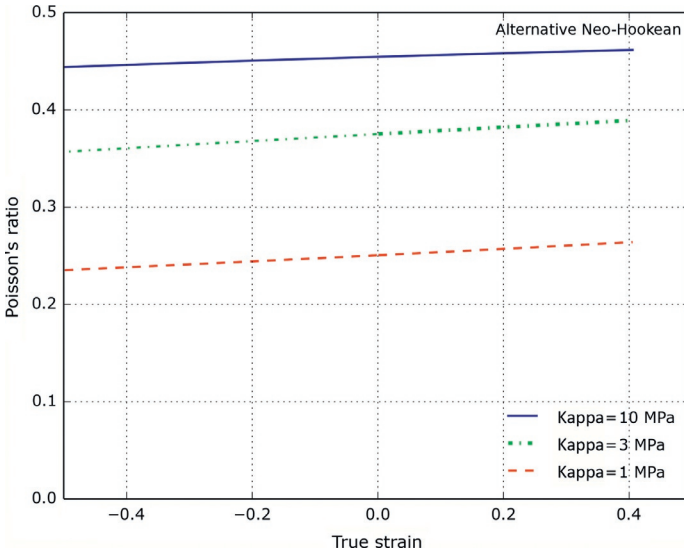
The mechanical behavior of compressible polymers, such as foams, can also be approximated using a hyperelastic representation. Since foams can undergo large volume change it is often beneficial to use slightly different forms of the Helmholtz free energy and to use a different set of experimental data when calibrating the hyperelastic model. One of the problems with the common hyperelastic models is illustrated in [Figures 5.38](#) and [5.39](#).

These figures show the uniaxial tension and compression response of the NH model with a stress function:

$$\boldsymbol{\sigma} = \frac{\mu}{J} \text{dev} [\mathbf{b}^*] + \kappa(J - 1)\mathbf{I}, \quad (5.74\text{-repeat})$$



**Figure 5.38** Predicted Poisson's ratio for the (standard) Neo-Hookean model with a shear modulus of 1 MPa.



**Figure 5.39** Predicted Poisson's ratio for the alternative Neo-Hookean model where the energy function is not divided into distortional and volumetric parts. The shear modulus is 1 MPa.

and the response of the alternative NH model:

$$\boldsymbol{\sigma} = \frac{\mu}{J} (\mathbf{b} - \mathbf{I}) + \kappa(J - 1)\mathbf{I}. \quad (5.85\text{-repeat})$$

As shown in the figures, the predicted Poisson's ratio from the standard NH material model becomes odd in compressive loading when the bulk modulus is similar in magnitude to the shear modulus. This is a problem when modeling foams since foams often have a small bulk modulus and a corresponding small (but positive) Poisson's ratio.

The alternative NH model avoids this problem by using the total deformation gradient (not only the deviatoric part) in combination with the volumetric response.

Different hyperelastic models have been proposed in order to better predict the non-linear elastic response of foam materials. The following subsections summarize some of the more commonly used the hyperelastic models for foams: the Blatz-Ko model, and the hyperfoam model.

### 5.6.1 Blatz-Ko Foam Model

A commonly used material model for polymer foams is the Blatz-Ko hyperelastic model [48]. This model was developed for porous materials with a known shear modulus, Poisson's ratio, and volume fraction of voids, and is called the *generalized Blatz-Ko* model.

A more commonly used version of the model is obtained by setting the void volume fraction parameter  $f = 0$ , and by setting the Poisson's ratio  $\nu = 0.25$ , giving a model that is typically just called the *Blatz-Ko* model [48, 49], and that has the following strain energy density:

$$\Psi = \frac{\mu}{2} \left( \frac{I_2}{I_3} + 2I_3^{1/2} - 5 \right). \quad (5.163)$$

As shown, this material model only has one material parameter: the shear modulus  $\mu$ .

The Cauchy stress for the Blatz-Ko model is given by:

$$\boldsymbol{\sigma} = \frac{\mu}{J^3} \left[ I_1 \mathbf{b} - \mathbf{b}^2 - (I_2 - J^3) \mathbf{I} \right]. \quad (5.164)$$

The following code example shows one way to implement the Blatz-Ko model for compressible uniaxial loading.

Additional Code to "Polymer\_Mechanics\_Chap05.py":

```
def blatzko_3D(stretch, param):
    """Blatz-Ko. 3D loading specified by stretches.
    param[0]=mu"""
    F = array([[stretch[0],0,0], [0,stretch[1],0], [0,0,stretch[2]]])
    J = det(F)
    b = dot(F, F.T)
    b2 = dot(b, b)
    I1 = trace(b)
    I2 = 0.5 * (I1**2 - trace(b2))
    return param[0]/J**3.0 * (I1*b - b2 - (I2-J**3.0) * eye(3))
```

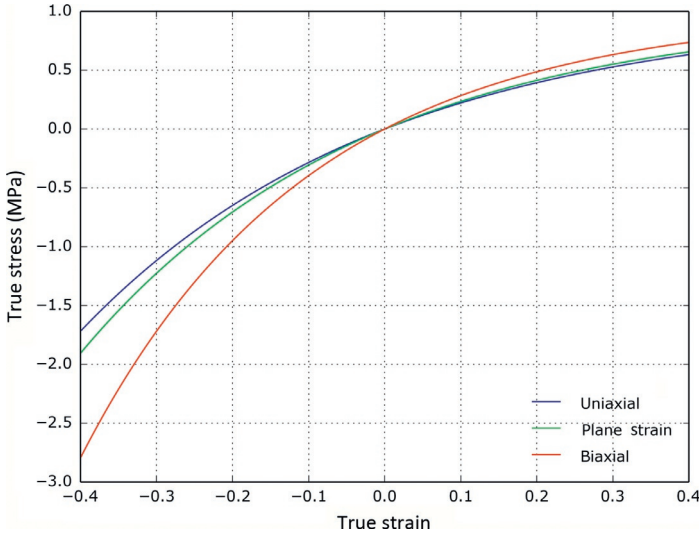
Python Code: "BlatzKo\_compressible\_uniaxial.py"

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

param = [1.0] # mu
trueStrain = linspace(0, 0.4, 100)

trueStress = uniaxial_stress(blatzko_3D, trueStrain, param)
plot(trueStrain, trueStress, 'b-', label='Uniaxial')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

The predicted stress-strain response in uniaxial, biaxial, and plane strain loading is shown in [Figure 5.40](#).



**Figure 5.40** Predicted stress-strain response for the Blatz-Ko model in different loading modes.

## 5.6.2 Hyperfoam Model

One interesting hyperelastic model for polymer foam materials is the hyperfoam model [27, 50]. In this model the Helmholtz free energy is given the following expression:

$$\Psi = \sum_{k=1}^N \frac{2\mu_k}{\alpha_k^2} \left[ (\lambda_1)^{\alpha_k} + (\lambda_2)^{\alpha_k} + (\lambda_3)^{\alpha_k} - 3 + \frac{1}{\beta_k} (J^{-\alpha_k \beta_k} - 1) \right], \quad (5.165)$$

where  $[\mu_i, \alpha_i, \beta_i]$ ,  $i = 1 \dots k$  are material parameters,  $J = \det[\mathbf{F}]$ , and  $\lambda_i$  are the principal stretches. As shown in this equation, the hyperfoam model is very similar to the Ogden model, except that it has a slightly different energy function for the volumetric response. The volumetric response was here selected in order to get a constant Poisson's ratio in uniaxial loading.

The Cauchy stress can be derived from Equation (5.52) giving:

$$\boldsymbol{\sigma} = \sum_{k=1}^N \frac{2\mu_k}{J\alpha_k} \left\{ \sum_{i=1}^3 [(\lambda_i)^{\alpha_k} - J^{-\alpha_k \beta_k}] \hat{\mathbf{n}}_i \otimes \hat{\mathbf{n}}_i \right\}. \quad (5.166)$$

The following code example shows one way to implement the hyperfoam model for uniaxial loading.

Additional Code to “Polymer\_Mechanics\_Chap05.py”:

```
def hyperfoam_3D(stretch, param):
    """Hyperfoam model. 3D loading specified by stretches.
    param: [mu1, ..., alpha1, ..., beta1, ...].
    Returns true stress."""
    J = stretch[0] * stretch[1] * stretch[2]
    lam = array(stretch)
    N = int(round(len(param)/3.0))
    mu = param[0:N]
    alpha = param[N:2*N]
    beta = param[2*N:3*N]
    Stress = zeros((3,3))
    for k in range(N):
        fac = 2.0 * mu[k] / (J * alpha[k])
        Stress[0,0] = Stress[0,0] + fac*(lam[0]**alpha[k] - \
            J**(-alpha[k]*beta[k]))
        Stress[1,1] = Stress[1,1] + fac*(lam[1]**alpha[k] - \
            J**(-alpha[k]*beta[k]))
        Stress[2,2] = Stress[2,2] + fac*(lam[2]**alpha[k] - \
            J**(-alpha[k]*beta[k]))
    return Stress
```

Python Code: “hyperfoam\_compressible\_uniaxial.py”

```
from pylab import *
from Polymer_Mechanics_Chap05 import *

# k=1: mu1, alpha1, beta1
param = [1.0, 2.0, 0.2]
trueStrain = linspace(0, 0.4, 40)

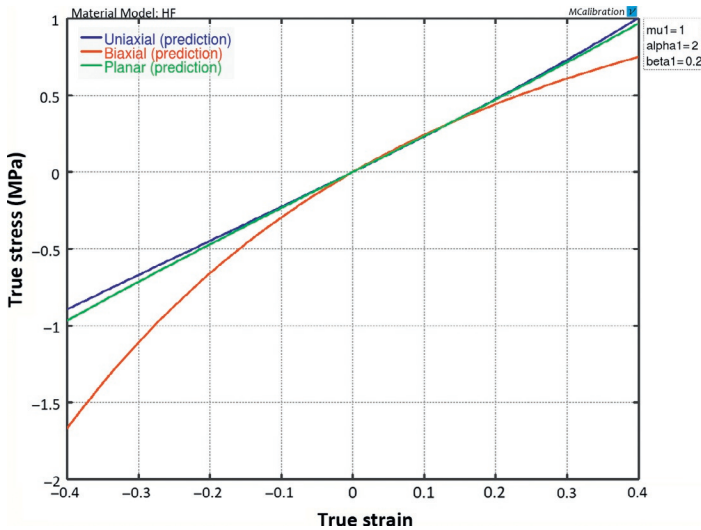
trueStress = uniaxial_stress(hyperfoam_3D, trueStrain, param)
plot(trueStrain, trueStress, 'b-', label='uniax')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

Exemplar stress-strain predictions for the hyperfoam model are shown in [Figure 5.41](#). This figure shows the model predictions when  $\mu = 1$  MPa,  $\alpha = 2$ , and  $\beta = 0.2$ . The model predictions of the compressive response are in qualitative agreement with many polymer foams, but the tensile predictions are not and may require additional terms in the energy function for proper predictions of an actual material.

It can be shown [27], that the Poisson’s ratio for the hyperfoam model is equal to

$$\nu = \frac{\beta}{1 + 2\beta}, \quad (5.167)$$

if all  $\beta_i$  terms are equal. This is a convenient feature of the hyperfoam model that allows all  $\beta$  parameters to be determined from the average Poisson’s ratio that can be experimentally measured. The  $\mu_i$  parameters specify the stiffness of the model response, and the  $\alpha_i$  parameters specify the shape of the predicted stress-strain response at larger strains.



**Figure 5.41** Predicted stress-strain response for the hyperfoam model with  $\mu = 1$  MPa,  $\alpha = 2$ ,  $\beta = 0.2$ .

## 5.7 Mullins Effect Models

The mechanical stress response of elastomers often undergoes significant softening during the first few load cycles, and after those load cycles the material response becomes repeatable [1, 51–54]. This material softening can be considered a damage accumulation in the material. One example showing experimental data with the Mullins effect is shown in Figure 2.3.

There is still some uncertainty about the details of the mechanisms causing the Mullins effect [54], but the following experimental observations are typically seen in all rubbers that exhibit Mullins damage:

- The extent of Mullins softening increases with filler particle concentration.
- The stress at a given strain is higher in the first load cycle, than in any of the following load cycles.
- The amount of damage that is introduced in the material is the highest in the first load cycle, and then goes down with subsequent load cycles. After about



3 to 5 load cycles (to a given strain level) the material response becomes repeatable.

- The amount of damage is dependent on the max applied strain. If a material is conditioned at, say, 30% strain, then the material will start to exhibit more damage accumulation again if the applied is increased to a larger strain level at a later time.
- The Mullins damage is not permanent, the material often slowly recovers with time. The rate of recovery depends on the temperature.

The most common physical interpretation of the Mullins effect is that it is caused by molecular chain breakage at interface of filler particles, or breakdown of filler particle clusters. Note that the Mullins effect only occurs in elastomers, and elastomer-like materials. Thermoplastic materials can also soften due to damage accumulation, but that behavior is not called the Mullins effect.

The following sections summarize two models for predicting the Mullins effect in a FE simulation.

### 5.7.1 Ogden-Roxburgh Mullins Effect Model

In an attempt to account for the Mullins effect, Ogden and Roxburgh [2] proposed a general extension of hyperelasticity in which the Helmholtz free energy per unit reference volume,  $\Psi$ , is taken to be a function not only of the applied deformation state but also on an internal state variable  $\eta$  that tracks the damage evolution in the material. This model is available as a built-in feature of most FE programs. For incompressible loading, the following form is used:

$$\Psi(\lambda_1, \lambda_2, \lambda_3, \eta) = \eta\Psi(\lambda_1, \lambda_2, \lambda_3), \quad (5.168)$$

where  $\lambda_i$  are the principal stretches. In the Ogden-Roxburgh (OR) model the damage variable  $\eta$  is taken to evolve with the applied deviatoric strain energy as follows<sup>12</sup>:

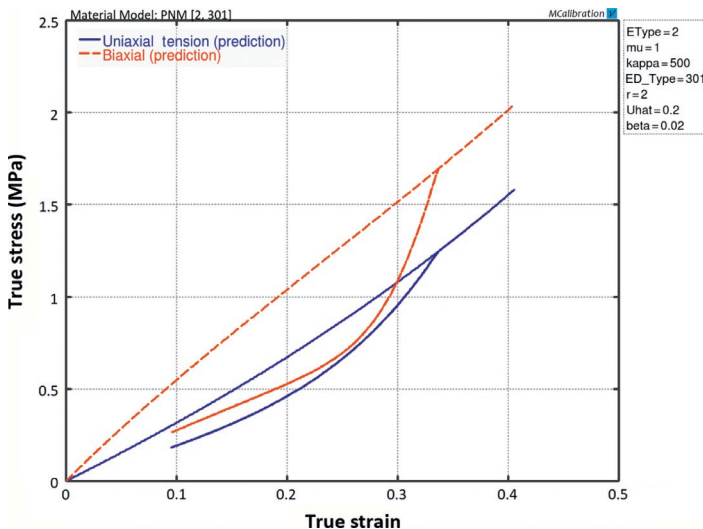
---

<sup>12</sup>This is an extension of the OR model that is used by the FE program Abaqus. The original OR model had  $\beta = 0$ .

$$\eta = 1 - \frac{1}{r} \operatorname{erf} \left( \frac{\Psi_{\text{dev}}^{\max} - \Psi_{\text{dev}}}{m + \beta \Psi_{\text{dev}}^{\max}} \right), \quad (5.169)$$

where  $r$ ,  $\beta$ , and  $m$  are material parameters,  $\operatorname{erf}(x)$  is the error function,  $\Psi_{\text{dev}}$  is the current deviatoric strain energy density, and  $\Psi_{\text{dev}}^{\max}$  is a state variable for the evolving maximum deviatoric strain energy density for the material point in its deformation history. This simple damage model can be applied to any isotropic hyperelasticity model.

An example of the behavior of the OR model is shown in [Figure 5.42](#). This figure plots stress-strain predictions for a uniaxial tension, and a biaxial load case. In both cases the material was stretched to 30% engineering strain, then unloaded to 10% engineering strain, and then finally loaded to 50% engineering strain. The material model was a NH hyperelastic model with Ogden-Roxburgh Mullins damage parameters:  $\mu = 1$  MPa,  $r = 2$ ,  $m = 0.2$ , and  $\beta = 0.02$ . The figure shows that the predicted stress-strain behavior is the same during unloading and during the following loading. The model also predicts that all Mullins



**Figure 5.42** Predicted stress-strain response for a Neo-Hookean hyperelastic model with Mullins damage.

damage occurs in the first load cycle, which is not what is typically not experimentally seen.

### 5.7.2 Qi-Boyce Mullins Effect Model

A different approach to account for the Mullins effect has been developed by Qi and Boyce [3]. In their model the virgin material is considered to consist of two phases: soft domains and hard domains. The Qi-Boyce (QB) model introduces a state variable to track the transformation of hard domains to soft domains with the applied deformation history. For incompressible loading, the following form is used:

$$\Psi = (1 - v_f) \cdot \Psi(I_1). \quad (5.170)$$

The volume fraction of hard domains,  $v_f$ , evolves with the applied chain stretch as follows:

$$v_f = v_{ff} - v_{fi} \exp \left[ \frac{-A \cdot (\Lambda_{\text{chain}}^{\text{max}} - 1)}{\lambda_{\text{lock}} - \Lambda_{\text{chain}}^{\text{max}}} \right], \quad (5.171)$$

where  $v_{ff}$  is the final volume fraction of hard domains,  $v_{fi}$  is the initial volume fraction of hard domains, and  $\Lambda_{\text{chain}}^{\text{max}}$  is the maximum chain stretch the during the deformation history of the material point.

The QB Mullins effect model takes four material parameters:  $v_{ff}$ ,  $v_{fi}$ ,  $A$ ,  $\lambda_{\text{lock}}$ . The QB-model can be applied to any hyperelastic model, but is currently not available in most FE programs.

## 5.8 Use of Hyperelasticity in Polymer Modeling

Hyperelasticity is commonly used to predict the behavior of different elastomer-like materials, primarily due to its ease of use and its availability in virtually all FE programs. Hyperelasticity is also a building block of linear viscoelasticity, and more advanced viscoplasticity models.

Not all hyperelastic models that are presented in this chapter are directly available in all major commercial FE codes. Due

to the simplicity of the hyperelasticity theory, however, it is typically straightforward to implement and use some of the more modern and accurate hyperelastic models if desired. Most FE codes have the capability to augment the built-in material models with user-defined material models, typically coded as external Fortran subroutines that are linked into the FE software during simulations. The use of external user subroutines can provide significant advantages in many circumstances. This topic is discussed in more detail in Chapter 10.

### 5.8.1 Experimental Testing

There are no strict requirements for what experimental tests should be performed in order to accurately calibrate a hyperelasticity model. The minimum set of recommended types of experimental tests is dependent on the hyperelastic model, and performing more experimental testing is always better than running too few experiments.

There is also some confusion in the general literature about the recommended types of experimental tests. Specifically, it is often said that two or preferably three different loading modes are necessary for accurate calibration of a hyperelastic model. It is never bad to perform experiments in three different loading modes (e.g. uniaxial, biaxial, shear), but for many hyperelastic models it is sufficient to use uniaxial loading only.

A general guideline is that hyperelastic models with no dependence on  $I_2$  typically require less experimental tests. The reason for this is that the free energy function in this case becomes a scalar function of only one variable ( $I_1$ ), and that functional dependence can be fully obtained from one experiment. Examples of models that fall into this category are: the NH model, the Gent model, the Arruda-Boyce EC model, and the Yeoh model.

Models with a free energy equation that depend on the principal stretches (such as the Ogden model), and models with  $I_2$  dependence require testing in two or more loading modes such as uniaxial tension, planar tension, simple shear, or biaxial tension.

Note that to verify the accuracy of a hyperelastic model it is sometimes recommended to perform experimental tests in

multiple loading modes. Also note that it is very important to run the experiments to larger strains than what is expected in the real application in order to get a hyperelastic model that is reliable for all strain levels of interest.

### 5.8.2 Drucker Stability

One of the few complications of hyperelasticity models is that some calibrated models are not always stable, i.e. increasing the strain can lead to a reduction in stress (see, for example, [Figure 5.16](#)). One way to examine if a model with a given set of material parameters is stable is to check its Drucker's stability defined by:

$$\Delta(J\boldsymbol{\sigma}) : \Delta(\mathbf{E}_{\ln}) \geq 0, \quad (5.172)$$

where  $J = \det[\mathbf{F}]$ ,  $\boldsymbol{\sigma}$  is the Cauchy stress,  $\mathbf{E}_{\ln}$  is the logarithmic strain.

The challenge is that some hyperelastic models can be Drucker's stable to large strains in tension, but unstable at small strains in a different loading mode such as shear. To ensure that a model is Drucker's stable for all strains of interest it is necessary to examine all loading model or to use a model with known stability behavior.

One common practical approach to examine the Drucker's stability of a hyperelastic material is to examine the stability of the model response in a set of common loading modes, such as uniaxial, biaxial, and simple shear, and apply pre-defined large strain histories. If the model passes these tests then the model may be considered relatively safe to use. This numerical approach to check the Drucker's stability is used by many of the major FE codes.

Also note that not all polymers are Drucker's stable. For example, many thermoplastics soften after yield causing the stress to go down as the strain increases. This is not a problem, it simply means that the Drucker's stability check for those materials is not relevant.

### 5.8.3 Determination of Material Parameters

To specify the complete hyperelastic material model it is necessary to first select a constitutive model, and then find the material parameters by *calibrating* the material parameters to experimental data. Since one or more experimental stress-strain curves are used for the calibration, the mathematical procedure of determining the material parameters involves solving an over constrained set of equations. Hence, it is typically not possible to get a perfect fit of the model to the experimental data. Furthermore, all experimental testing include experimental errors, so the material calibration in practice is limited to finding a set of material parameters that gives a good as possible prediction of the experimental data.

The material calibration, even for a simple material model such as linear elasticity, include selecting a fitness function that express how close a model prediction is to the experimental data, and then using an optimization algorithm to calibrate the material model. Practical aspects of material model calibration are discussed in Chapter 9.

### 5.8.4 Limitations of Hyperelasticity

Hyperelastic models can be quite useful for predicting the behavior of rubber-like polymers. The main strengths of hyperelastic models include:

- Easy to use and calibrate.
- Accessible in major commercial FE codes.
- Computationally efficient.

The main limitations of hyperelasticity models include the following:

- Mainly applicable to rubber-like materials.<sup>13</sup>
- Only accurate for monotonic loading.

---

<sup>13</sup>*Hyperelasticity can also be used to predict the small strain behavior of amorphous behavior of amorphous and semicrystalline glassy polymers. For these materials, however, the hyperelasticity models do not provide any significant advantages over linear elasticity.*

- Do not capture rate-effects or viscoelasticity.
- Do not capture hysteresis during cyclic loading.

A simple way to extend the hyperelastic models presented in this chapter is to augment the hyperelastic model with linear viscoelasticity as presented in the next chapter.

## 5.9 Hyperelastic Code Examples

The behavior of a number of the hyperelastic models presented in this chapter were examined using short Python functions. In each example, it was listed “Additional Code to Polymer\_Mechanics\_Chap05.py.” This section summarizes the file `Polymer_Mechanics_Chap05.py`. This file can also be downloaded from this web address: [http://PolymerMechanics.com/Polymer\\_Mechanics\\_Chap05.zip](http://PolymerMechanics.com/Polymer_Mechanics_Chap05.zip).

```

from pylab import *
import scipy.optimize

def uniaxial_stress(model, trueStrainVec, params):
    """Compressible uniaxial loading. Returns true stress."""
    stress = zeros(len(trueStrainVec))
    for i in range(len(trueStrainVec)):
        lam1 = exp(trueStrainVec[i])
        calcS22Abs = lambda x: abs(model([lam1,x,x],params)[1,1])
        # search for transverse stretch that gives S22=0
        lam2 = scipy.optimize.fmin(calcS22Abs, x0=1/sqrt(lam1),
                                   xtol=1e-9, ftol=1e-9, disp=False)
        stress[i] = model([lam1,lam2,lam2], params)[0,0]
    return stress

def biaxial_stress(model, trueStrainVec, params):
    """Compressible biaxial loading. Returns true stress."""
    stress = zeros(len(trueStrainVec))
    for i in range(len(trueStrainVec)):
        lam1 = exp(trueStrainVec[i])
        calcS33Abs = lambda x: abs(model([lam1,lam1,x],params)[2,2])
        # search for transverse stretch that gives S33=0
        lam3 = scipy.optimize.fmin(calcS33Abs, x0=1/sqrt(lam1),
                                   xtol=1e-9, ftol=1e-9, disp=False)
        stress[i] = model([lam1,lam1,lam3], params)[0,0]
    return stress

def planar_stress(model, trueStrainVec, params):
    """Compressible planar loading. Returns true stress."""
    stress = zeros(len(trueStrainVec))
    for i in range(len(trueStrainVec)):
        lam1 = exp(trueStrainVec[i])
        calcS33Abs = lambda x: abs(model([lam1,1.0,x],params)[2,2])
        # search for transverse stretch that gives S33=0

```

```

    lam3 = scipy.optimize.fmin(calcS33Abs, x0=1/sqrt(lam1),
                               xtol=1e-9, ftol=1e-9, disp=False)
    stress[i] = model([lam1,1.0,lam3], params)[0,0]
return stress

def NH_3D(stretch, param):
    """Neo-Hookean. 3D loading specified by stretches.
    param[0]=mu, param[1]=kappa"""
    F = array([[stretch[0],0,0], [0,stretch[1],0], [0,0,stretch[2]]])
    J = det(F)
    Fstar = J**(-1/3) * F
    bstar = dot(Fstar, Fstar.T)
    dev_bstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/J * dev_bstar + param[1]*(J-1) * eye(3)

def MR_3D(stretch, param):
    """Mooney-Rivlin. 3D loading specified by stretches.
    param: [C10, C01, kappa]"""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    bstar2 = dot(bstar, bstar)
    I1s = trace(bstar)
    I2s = 0.5 * (I1s**2 - trace(bstar2))
    C10 = param[0]
    C01 = param[1]
    kappa = param[2]
    return 2/J*(C10+C01*I1s)*bstar - 2*C01/J*bstar2 + \
        (kappa*(J-1) - 2*I1s*C10/(3*J) - 4*I2s*C01/(3*J))*eye(3)

def Yeoh_3D(stretch, param):
    """Yeoh. 3D loading specified by stretches.
    param: [C10, C20, C30, kappa]. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    I1s = trace(bstar)
    return 2/J*(param[0] + 2*param[1]*(I1s-3) + 3*param[2]*(I1s-3)**2)
        *devbstar \ + param[3]*(J-1) * eye(3)

def invLangevin(x):
    EPS = spacing(1)
    if type(x) == float or type(x) == float64: # x is a scalar
        if x >= 1-EPS: x = 1 - EPS
        if x <= -1+EPS: x = -1 + EPS
        if abs(x) < 0.839:
            return 1.31435 * tan(1.59*x) + 0.911249*x
        return 1.0 / (sign(x) - x)
    # x is an array
    x[x >= 1-EPS] = 1 - EPS
    x[x <= -1+EPS] = -1 + EPS
    res = zeros(size(x))
    index = abs(x) < 0.839
    res[index] = 1.31435 * tan(1.59*x[index]) + 0.911249*x[index]
    index = abs(x) >= 0.839
    res[index] = 1.0 / (sign(x[index]) - x[index])
    return res

```



```

def EC_3D(stretch, param):
    """Eight-Chain. 3D loading specified by stretches.
    param: [mu, lambdaL, kappa]. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    lamChain = sqrt(trace(bstar)/3)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/(J*lamChain) * invLangevin(lamChain/param[1]) / \
        invLangevin(1/param[1]) * devbstar + param[2]*(J-1) * eye(3)

def Gent_3D(stretch, param):
    """Gent. 3D loading specified by stretches.
    param: [mu, Jm, kappa]. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    I1s = trace(bstar)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/J / (1 - (I1s-3)/param[1]) * devbstar + \
        param[2]*(J-1) * eye(3)

def HS_3D(stretch, param):
    """Horgan-Saccomandi. 3D loading specified by stretches.
    param: mu, lamMax, kappa. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    bstar2 = dot(bstar, bstar)
    I1s = trace(bstar)
    I2s = 0.5 * (I1s**2 - trace(bstar2))
    mu = param[0]
    lamM = param[1]
    kappa = param[2]
    fac = mu * lamM**4 / J
    den = lamM**6 - lamM**4 * I1s + lamM**2 * I2s - 1
    return fac/den * ((lamM**2 - I1s)*bstar + bstar2 - (lamM**2*I1s-2*I2s)/
        3*eye(3)) \ + kappa*(J-1) * eye(3)

def Ogden_3D(stretch, param):
    """Ogden model. 3D loading specified by stretches.
    param: [mu1, mu2, ..., alpha1, alpha2, kappa]. Returns true stress."""
    J = stretch[0] * stretch[1] * stretch[2]
    lam = J**(-1/3) * stretch
    N = round((len(param)-1)/2)
    mu = param[0:N]
    alpha = param[N:2*N]
    kappa = param[-1]
    Stress = kappa*(J-1)*eye(3)
    for i in range(N):
        fac = (2/J) * mu[i] / alpha[i]
        tmp = (lam[0]**alpha[i] + lam[1]**alpha[i] + lam[2]**alpha[i]) / 3
        Stress[0,0] = Stress[0,0] + fac * (lam[0]**alpha[i] - tmp)
        Stress[1,1] = Stress[1,1] + fac * (lam[1]**alpha[i] - tmp)
        Stress[2,2] = Stress[2,2] + fac * (lam[2]**alpha[i] - tmp)
    return Stress

```

```

def ETube_3D(stretch, param):
    """Extended Tube model. 3D loading specified by stretches.
    Param: Ge, Gc, delta, beta, kappa"""
    Ge = param[0]
    Gc = param[1]
    delta = param[2]
    beta = param[3]
    kappa = param[4]
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    lam = J**(-1/3) * array(stretch)
    I1s = trace(bstar)
    fac1 = (1+(1+I1s**2-4*I1s)*delta**2 + (5*I1s-I1s**2-6)*delta**4) / \
        (1 - (I1s-3)*delta**2)**2
    stressC = Gc/J * fac1 * devbstar
    fac2 = -2*Ge/(J*beta)
    tmp = (lam[0]**(-beta) + lam[1]**(-beta) + lam[2]**(-beta)) / 3
    stressE = zeros((3,3))
    stressE[0,0] = fac2 * (lam[0]**(-beta) - tmp)
    stressE[1,1] = fac2 * (lam[1]**(-beta) - tmp)
    stressE[2,2] = fac2 * (lam[2]**(-beta) - tmp)
    stressV = kappa*(J-1)*eye(3)
    return stressC + stressE + stressV

def Knowles_3D(stretch, param):
    """Knowles. 3D loading specified by stretches.
    param: mu, n, b, kappa. Returns true stress."""
    L1 = stretch[0]
    L2 = stretch[1]
    L3 = stretch[2]
    F = array([[L1,0,0], [0,L2,0], [0,0,L3]])
    J = det(F)
    bstar = J**(-2.0/3.0) * dot(F, F.T)
    I1s = trace(bstar)
    devbstar = bstar - trace(bstar)/3 * eye(3)
    return param[0]/J * (1+param[2]/param[1]*(I1s-3))**(param[1]-1) *
        devbstar \ + param[3]*(J-1) * eye(3)

def blatzko_3D(stretch, param):
    """Blatz-Ko. 3D loading specified by stretches.
    param[0]=mu"""
    F = array([[stretch[0],0,0], [0,stretch[1],0], [0,0,stretch[2]]])
    J = det(F)
    b = dot(F, F.T)
    b2 = dot(b, b)
    I1 = trace(b)
    I2 = 0.5 * (I1**2 - trace(b2))
    return param[0]/J**3.0 * (I1*b - b2 - (I2-J**3.0) * eye(3))

def hyperfoam_3D(stretch, param):
    """Hyperfoam model. 3D loading specified by stretches.
    param: [mu1, mu2, ..., alpha1, alpha2, ..., beta1, beta2, ...].
    Returns true stress."""
    J = stretch[0] * stretch[1] * stretch[2]
    lam = array(stretch)
    N = int(round(len(param)/3.0))
    mu = param[0:N]
    alpha = param[N:2*N]
    beta = param[2*N:3*N]
    Stress = zeros((3,3))
    for k in range(N):

```

```

fac = 2.0 * mu[k] / (J * alpha[k])
Stress[0,0] = Stress[0,0] + fac*(lam[0]**alpha[k]
- J**(-alpha[k]*beta[k]))
Stress[1,1] = Stress[1,1] + fac*(lam[1]**alpha[k]
- J**(-alpha[k]*beta[k]))
Stress[2,2] = Stress[2,2] + fac*(lam[2]**alpha[k]
- J**(-alpha[k]*beta[k]))
return Stress

```

## 5.10 Exercises

1. In what applications or situations can linear elasticity be a suitable material model for FE modeling of a polymer?
2. What is the definitions of the three invariants  $I_1$ ,  $I_2$ , and  $I_3$ ?
3. Derive the expression for true stress as a function of the applied stretch for incompressible uniaxial loading of a NH material model.
4. Why cannot the NH material model accurately predict the large strain response of most elastomers?
5. What is the difference between the NH and the Mooney-Rivlin hyperelastic models? Why is it more difficult to properly use the Mooney-Rivlin model?
6. Under what conditions is the Yeoh model unstable? What can be done to improve the stability of the Yeoh model?
7. Show that the Cauchy stress for the Arruda-Boyce EC model given in Equation (5.98) can be directly derived from Equation (5.99).
8. What is the definition of the molecular chain stretch that is defined in the Arruda-Boyce EC model?
9. In what ways are the Ogden model different than the  $I_1$ ,  $I_2$ ,  $I_3$  based hyperelastic models? Is the Ogden model always stable?
10. How many experimental tests are required to calibrate the Gent model?
11. What is the foundation of the extended tube model? What is the reason the extended tube model so accurately predicts experimental hyperelastic behaviors.

12. Which hyperelastic model is your favorite? Why?
13. What are the two most common ways to formulate anisotropic hyperelasticity?
14. What is the difference between traditional hyperelasticity and hyperelastic models for foams?
15. Can a Mullins effect model be used to predict hysteresis of a material exposed to cyclic loading?
16. What is Drucker stability and why is it important?
17. What are the strengths and limitations of hyperelasticity?

## References

- [1] L. Mullins, N.R. Tobin. Stress softening in rubber vulcanizates. Part I. Use of a strain amplification factor to describe the elastic behavior of filler-reinforced vulcanized rubber. *J. Appl. Polym. Sci.*, 9 1965 2993-3009.
- [2] R.W. Ogden, D.G. Roxburgh. A pseudo-elastic model for the Mullins effect in filled rubber. *Proc. R. Soc. Lond. A*, 455 1999 2861-2877.
- [3] H.J. Qi, M.C. Boyce. Constitutive model for stretch-induced softening of the stress-strain behavior of elastomeric materials. *J. Mech. Phys. Solids*, 52 2004 2187-2205.
- [4] S.H. Crandall, T.J. Lardner, N.C. Dahl. *An Introduction to the Mechanics of Solids*. McGraw Hill, New York, 1999.
- [5] J.M. Gere, S.P. Timoshenko. *Mechanics of Materials*. PWS Publishing Company, Boston, 1997.
- [6] M.F. Ashby, D.R. Jones. *Engineering Materials: An introduction to Their Properties and Applications*. Pergamon, New York, 1980.
- [7] J.E. Marsden, T.J.R. Hughes. *Mathematical Foundations of Elasticity*. Dover Publications, New York, 1994.
- [8] A.E. Love. *Treatise on the Mathematical Theory of Elasticity*. Dover Publications, New York, 1927.
- [9] S.P. Timoshenko. *Theory of Elasticity*. McGraw-Hill, New York, 1970.
- [10] I. Sokolnikoff. *Mathematical Theory of Elasticity*. McGraw-Hill, New York, 1956.
- [11] A.E. Green, W. Zerna. *Theoretical Elasticity*. Dover Publications, New York, 1968.

- [12] R.W. Ogden. *Non-linear Elastic Deformations*. Dover Publications, New York, 1997.
- [13] G.A. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. John Wiley & Sons, Chichester, 2000.
- [14] P. Chadwick. *Continuum Mechanics: Concise Theory and Problems*. Dover Publications, New York, 1999.
- [15] L.R.G. Treloar. Stress-strain data for vulcanized rubber under various types of deformation. *Trans. Faraday Soc.*, 40 1944 59-70.
- [16] L.R. Treloar. *The Physics of Rubber Elasticity*. Oxford University Press, Oxford, 1975.
- [17] O. Kratky, G. Porod. Röntgenuntersuchung gelöster fadenmoleküle. *Recl. Trav. Chim. Pays-Bas.*, 68 1949 1106-1122.
- [18] M. Fixman, J. Kovac. Polymer conformational statistics. III. Modified Gaussian models of stiff chains. *J. Chem. Phys.*, 58 1973 1564-1568.
- [19] J.F. Marko, E.D. Siggia. Stretching DNA. *Macromolecules*, 28 1995 8759-8770.
- [20] F. Reif. *Fundamentals of Statistical and Thermal Physics*. McGraw-Hill, New York, 1965.
- [21] P.J. Flory. *Statistical Mechanics of Chain Molecules*. Oxford University Press, Oxford, 1988.
- [22] W. Kuhn, F. Gr $\ddot{u}$ n. Beziehungen zwischen elastischen Konstanten und Dehnungsdoppelbrechung hochelastischer Stoffe. *Kolloid Z.*, 101 1942 248.
- [23] J.S. Bergstr $\ddot{o}$ m, M.C. Boyce. Mechanical behavior of particle filled elastomers. *Rubber Chem. Technol.*, 72 1999 633-656.
- [24] O.H. Yeoh. Some forms of the strain energy function for rubber. *Rubber Chem. Technol.*, 66 (5) 1993 754-771.
- [25] M. Kaliske, H. Rothert. On the finite element implementation of rubber-like materials at finite strains. *Eng. Comput.*, 14 (2) 1997 216-232.
- [26] S. Kawabata, Y. Yamashita, S. Yoshida. Mechanism of carbon-black reinforcement of rubber vulcanizate. *Rubber Chem. Technol.*, 68 1995 311-329.
- [27] HKS, Inc.. ABAQUS, Pawtucket, RI, ver. 6.14, 2014.
- [28] E.M. Arruda, M.C. Boyce. A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials. *J. Mech. Phys. Solids*, 41 (2) 1993 389-412.
- [29] M.C. Boyce. Direct comparison of the Gent and the Arruda-Boyce constitutive models of rubber elasticity. *Rubber Chem. Technol.*, 69 1997 781-785.

- [30] A. Cohen. A Pade approximant to the inverse Langevin function. *Rheol. Acta*, 30 1991 270-273.
- [31] PolyUMod. <http://PolyUMod.com/>.
- [32] A.N. Gent. A new constitutive relation for rubber. *Rubber Chem. Technol.*, 69 1996 59.
- [33] C.O. Horgan, G. Saccomandi. Constitutive modeling of rubber-like and biological materials with limiting chain extensibility. *Math. Mech. Solids*, 7 2002 353-371.
- [34] C.O. Horgan, J.G. Schwartz. Constitutive modeling and the trousers test for fracture of rubber-like materials. *J. Mech. Phys. Solids*, 53 2005 545-564.
- [35] J.K. Knowles. The finite anti-plane shear field near the tip of a crack for a class of incompressible elastic solids. *Int. J. Fracture*, 13 1977 611-639.
- [36] C. Suchocki. A finite element implementation of knowles stored-energy function: theory, coding and applications. *Arch. Mech. Eng.*, LVIII (3) 2011 319-346.
- [37] D. Hayman, C. Bergerson, S. Miller, M. Moreno, J.E. Moore. The effect of static and dynamic loading on degradation of PLLA stent fibers. *J. Biomech. Eng.*, 136 2014.
- [38] T. Sussman, K.J. Bathe. A model of incompressible isotropic hyperelastic material behavior using spline interpolations of tension-compression data. *Commun. Numer. Methods Eng.*, 25 2009 53-63.
- [39] M. Kaliske, G. Heinrich. An extended tube-model for rubber elasticity: statistical-mechanical theory and finite element implementation. *Rubber Chem. Technol.*, 72 1999 602-632.
- [40] Y. Fung. *Biomechanics: Mechanical Properties of Living Tissues*. Springer, New York, 1993.
- [41] A.J.M. Spencer. *Constitutive Theory for Strongly Anisotropic Solids. Continuum Theory and the Mechanics of Fiber-Reinforced Composites*, CISM Courses and Lectures No. 282, 1984 1-32.
- [42] W. Sun, M.S. Sacks. Finite element implementation of a generalized Fung-elastic constitutive model for planar soft tissues. *Biomech. Model Mechanobiol.*, 4 2005 190-199.
- [43] J.S. Bergström. *Large Strain Time-Dependent Behavior of Elastomeric Materials*. Ph.D. thesis, MIT 1999.
- [44] J.E. Bischoff, E.A. Arruda, K. Grosh. A Microstructurally Based Orthotropic Hyperelastic Constitutive Law. *Trans. ASME* 2002 570-579.

- [45] Y. Lanir, Y.C. Fung. Two-dimensional mechanical properties of rabbit skin. II. Experimental results. *Journal of Biomechanics*, 7 1974 171-182.
- [46] G.A. Holzapfel, T.C. Gasser, R.W. Ogden. A new constitutive framework for arterial wall mechanics and a comparative study of material models. *J. Elasticity*, 61 2000 1-48.
- [47] T.C. Gasser, R.W. Ogden, G.A. Holzapfel. Hyperelastic modelling of arterial layers with distributed collagen fibre orientations. *J. R. Soc. Interface*, 3 2006 15-35.
- [48] P.J. Blatz, W.L. Ko. Application of finite elastic theory to the deformation of rubber materials. *Trans. Soc. Rheol.* 1962 223-251.
- [49] C.O. Horgan. Material instabilities for large deformations of the generalized Blatz-Ko material. *Appl. Mech. Rev.*, 50 S93-S96.
- [50] B. Storåkers. On material representation and constitutive branching in finite compressible elasticity. *J. Mech. Phys. Solids*, 34 (2) 1986 125-145.
- [51] J.A.C. Harwood, L. Mullins, A.R. Payne. Stress softening in natural rubber vulcanizates. Part II. Stress softening in pure gum and filler loaded rubbers. *J. Appl. Polym. Sci.*, 9 1965 3011-3021.
- [52] J.A.C. Harwood, A.R. Payne. Stress softening in natural rubber vulcanizates. Part III. Carbon black-filled vulcanizates. *J. Appl. Polym. Sci.*, 10 1966 315-324.
- [53] J.A.C. Harwood, A.R. Payne. Stress softening in natural rubber vulcanizates. Part IV. Unfilled vulcanizates. *J. Appl. Polym. Sci.*, 10 1966 1203-1211.
- [54] J. Diani, B. Fayolle, P. Gilormini. A review of Mullins effect. *Eur. Polym. J.*, 45 2009 601-612.

# 6 Linear Viscoelasticity

---

## Chapter Outline

6.1	Introduction	310
6.2	Small Strain Linear Viscoelasticity	310
6.2.1	Stress Relaxation Behavior	313
6.2.2	Monotonic Loading Response	314
6.2.3	Cyclic Loading Response	320
6.2.4	Experimental Determination of the Storage and Loss Moduli	322
6.2.5	Representing Linear Viscoelasticity Using Spectra	328
6.2.6	Computer Implementation	329
	Matlab Implementation	329
	Python Implementation	330
6.3	Large Strain Linear Viscoelasticity	331
6.3.1	Numerical Implementation	332
6.4	Creep Compliance Behavior	335
6.4.1	Relationships Between Creep Compliance and Relaxation Modulus	336
6.5	Differential Form of Linear Viscoelasticity	337
6.5.1	Rheological Models	338
	Maxwell Model	338
	Multi-Network Maxwell Model	339
6.6	The Use of Shift Functions to Generalize Linear Viscoelasticity Theory	340
6.6.1	Time-Temperature Equivalence	341
6.6.2	Vertical Shifts	345
6.7	Use of Linear Viscoelasticity in Polymer Modeling	345
6.8	Exercises	349
	References	350



## 6.1 Introduction

Many polymeric materials exhibit a combination of elastic and viscous responses when subjected to external loads and displacements. The simplest way to model this combined behavior is through *linear viscoelasticity*, which is a material model framework that has been used extensively for many years and its mathematical foundation has been extensively studied [1–3].

The following sections present the basic theory of linear viscoelasticity in relative detail, more advanced concepts and models that extend linear viscoelasticity are discussed in later chapters. The theory that is presented is derived from simple arguments, and is closely tied to the linear elastic and hyperelastic models that were discussed in the previous chapter.

## 6.2 Small Strain Linear Viscoelasticity

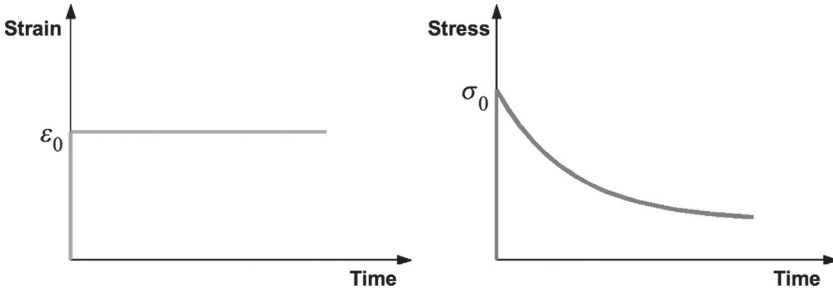
This section summarizes the theory for small strain linear viscoelasticity. That is, a material model framework based on linear elasticity with linear viscoelasticity. The more general large strain viscoelasticity that is based on hyperelasticity and linear viscoelasticity is discussed in [Section 6.3](#).

The foundation of linear viscoelasticity is Boltzmann's superposition principle [4]. One way to state this linear superposition principle is through the statement:

*Each loading step makes an independent contribution to the final state.*

This formulation of the Boltzmann superposition principle can be used to create an integral equation representation of linear viscoelasticity. The approach is to perform a thought experiment in which a step function in strain is applied:  $\varepsilon(t) = \varepsilon_0 H(t)$ , and the stress response  $\sigma(t)$  is measured, see [Figure 6.1](#). In this equation  $H(t)$  is the Heaviside step function defined by:

$$H(t) = \begin{cases} 0, & \text{if } t < 0, \\ 1/2, & \text{if } t = 0, \\ 1, & \text{if } t > 0. \end{cases} \quad (6.1)$$



**Figure 6.1** Stress relaxation experiment in which the applied strain is rapidly ramped from 0 to  $\varepsilon_0$  and then held constant during which the stress response is monitored.

In this section we are only considering small deformations, and only load cases with one non-zero stress component (such as uniaxial loading).

The stress response from the strain jump experiment can be used to define a stress relaxation modulus:

$$E_r(t) \equiv \frac{\sigma(t)}{\varepsilon_0}. \quad (6.2)$$

Note that the applied strain jump  $\varepsilon_0$  does not have to be infinitesimal when calculating the stress relaxation modulus due to the assumed superposition principle.

To develop a model capable of predicting the stress response due to an arbitrary applied strain history, and not only a step in strain, we will start by decomposing the strain history into a sum of infinitesimal strain steps:

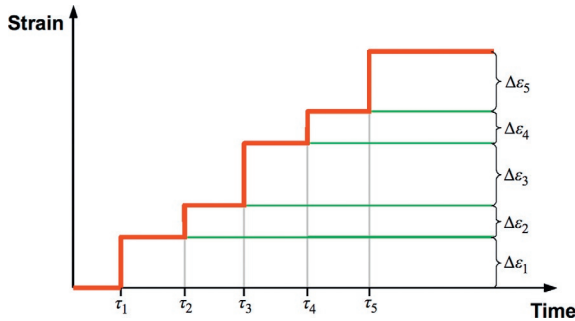
$$\varepsilon(t) = \sum_{i=1}^{\infty} \Delta\varepsilon_i H(t - \tau_i), \quad (6.3)$$

where  $\Delta\varepsilon_i$  is the strain increment applied at time  $\tau_i$ , see [Figure 6.2](#).

The total stress response from applying this strain history can be obtained from the superposition principle:

$$\sigma(t) = \sum_{i=1}^{\infty} \Delta\varepsilon_i E_R(t - \tau_i). \quad (6.4)$$

This stress response can be written in integral form as the number of strain increments goes to infinity:



**Figure 6.2** The applied strain history can be represented as number of discrete time increments.

$$\sigma(t) = \int_{-\infty}^t E_R(t - \tau) d\epsilon(t) = \int_{-\infty}^t E_R(t - \tau) \frac{d\epsilon(\tau)}{d\tau} d\tau. \quad (6.5)$$

Note that if the material response is purely elastic (i.e.  $E(t) = E_0$ ), then the stress response becomes equal to Hookes law:  $\sigma(t) = E_0\epsilon(t)$ . Also, once  $E_R(t)$  has been determined then Equation (6.5) can be used to predict the stress response due to any imposed strain history.

In order to create a complete material model, the results presented so far need to be generalized into a three-dimensional form suitable for arbitrary deformation histories and finite element analysis. The most common way to do this is to separate the stress and strain into deviatoric and volumetric parts, similarly to what is commonly done for hyperelastic models as discussed in Chapter 5:

$$\sigma(t) = \int_0^t 2\mu_R(t - \tau) \dot{\epsilon}_{\text{dev}} d\tau + \int_0^t \kappa_R(t - \tau) \dot{\epsilon}_{\text{vol}} d\tau. \quad (6.6)$$

In this equation  $\sigma$  is the Cauchy stress tensor,  $\mu_R(t)$  is the stress relaxation shear modulus,  $\dot{\epsilon}_{\text{dev}} = \frac{d}{dt}[\text{dev}[\epsilon]]$  the time derivative of the applied deviatoric strain tensor,  $\kappa_R(t)$  the stress relaxational bulk modulus, and  $\dot{\epsilon}_{\text{vol}} = \frac{d}{dt}[\text{vol}[\epsilon]]$  the time derivative of the applied volumetric strains.

In this formulation only two relaxation moduli need to be determined in order to predict any arbitrary deformation: the shear and bulk relaxation moduli. For most polymeric materials, the volumetric relaxation is typically much smaller and less

influential than the deviatoric relaxation. It is therefore often possible to neglect the volumetric flow response. This assumption makes it possible to obtain the shear relaxation modulus from the uniaxial relaxation modulus by:

$$\mu_R(t) = \frac{E_R(t)}{3 - E_R(t)/(3\kappa)}, \quad (6.7)$$

which can be approximated as  $\mu_R(t) = E_R(t)/3$  if  $\kappa \gg E$ .

The linear viscoelastic stress-strain equations in Equation (6.6) can be generalized to anisotropic materials (see also Section 5.2.2) by replacing the shear and bulk relaxation moduli with anisotropic relaxation moduli:

$$\sigma_{ij}(t) = C_{ijkl}(0)\varepsilon_{kl}(t) + \int_0^t \varepsilon_{kl}(t - \tau) \frac{dC_{ijkl}(\tau)}{d\tau} d\tau. \quad (6.8)$$

Here  $C_{ijkl}$  is the time-dependent stiffness relaxation tensor that need to be determined from experimental data.

### 6.2.1 Stress Relaxation Behavior

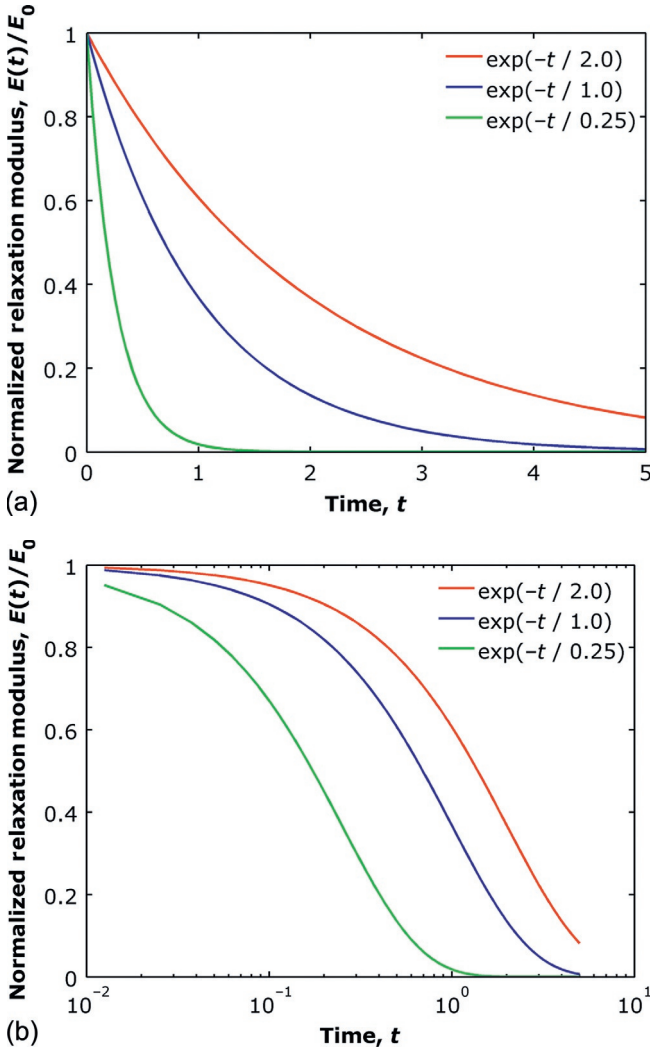
As illustrated in the previous section, a linear viscoelastic model is completely characterized by its stress relaxation moduli. This section presents a selection of commonly used functional forms of the stress relaxation moduli.

First, as an example, consider a polymer sample that is uniaxially loaded and that has a stress relaxation modulus that is exponentially decaying:

$$E_R(t) = \begin{cases} E_0 \exp(-t/\tau_0), & \text{if } t \geq 0, \\ 0, & \text{if } t < 0, \end{cases} \quad (6.9)$$

where  $E_0$  is the instantaneous Young's modulus, and  $\tau_0$  the characteristic relaxation time, see Figure 6.3 for a graphical representation. The characteristic relaxation time is here defined as the time at which the modulus has decayed down to 36.7%<sup>1</sup> of its original value.

<sup>1</sup>36.7% is an approximation of the value  $1/e$ .



**Figure 6.3** Relaxation modulus as a function of time. Figure (a) shows the relaxation modulus as a function of time on a linear scale, and figure (b) shows the relaxation modulus as a function of time on a logarithmic scale.

### 6.2.2 Monotonic Loading Response

Now consider a load case in which the applied strain is increasing linearly with time:

$$\varepsilon(t) = \begin{cases} 0, & \text{if } t < 0, \\ \dot{\varepsilon}_0 t, & \text{if } t \geq 0. \end{cases} \quad (6.10)$$

Inserting this strain history into Equation (6.5) gives the stress

$$\sigma(t) = \int_0^t E_0 \dot{\varepsilon}_0 \exp\left[-\frac{(t-\tau)}{\tau_0}\right] d\tau, \quad (6.11)$$

which can be simplified to:

$$\sigma(t) = E_0 \dot{\varepsilon}_0 \tau_0 \left[1 - \exp\left(\frac{-t}{\tau_0}\right)\right], \quad (6.12)$$

or when expressed in terms of applied strains:

$$\sigma(t) = E_0 \dot{\varepsilon}_0 \tau_0 \left[1 - \exp\left(\frac{-\varepsilon}{\dot{\varepsilon} \tau_0}\right)\right]. \quad (6.13)$$

This stress-strain response is plotted in [Figure 6.4](#) illustrating that a small characteristic relaxation time  $\tau_0$  corresponds to fast relaxation and more viscous response.

From Equation (6.11) it is also clear that at a fixed time, the stress response is always predicted to be proportional to the applied strain rate; but at a fixed strain, the stress is not a linear function of the applied strain rate. Furthermore, by introducing a characteristic strain  $\hat{\varepsilon} \equiv \dot{\varepsilon} \tau_0$  it is clear that for  $\varepsilon \gg \hat{\varepsilon}$  the stress  $\sigma \rightarrow E_0 \hat{\varepsilon} = E_0 \dot{\varepsilon} \tau_0$ . Also, when the applied strain is equal to  $\hat{\varepsilon}$  the stress will have reached 63% of its final value.

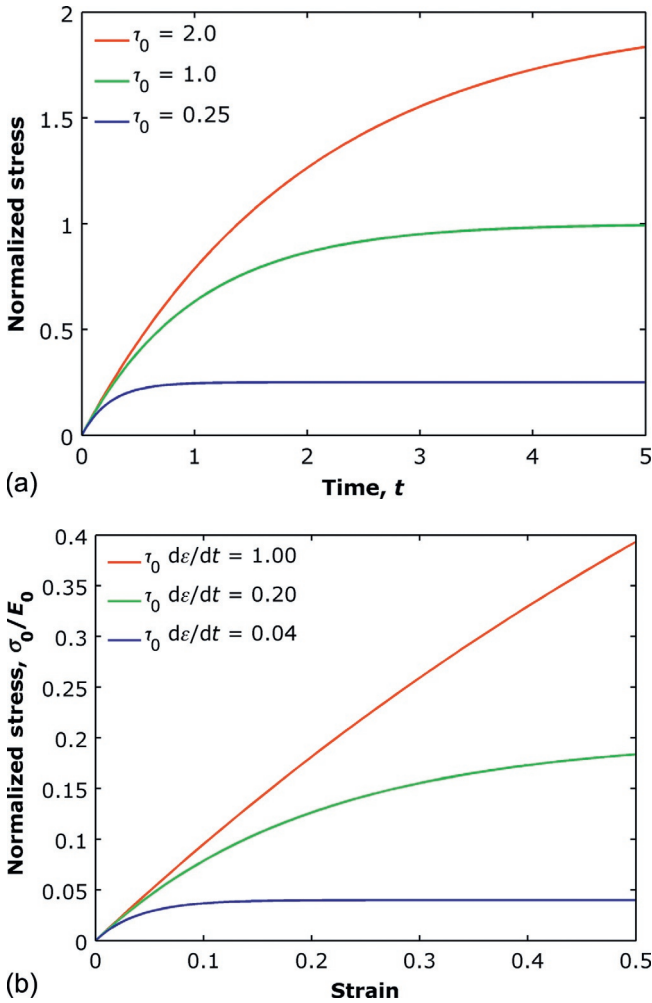
More commonly, the stress relaxation function is written as the following normalized series expansion:

$$g_R(t) = \frac{E_R(t)}{E_0} = 1 - \sum_{i=1}^N g_i (1 - e^{-t/\tau_i}) \quad (6.14)$$

$$= \left(1 - \sum_{i=1}^N g_i\right) + \sum_{i=1}^N g_i e^{-t/\tau_i}. \quad (6.15)$$

This relaxation function is used, for example, by Abaqus [5], and is called a *Prony series*.

[Figure 6.5](#) shows one example of a two-term Prony series plotted on a logarithmic time scale. The main differences between this relaxation function and the simple exponential relaxation



**Figure 6.4** (a) Stress response as a function of time. (b) Stress response as a function of strain.

function in Equation (6.9) are that the modulus can stay finite as  $t \rightarrow \infty$ , and that multiple terms can be used to fit the relaxation function to experimental data.

The stress response in monotonic uniaxial loading for the stress relaxation modulus in Figure 6.5 is shown in Figure 6.6.

Another interesting but less used functional form of the relaxation modulus is the stretched exponential (often called the KWW form [6, 7]):

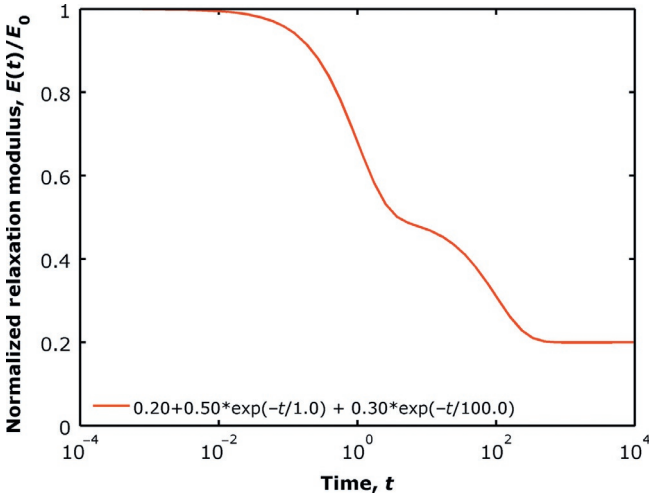


Figure 6.5 Exemplar 2-term Prony series.

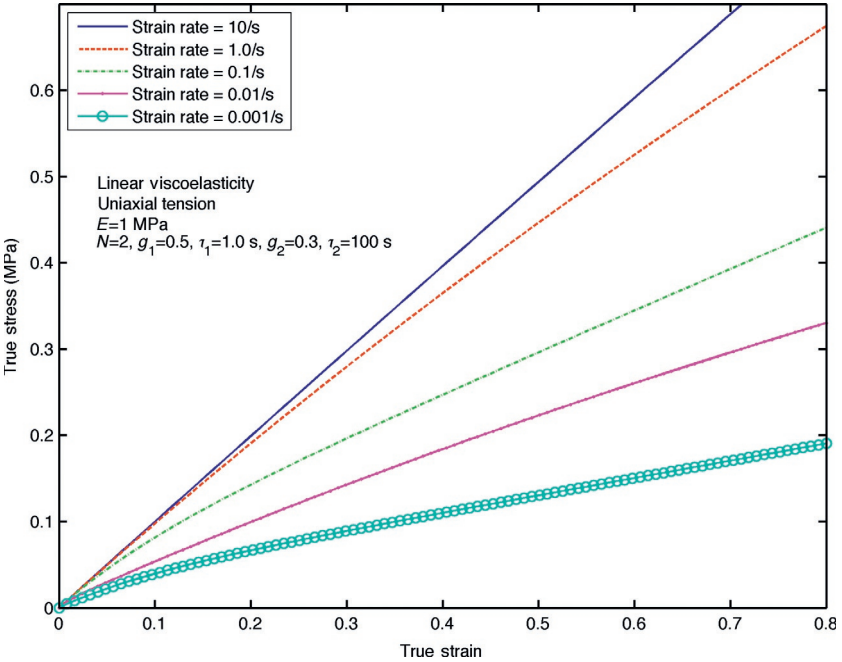
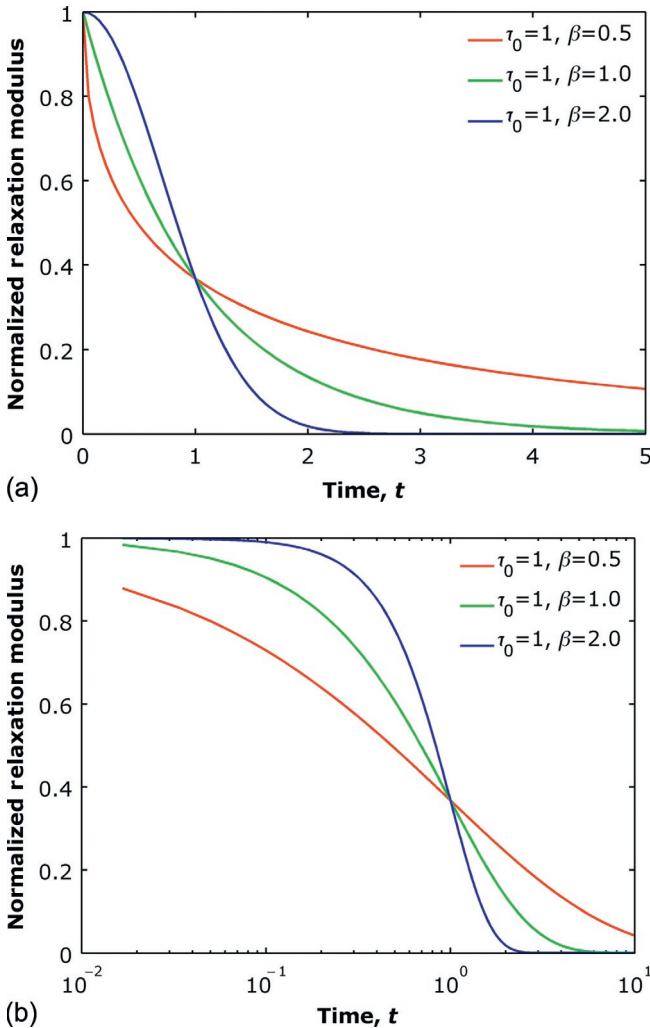


Figure 6.6 Predicted stress response from the Prony series shown in Figure 6.5.



$$E(t) = E_0 \exp \left[ - \left( \frac{t}{\tau_0} \right)^\beta \right], \quad (6.16)$$

where  $E_0$ ,  $\tau_0$ , and  $\beta$  are material parameters, see [Figure 6.7](#). With this representation, the relaxation modulus decreases down to 36.7% of its original value at the time  $t = \tau_0$ , and  $\beta$  specifies the shape of the relaxation function.

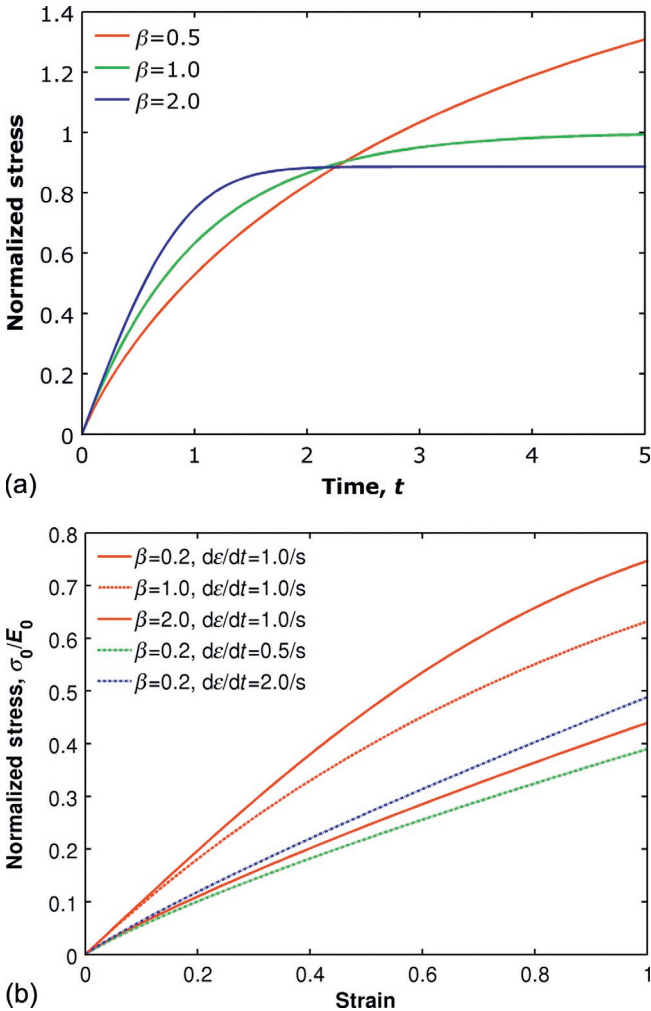


**Figure 6.7** Stretched exponential stress relaxation modulus as a function of time. (a) Linear time, (b) Logarithmic time.

If we again consider a constant strain rate loading situation, the stress response becomes:

$$\sigma(t) = \int_0^t E_0 \dot{\epsilon} \exp \left[ - \left( \frac{t}{\tau_0} \right)^\beta \right] dt, \quad (6.17)$$

which is not easy to solve in closed-form but can be solved numerically, see [Figure 6.8](#).



**Figure 6.8** Stress response using the stretched exponential stress relaxation modulus,  $\tau_0 = 1$  s.

Figure 6.8 shows that the parameter  $\beta$  strongly influences the shape of the stress-strain response.

### 6.2.3 Cyclic Loading Response

One of the more common applications of linear viscoelasticity is for predictions of the small-strain dynamic response of polymers. This section presents in more detail the response of a general linear viscoelastic model when exposed to a sinusoidal applied strain. Specifically, consider a case when the applied strain is given by:

$$\varepsilon(t) = \begin{cases} \varepsilon_0 \sin(\omega t), & \text{if } t \geq 0, \\ 0, & \text{if } t < 0. \end{cases} \quad (6.18)$$

The resulting stress response for this strain history can be obtained from Equation (6.5) giving:

$$\sigma(t) = \int_0^\infty E(s) \omega \varepsilon_0 \cos[\omega(t-s)] ds, \quad (6.19)$$

where  $s \equiv t - \tau$ . By using the relationship  $\cos(\alpha - \beta) = \cos \alpha \cos \beta + \sin \alpha \sin \beta$ , this equation can be expanded into

$$\begin{aligned} \sigma(t) = \varepsilon_0 \sin(\omega t) & \left[ \omega \int_0^\infty E(s) \sin(\omega s) ds \right] \\ & + \varepsilon_0 \cos(\omega t) \left[ \omega \int_0^\infty E(s) \cos(\omega s) ds \right]. \end{aligned} \quad (6.20)$$

Note, the integrals in Equation (6.20) only converge if

$$\lim_{s \rightarrow \infty} E(s) = 0. \quad (6.21)$$

By defining two frequency-dependent functions: the *storage modulus*  $E'(\omega)$  and the *loss modulus*  $E''(\omega)$ , the stress response can be defined by

$$\sigma(t) = \varepsilon_0 [E'(\omega) \sin(\omega t) + E''(\omega) \cos(\omega t)], \quad (6.22)$$

where

$$\begin{cases} E'(\omega) = \omega \int_0^\infty E(s) \sin(\omega s) ds, \\ E''(\omega) = \omega \int_0^\infty E(s) \cos(\omega s) ds. \end{cases} \quad (6.23)$$

The stress response can also be written

$$\sigma(t) = \sigma_0 \sin(\omega t + \delta), \quad (6.24)$$

$$= \sigma_0 \sin(\omega t) \cos \delta + \sigma_0 \cos(\omega t) \sin \delta. \quad (6.25)$$

Hence,

$$\varepsilon_0 E'(\omega) = \sigma_0 \cos \delta, \quad (6.26)$$

$$\varepsilon_0 E''(\omega) = \sigma_0 \sin \delta, \quad (6.27)$$

giving

$$\tan \delta = \frac{E''}{E'}. \quad (6.28)$$

It is sometimes convenient to introduce a complex variable based notation for the dynamic moduli  $E'$  and  $E''$ :

$$E^* = \frac{\sigma^*}{\varepsilon^*} = E' + iE''. \quad (6.29)$$

It is also possible to consider a stress driven oscillation giving a complex compliance:

$$J^* = \frac{\varepsilon^*}{\sigma^*} = \frac{1}{G^*} = J' - iJ''. \quad (6.30)$$

This approach is discussed in more detail in various texts [1]. It can be shown, for example, that the relationships between  $J'$ ,  $J''$ ,  $E'$ , and  $E''$  are:

$$J' = \frac{G'}{G'^2 + G''^2}, \quad (6.31)$$

$$J'' = \frac{G''}{G'^2 + G''^2}, \quad (6.32)$$

$$G' = \frac{J'}{J'^2 + J''^2}, \quad (6.33)$$

$$G'' = \frac{J''}{J'^2 + J''^2}. \quad (6.34)$$

These relations, whose derivations are left as an exercise, are not pursued in more detail in this text.

As a last example consider a shear relaxation function that is given by an exponential function:

$$G(t) = \begin{cases} G_0 \exp(-\alpha t) & \text{if } t \geq 0, \\ 0 & \text{if } t < 0 \end{cases} \quad (6.35)$$

and an applied strain that is sinusoidal:

$$\varepsilon(t) = \begin{cases} 0 & \text{if } t < 0, \\ \varepsilon \sin(\omega_0 t) & \text{if } t \geq 0. \end{cases} \quad (6.36)$$

In this case it can be shown that the stress response is given by the following equations:

$$\begin{aligned} \sigma(t) &= \int_0^t G_0 \exp(-\alpha \tau) \varepsilon \omega \cos(\omega \tau) d\tau \\ &= \frac{G_0 \varepsilon \omega}{\alpha^2 + \omega^2} [\alpha \cos(\omega t) + \omega \sin(\omega t) - \alpha \exp(-\alpha t)]. \end{aligned} \quad (6.37)$$

The stress response consists of a transient term superimposed on a sinusoidal term that is out of phase with the applied strain. The response due to a sinusoidal strain history is studied in more detail in the next section.

#### 6.2.4 Experimental Determination of the Storage and Loss Moduli

As discussed in Chapter 2, the dynamic properties of a polymer can be determined using a wide variety of loading modes and experimental techniques. This section presents the use of uniaxial sinusoidal loading to determine the storage modulus, loss modulus,  $\tan(\delta)$ , and the amount of hysteresis. The results presented here can easily be converted to other loading modes, for example simple or pure shear.

First, consider an experiment in which a specimen is uniaxially loaded with a sinusoidal strain history:

$$\varepsilon(t) = \varepsilon_m + \varepsilon_a \sin(\omega t), \quad (6.38)$$

where  $\varepsilon_m$  is the mean strain,  $\varepsilon_a$  is the strain amplitude, and  $\omega$  is the angular frequency of the applied strain. Assume that the experimentally determined stress response is also sinusoidal with a mean stress  $\sigma_m$  and a stress amplitude  $\sigma_a$ , but is shifted with a phase angle  $\delta$ :

$$\sigma(t) = \sigma_m + \sigma_a \sin(\omega t + \delta). \quad (6.39)$$

This stress response can be expanded into a term that is in-phase with the applied strain and a term that is  $90^\circ$  out of phase with the applied strain:

$$\sigma(t) = \sigma_m + \sigma_a \cos(\delta) \sin(\omega t) + \sigma_a \sin(\delta) \cos(\omega t). \quad (6.40)$$

The *storage modulus* ( $E'$ ) is defined as the stiffness of the response that is in-phase with the applied strain, and the *loss modulus* ( $E''$ ) is defined as the stiffness of the response that is lagging  $90^\circ$  behind the applied strain. Hence,  $E'$  and  $E''$  can be defined by<sup>2</sup>:

$$\sigma(t) \equiv \sigma_m + \varepsilon_a E' \sin(\omega t) + \varepsilon_a E'' \cos(\omega t). \quad (6.41)$$

By comparing terms between Equation (6.40) and Equation (6.41), and by defining  $E^* = \sigma_a/\varepsilon_a$ , the storage modulus  $E'$  and the loss modulus  $E''$  can be determined from the following equations:

$$E' = E^* \cos(\delta), \quad (6.42)$$

$$E'' = E^* \sin(\delta). \quad (6.43)$$

The amount of energy loss (hysteresis) per unit reference volume per load cycle is given by:

$$u_{\text{loss}} = \oint_{1 \text{ cycle}} \sigma \, d\varepsilon = \int_0^{2\pi/\omega} \sigma(t) \dot{\varepsilon} \, dt. \quad (6.44)$$

Inserting (6.38) and (6.40) into (6.44) gives

$$u_{\text{loss}} = \pi \varepsilon_a \sigma_a \sin(\delta), \quad (6.45)$$

which can also be written

$$u_{\text{loss}} = \pi \varepsilon_a^2 E''. \quad (6.46)$$

The quantity  $\tan(\delta)$  can therefore also be calculated from:

$$\tan(\delta) = \frac{u_{\text{loss}}}{\sqrt{(\pi \varepsilon_a \sigma_a)^2 - u_{\text{loss}}^2}}. \quad (6.47)$$

<sup>2</sup>The storage modulus and the loss modulus are secant moduli.

In summary, if the applied strain is sinusoidal with the amplitude  $\varepsilon_a$ , the resulting stress is sinusoidal with known amplitude  $\sigma_a$ , and the amount of hysteresis per cycle ( $u_{\text{loss}}$ ) are known, then the storage modulus, loss modulus, and  $\tan(\delta)$  can be calculated from:

$$E'' = \frac{u_{\text{loss}}}{\pi \varepsilon_a^2}, \quad (6.48)$$

$$E^* = \frac{\sigma_a}{\varepsilon_a}, \quad (6.49)$$

$$E' = \sqrt{(E^*)^2 - (E'')^2}, \quad (6.50)$$

$$\tan(\delta) = \frac{E''}{E'}. \quad (6.51)$$

### Example

Assume that the experimentally determined stress response from a uniaxial test with a sinusoidal applied strain is given by a function  $\sigma(t)$ . Then, approximate  $\sigma(t)$  by sum of a sine and a cosine term<sup>3</sup>:

$$\sigma(t) \approx \sigma_m + A \sin(\omega t) + B \cos(\omega t), \quad (6.52)$$

where  $\sigma_m = \frac{\omega}{2\pi} \int_0^{2\pi/\omega} \sigma(t) dt$  is the mean stress. In order to determine the storage modulus ( $E'$ ) and the loss modulus ( $E''$ ) multiply each side of (6.52) by  $\sin(\omega t)$  or  $\cos(\omega t)$ , and then integrate<sup>4</sup> from 0 to  $N\pi/\omega$ :

$$\begin{aligned} \int_0^{N\pi/\omega} \sigma(t) \sin(\omega t) dt &= A \int_0^{N\pi/\omega} \sin^2(\omega t) dt \\ &+ B \int_0^{N\pi/\omega} \sin(\omega t) \cos(\omega t) dt, \end{aligned} \quad (6.53)$$

$$\begin{aligned} \int_0^{N\pi/\omega} \sigma(t) \cos(\omega t) dt &= A \int_0^{N\pi/\omega} \sin(\omega t) \cos(\omega t) dt \\ &+ B \int_0^{N\pi/\omega} \cos^2(\omega t) dt. \end{aligned} \quad (6.54)$$

<sup>3</sup>This is a first order Fourier series expansion of  $\sigma(t)$ .

<sup>4</sup>The variable  $N$  is an integer larger than or equal to 1.

This set of equations is solved if

$$A = \frac{2\omega}{N\pi} \int_0^{N\pi/\omega} \sigma(t) \sin(\omega t) dt, \quad (6.55)$$

$$B = \frac{2\omega}{N\pi} \int_0^{N\pi/\omega} \sigma(t) \cos(\omega t) dt. \quad (6.56)$$

As before, the definition of  $E'$  and  $E''$  is given by

$$\sigma(t) \equiv \sigma_m + \varepsilon_a E' \sin(\omega t) + \varepsilon_a E'' \cos(\omega t). \quad (6.57)$$

By comparing terms between (6.52), (6.55)–(6.57), the storage modulus  $E'$  and the loss modulus  $E''$  can then be determined from the following equations:

$$E' = \frac{2\omega}{N\pi} \int_0^{N\pi/\omega} \frac{\sigma(t)}{\varepsilon_a} \sin(\omega t) dt, \quad (6.58)$$

$$E'' = \frac{2\omega}{N\pi} \int_0^{N\pi/\omega} \frac{\sigma(t)}{\varepsilon_a} \cos(\omega t) dt. \quad (6.59)$$

### Example

Now consider a case in which the applied strain is not sinusoidal but instead triangular, square, or some other similar waveform specified by a function  $f(\cdot)$ :

$$\varepsilon(t) = \varepsilon_m + \varepsilon_a f(\omega t), \quad (6.60)$$

where as before  $\varepsilon_a$  is the applied strain amplitude. Assume that the stress response in this case is given by a term that is in-phase with the applied strain and a term that is  $90^\circ$  out-of phase with the applied strain:

$$\sigma(t) = Af(\omega t) + Bf\left(\omega t + \frac{\pi}{2\omega}\right). \quad (6.61)$$

Then, if  $f(x)$  and  $f(x + \pi/2)$  are orthogonal over the interval  $[0, N\pi]$ , the constants  $A$  and  $B$  can be found by multiplying (6.61) by  $f(\omega t)$  or  $f(\omega t + \pi/2\omega)$  and then integrating over  $[0, N\pi/\omega]$ . Using this approach the storage modulus and loss modulus can be calculated from:



$$E' = \frac{1}{\varepsilon_a} \frac{\int_0^{N\pi/\omega} \sigma(t) f(\omega t) dt}{\int_0^{N\pi/\omega} f^2(\omega t) dt}, \quad (6.62)$$

$$E'' = \frac{1}{\varepsilon_a} \frac{\int_0^{N\pi/\omega} \sigma(t) f\left(\omega t + \frac{\pi}{2\omega}\right) dt}{\int_0^{N\pi/\omega} f^2\left(\omega t + \frac{\pi}{2\omega}\right) dt}. \quad (6.63)$$

### Example

An alternative approach to determine the storage and loss moduli from experimental time-strain-stress data is to use the Fourier transform approach [8]:

1. Take the Fourier transform of the strain and stress data. Shift the means strain to be zero if necessary.
2. Calculate the complex modulus from the ratio of the transformed stress to transformed strain, at the location where the transformed strain has the largest magnitude.
3. The storage modulus is given the real part of the complex modulus, and the loss modulus is given by the imaginary part of the complex modulus.

Python code illustrating this approach is listed below:

Python Code: "analyze\_data.py"

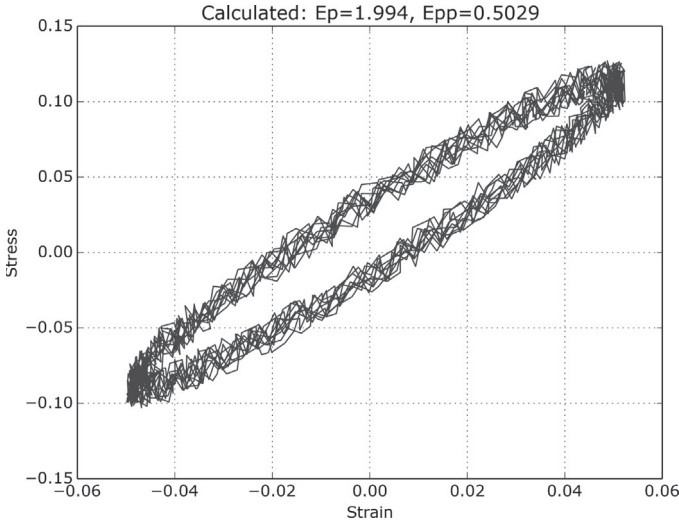
```
from pylab import *

### Input
freq_Hz = 1.4
strain_mean = 0.0 # subtract out the mean strain
strain_amp = 0.05
stress_mean = 0.0
nr_cycles = 10.0
nr_dpts_per_cycle = 144.0
Ep = 2.0
Epp = 0.5

### Create virtual time-strain-stress data
freq_rad = freq_Hz * 2.0 * pi
time = linspace(0.0, 1.0 * 2.0*pi*nr_cycles / freq_rad, \
    nr_cycles * nr_dpts_per_cycle)
strain = strain_mean + strain_amp * sin(time * freq_rad) + \
    0.05 * strain_amp * random(size(time))
stress = stress_mean + strain_amp * Ep * sin(time * freq_rad) + \
    strain_amp * Epp * cos(time * freq_rad) + \
    0.05 * Epp * random(size(time))

### Extract the storage and loss moduli
hanning_window = hanning(size(time))
fft_strain = fft(strain * hanning_window)
fft_stress = fft(stress * hanning_window)
fft_strain_mag = abs(fft_strain)
i = argmax(fft_strain_mag)
Gstar = fft_stress[i] / fft_strain[i]
Ep_calc = Gstar.real
Epp_calc = Gstar.imag
```

The results from running the code is shown in the following figure.



Note the following restrictions on the dynamic properties discussed in this section:

- It may or may not be possible to determine  $E'$  and  $E''$  for a material that is *non-linear* viscoelastic, or for a case when the applied mean strain and/or strain amplitudes are large. The usefulness of  $E'$  and  $E''$  depends on whether the resulting stress response is proportional to a scaled and shifted representation of the input strain. If the resulting stress has a different shape (e.g. due to yielding, or other non-linear behavior) then it is not appropriate to utilize  $E'$  and  $E''$  to describe the material response.
- It is not possible to determine  $E'$  and  $E''$  from tests using monotonic strain histories. The applied strain need to oscillate with a constant frequency, strain amplitude, and wave form.

### 6.2.5 Representing Linear Viscoelasticity Using Spectra

Several different means of specifying viscoelastic mechanical properties have been given in the previous section. Specifically, relaxation functions, creep functions, and complex moduli have been discussed. Another way of characterizing the material response is through *spectra*. To introduce the concept of a relaxation spectrum consider a relaxation modulus given by a series of exponential terms:

$$E(t) = \sum_{i=1}^n E_i \exp \left[ \frac{-t}{\tau_i} \right]. \quad (6.64)$$

In the limit as  $i \rightarrow \infty$ , the summation can be replaced by an integral:

$$E(t) = \int_0^{\infty} f(\tau) \exp \left[ \frac{-t}{\tau} \right] d\tau. \quad (6.65)$$

The function  $f(\tau)$  is called the *relaxation time spectrum*. In practice it is often more convenient to use a logarithmic time scale when defining the relaxation time spectrum

$$G(t) = \int_{-\infty}^{+\infty} H(\tau) \exp \left[ \frac{-t}{\tau} \right] d(\ln \tau) + G(\infty). \quad (6.66)$$

Similarly, a retardation time spectrum can be defined by

$$J(t) = \int_{-\infty}^{+\infty} L(\tau) \left[ 1 - \exp \left( \frac{-t}{\tau} \right) \right] d(\ln \tau) + J(\infty). \quad (6.67)$$

It is possible to solve for  $H(t)$  and  $L(t)$  by using Laplace transforms, see [1] for more details on the use of relaxation spectra. The theory of relaxation spectra is elegant and provides a concise way to represent linear viscoelastic behavior. It is, however, of limited use in practice since the major finite element programs specify the linear viscoelastic response using Prony series instead of relaxation spectra.

Finally, even if a relaxation spectrum is chosen, the Boltzmann's superposition principle does not allow for a sigmoidal shaped stress-strain response that is typical for polymers at intermediate to large strains.

## 6.2.6 Computer Implementation

To make the linear viscoelasticity theory more concrete it is useful to consider a simple one-dimensional computational implementation. This section summarizes two implementations, one using Matlab and one using Python.

### Matlab Implementation

The Matlab files `mat_LVE.m` and `test_mat_LVE.m` illustrate one possible implementation of small strain linear viscoelasticity for uniaxial loading. The algorithm that is used here is directly follows the theory that was derived in previous sections, but the code is not numerically efficient.

The `mat_LVE()` function takes a time vector, a strain vector, and a vector with material parameters where the first parameter is the initial Young's modulus and the remaining parameters are the pairs of Prony series parameters  $[g_i, \tau_i]$ . The code is using the built-in Matlab function `quad()` to numerically integrate the integral.

Matlab File Name: "mat\_LVE.m"

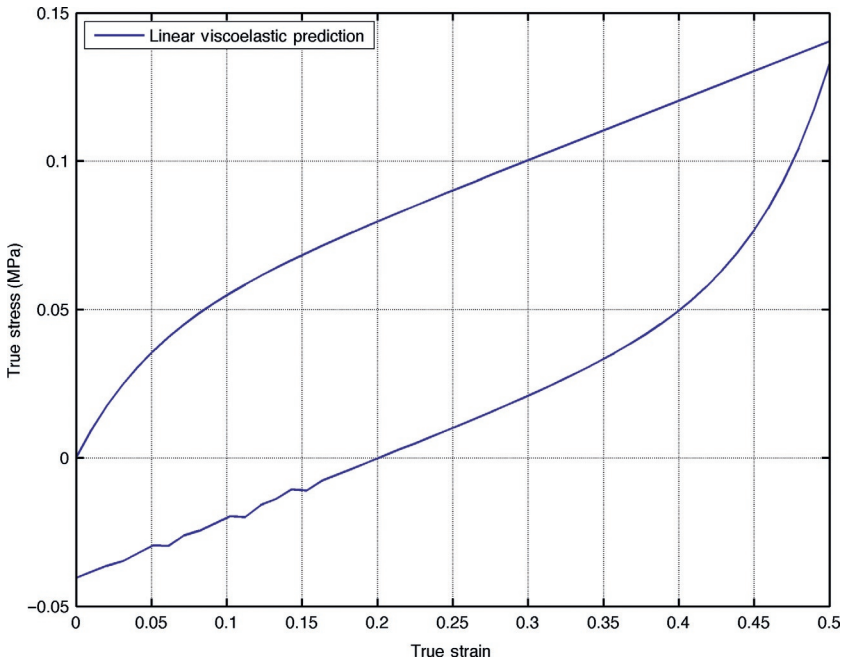
```
function stressVec = mat_LVE(timeVec, strainVec, params)
%mat_LVE Linear viscoelastic model
%Uniaxial loading
for i = 1 : length(timeVec)
    tmax = timeVec(i);
    F = @(t) integrand(t, tmax, timeVec, strainVec, params);
    stressVec(i) = quad(F, eps, tmax);
end
end

function res = integrand(tau, tmax, timeVec, strainVec, params)
E0 = params(1); % ABAQUS-style Prony series
g = params(2:2:end);
tauG = params(3:2:end);
for i = 1 : length(tau)
    ER = E0 * (1 - sum( g .* (1 - exp(-(tmax-tau(i))./tauG)) ));
    edot = (interp1(timeVec, strainVec, tau(i) + 1e-4, ...
        'linear', strainVec(end)) - ...
        interp1(timeVec, strainVec, tau(i))) / 1e-4;
    res(i) = ER * edot;
end
end
```

Matlab File Name: "test\_mat\_LVE.m"

```
time = linspace(0,2)';
strain = [linspace(0, 0.5,50)'; linspace(0.5, 0, 50)'];
stress = mat_LVE(time, strain, [1.0 0.8 0.1]);
```

The predictions from running the `test_mat_LVE` function are shown in [Figure 6.9](#). This figure shows that the model predicts significant hysteresis during cyclic loading.



**Figure 6.9** Example prediction of the linear viscoelasticity model. The prediction was created using the Matlab file `test_mat_LVE.m`.

## Python Implementation

A linear elastic model with linear viscoelasticity model can also readily be implemented in Python. Similar to the Matlab example from the previous section, this implementation aims to illustrate a simple practical algorithm, and not focus on numerical efficiency.

```

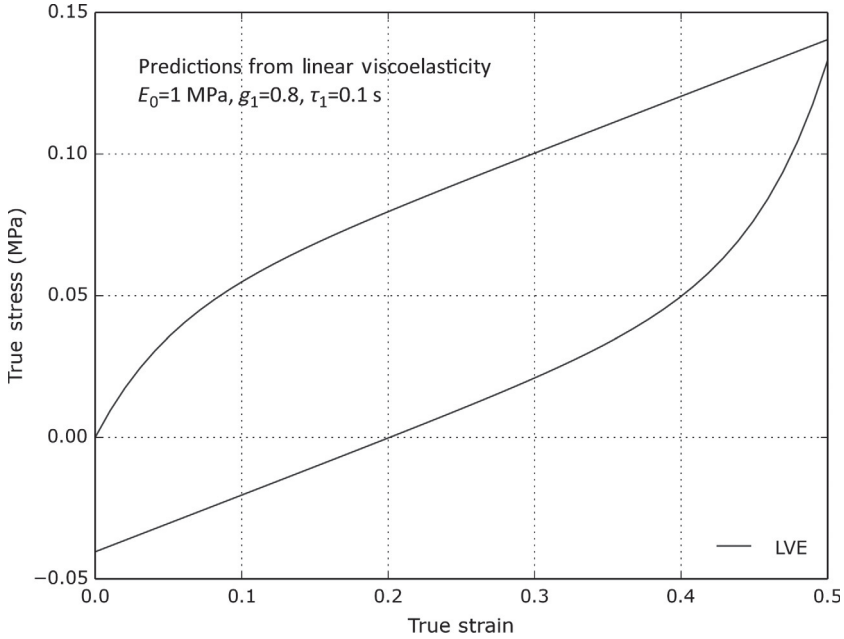
Python File Name: "LVE_small_strain.py"
from pylab import *
from scipy.integrate import quad
from scipy.interpolate import interp1d

def integrand(tau, tmax, timeVec, strainVec, params):
    E0 = params[0]
    g = params[1::2]
    tauG = params[2::2]
    ER = E0 * (1.0 - sum(g * (1 - exp((-tmax+tau)/tauG) )))
    f = interp1d(timeVec, strainVec)
    edot = (f(tau+1e-5) - f(tau)) / 1e-5
    return ER * edot

def mat_LVE(timeVec, strainVec, params):
    """Linear viscoelasticity. Uniaxial loading."""
    stressVec = zeros(len(timeVec))
    for i in arange(len(timeVec)):
        tmax = timeVec[i]
        stressVec[i] = quad(integrand, 0, tmax, \
            args=(tmax, timeVec, strainVec, params))[0]
    return stressVec

time = linspace(0, 2, 100)
strain = concatenate((linspace(0,0.5), linspace(0.5,0)))
stress = mat_LVE(time, strain, [1.0, 0.8, 0.1])
plot(strain, stress, label='LVE')
show()

```



**Figure 6.10** Example prediction of the linear viscoelasticity model. The prediction was created using the Python file `LVE_small_strain.py`.

The predictions from running the file `LVE_small_strain.py` are shown in [Figure 6.10](#).

### 6.3 Large Strain Linear Viscoelasticity

Linear viscoelasticity is often used in large strain predictions. In order to perform these calculations it is recommended to generalize the previous derivation of linear viscoelasticity to use hyperelasticity instead of linear elasticity as the instantaneous and long-term response.

One way to illustrate this generalization (in a non-rigorous way) is to start with the basic linear viscoelasticity equation:

$$\sigma(t) = \int_{-\infty}^t E_R(t - \tau) d\varepsilon(t) = \int_{-\infty}^t E_R(t - \tau) \frac{d\varepsilon(\tau)}{d\tau} d\tau. \quad (6.5\text{-repeat})$$

This equation can be integrated in parts to:

$$\sigma(t) = E_0 \varepsilon(t) - \int_0^t \dot{g}_R(t - \tau) E_0 \varepsilon(\tau) d\tau, \quad (6.68)$$

where the normalized relaxation modulus  $g_R(t) = E_R(t)/E_0$  has been introduced.

We can now introduce the hyperelastic stress function:

$$\sigma_{\text{hyp}}(\varepsilon) = \sigma_{\text{hyp}}(\varepsilon(t)) \equiv E_0 \varepsilon(t), \quad (6.69)$$

which when inserted into (6.5) becomes:

$$\sigma(t) = \sigma_{\text{hyp}}(\varepsilon(t)) - \int_0^t \dot{g}_R(t - \tau) \sigma_{\text{hyp}}(\varepsilon(\tau)) d\tau. \quad (6.70)$$

This equation expresses how a stress relaxation function (expressed as a Prony series) and a hyperelastic stress function can be combined to give a complete material model capable of predicting the stress response caused by any arbitrary large uniaxial deformation strain history.

The large-strain linear viscoelasticity model is available in all major finite element programs.

### 6.3.1 Numerical Implementation

Consider a case of uniaxial incompressible loading. In this case the stress response at time  $t$  is given by Equation (6.5), which can also be written:

$$\sigma(t) = \sigma_{\text{hyp}}(t) - \sigma_{\text{visc}}(t), \quad (6.71)$$

where  $\sigma_{\text{visc}}(t)$  is the viscoelastic stress at time  $t$ . Similarly, the stress at  $t + \Delta t$  is given by:

$$\sigma(t + \Delta t) = \sigma_{\text{hyp}}(t + \Delta t) - \int_0^{t+\Delta t} \dot{g}_R(t + \Delta t - \tau) \sigma_{\text{hyp}}(\tau) d\tau \quad (6.72)$$

$$\equiv \sigma_{\text{hyp}}(t + \Delta t) - \sum_{i=1}^N \sigma_{\text{visc}}^i(t + \Delta t), \quad (6.73)$$

where  $\sigma_{\text{visc}}^i$  is the viscoelastic stress due to Prony series term  $i$ , which is given by:

$$\sigma_{\text{visc}}^i(t + \Delta t) = \int_0^{t+\Delta t} \frac{d}{d\tau} \left[ (1 - g_i) + g_i e^{-(t+\Delta t-\tau)/\hat{\tau}_i} \right] \sigma_{\text{hyp}}(\tau) d\tau \quad (6.74)$$

$$= \int_0^t \frac{d}{d\tau} \left[ (1 - g_i) + g_i e^{-(t+\Delta t-\tau)/\hat{\tau}_i} \right] \sigma_{\text{hyp}}(\tau) d\tau \quad (6.75)$$

$$+ \int_t^{t+\Delta t} \frac{d}{d\tau} \left[ (1 - g_i) + g_i e^{-(t+\Delta t-\tau)/\hat{\tau}_i} \right] \sigma_{\text{hyp}}(\tau) d\tau \quad (6.76)$$

$$\equiv A_i + B_i. \quad (6.77)$$

The first integral  $A_i$  is given by:

$$A_i = \int_0^t \frac{g_i}{\hat{\tau}_i} \left[ e^{-t/\hat{\tau}_i} \cdot e^{-\Delta t/\hat{\tau}_i} \cdot e^{\tau/\hat{\tau}_i} \right] \sigma_{\text{hyp}}(\tau) d\tau \quad (6.78)$$

$$= e^{-\Delta t/\hat{\tau}_i} \cdot \sigma_{\text{visc}}^i(t). \quad (6.79)$$

To evaluate the second integral  $B_i$ , assume  $\sigma_{\text{hyp}}(\tau)$  varies linearly with  $\tau$  in the interval  $[t, t + \Delta t]$ :

$$\sigma_{\text{hyp}}(\tau) = \sigma_{\text{hyp}}(t) + \Delta\sigma_{\text{hyp}} \cdot \frac{\tau - t}{\Delta t}, \quad (6.80)$$

where

$$\Delta\sigma_{\text{hyp}} = \sigma_{\text{hyp}}(\varepsilon(t + \Delta t)) - \sigma_{\text{hyp}}(\varepsilon(t)). \quad (6.81)$$

With this approximation the integral  $B_i$  can be solved in closed form. The algebra is here left as an exercise, the results become:

$$B_i = g_i \sigma_{\text{hyp}}(t) \left[ 1 - e^{-\Delta t/\hat{\tau}_i} \right] + g_i \frac{\Delta\sigma_{\text{hyp}}}{\Delta t} \left[ (\Delta t - \hat{\tau}_i) + \hat{\tau}_i e^{-\Delta t/\hat{\tau}_i} \right]. \quad (6.82)$$

To summarize, the stress at  $t + \Delta t$  can be calculated from:

$$\sigma(t + \Delta t) = \sigma_{\text{hyp}}(t + \Delta t) - \sum_{i=1}^N \sigma_{\text{visc}}^i(t + \Delta t), \quad (6.83)$$



where

$$\sigma_{\text{visc}}^i(t + \Delta t) = e^{-\Delta t/\hat{\tau}_i} \cdot \sigma_{\text{visc}}^i(t) + g_i \sigma_{\text{hyp}}(t) \left[ 1 - e^{-\Delta t/\hat{\tau}_i} \right] + g_i \frac{\Delta \sigma_{\text{hyp}}}{\Delta t} \left[ (\Delta t - \hat{\tau}_i) + \hat{\tau}_i e^{-\Delta t/\hat{\tau}_i} \right]. \quad (6.84)$$

In this equation,  $\sigma_{\text{visc}}^i$  are internal state variables for each Prony series term. The large strain linear viscoelasticity model shown here can easily be implemented in Python, as shown below. The implementation shown here is using a Neo-Hookean hyperelastic model with a 1-term Prony series, and the implementation is numerically very efficient. A numerically efficient implementation for compressible, multiaxial deformations can also be implemented [5], but is outside the scope of this chapter.

Python File Name: "LVE\_large\_strain.py"

```
from pylab import *

def NH(strain, params):
    """Neo-Hookean model. Incompressible uniaxial loading."""
    mu = params[0]
    lam = exp(strain)
    return mu * (lam*lam - 1/lam)

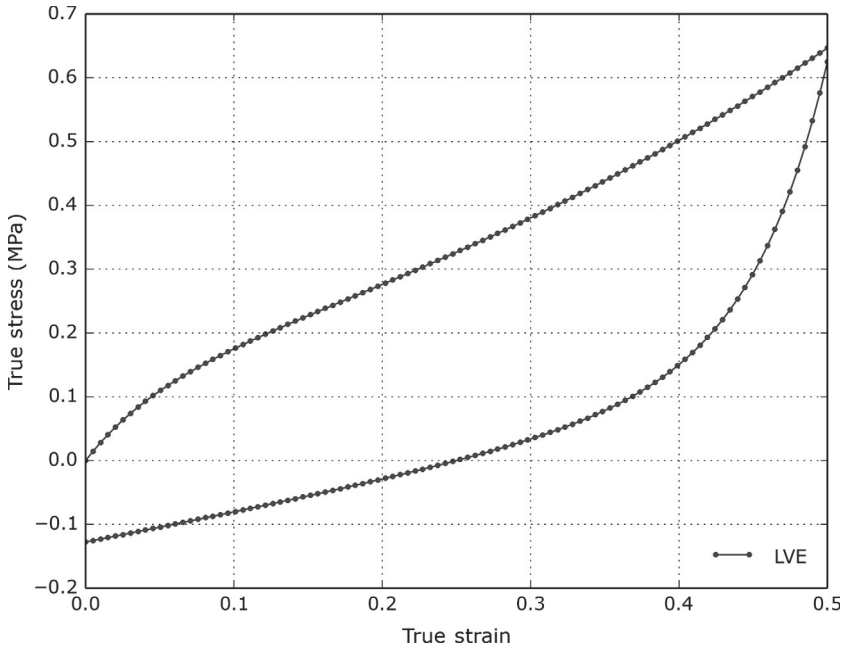
def LVE_uniax(time, strain, params):
    """Linear viscoelasticity. [mu, g1, tau1, g2, tau2, ...]"""
    stress = zeros(len(time))
    g = params[1::2]
    tau = params[2::2]
    stressV = zeros(len(g))
    stressH0 = NH(strain[0], params)
    for i in range(1, len(time)):
        stressH1 = NH(strain[i], params)
        dstressH = stressH1 - stressH0
        dt = time[i] - time[i-1]
        stress[i] = stressH1
        for j in range(len(g)):
            stressV[j] = exp(-dt/tau[j]) * stressV[j] + \
                g[j]*stressH0*(1 - exp(-dt/tau[j])) + \
                g[j]*dstressH/dt*(dt-tau[j]+tau[j]*exp(-dt/tau[j]))
            stress[i] = stress[i] - stressV[j]
        stressH0 = stressH1
    return stress

N = 100
time = linspace(0, 2, 2*N)
strain = concatenate((linspace(0,0.5,N), linspace(0.5,0,N)))

params = [1.0, 0.8, 0.1] # [mu, g, tau]
stress = LVE_uniax(time, strain, params)

plot(strain, stress, 'b.-', label='LVE')
show()
```

The predictions from running this file are shown in [Figure 6.11](#).



**Figure 6.11** Linear viscoelastic predictions using the Python file `LVE_large_strain.py`. Material parameters:  $\mu = 1$  MPa,  $g_1 = 0.8$ , and  $\tau_1 = 0.1$  s.

## 6.4 Creep Compliance Behavior

The linear viscoelasticity theory presented in this chapter has been formulated in terms of strains as the driving quantity. It is also possible to formulate the linear viscoelasticity theory for the case when the stress is the driving quantity.

In this case the creep compliance, defined by  $J(t) \equiv \varepsilon(t)/\sigma$ , is the key quantity. And the resulting strain from an applied stress history can be calculated from:

$$\varepsilon(t) = \int_{-\infty}^t J(t - \tau) \frac{d\sigma(\tau)}{d\tau} d\tau. \quad (6.85)$$

Since most major FE programs are formulated in displacement form, with strain as the driving quantity, this alternative formulation of linear viscoelasticity in terms of stresses is only briefly discussed in the following, and is discussed in more detail in various references [1–3].

### 6.4.1 Relationships Between Creep Compliance and Relaxation Modulus

The formal relationship between the relaxation modulus  $G(t)$  and the creep compliance  $J(t)$  can be derived by taking the Laplace transformation:

$$\mathcal{L}\{f(t)\} = \int_0^{\infty} e^{-st} f(t) dt, \quad (6.86)$$

of Equations (6.5) and (6.85) giving:

$$\tilde{\sigma}(s) = s\tilde{G}(s)\tilde{\varepsilon}(s), \quad (6.87)$$

$$\tilde{\varepsilon}(s) = s\tilde{J}(s)\tilde{\sigma}(s). \quad (6.88)$$

This equation can be rewritten

$$\tilde{G}(s) = \frac{1}{s^2\tilde{J}(s)}. \quad (6.89)$$

Based on the Laplace limit theorems it can be shown [1] that:

$$\lim_{t \rightarrow 0} J(t) = \lim_{t \rightarrow 0} \frac{1}{G(t)}, \quad (6.90)$$

$$\lim_{t \rightarrow \infty} J(t) = \lim_{t \rightarrow \infty} \frac{1}{G(t)}. \quad (6.91)$$

Also since  $\mathcal{L}^{-1}(1/s^2) = t$  we directly get:

$$\int_0^t G(t-\tau)J(\tau) d\tau = \int_0^t J(t-\tau)G(\tau) d\tau = t. \quad (6.92)$$

These functional relationships can be used when converting a stress relaxation function to a creep compliance function, or when converting a creep compliance function to a stress relaxation function. This type of conversion can also be achieved through direct numerical optimization of, for example, the Prony series. This may be of interest if the creep compliance is known and the corresponding Prony series is required for a finite element analysis.

One way to perform this conversion is to:

1. Select an initial guess of the Prony series parameters.
2. Calculate the creep compliance through a direct simulation

3. Use an optimization algorithm (e.g. the Nelder-Mead simplex algorithm) to modify the current value of the Prony series parameters.
4. If the predicted creep compliance is in good agreement with the experimental creep compliance stop, otherwise go to (2).

This algorithm is implemented in specialized software, e.g. MCalibration [9], and can be effectively used to determine the Prony series parameters from creep data.

## 6.5 Differential Form of Linear Viscoelasticity

So far the presentation of linear viscoelasticity has used an integral equation formulation:

$$\sigma(t) = \int_{-\infty}^t E(t - \tau) \frac{d\varepsilon(\tau)}{d\tau} d\tau. \quad (6.5\text{-repeat})$$

This equation can also be written in differential form as

$$P(D)\sigma(t) = Q(D)\varepsilon(t), \quad (6.93)$$

where  $P(D)$  and  $Q(D)$  are polynomials of the differential operator  $D \equiv d/dt$ . To show this start by taking the Laplace transform of Equation (6.93):

$$\begin{aligned} P(s)\tilde{\sigma}(s) - \frac{1}{s} \sum_{k=1}^N p_k \sum_{r=1}^k s^r \sigma^{(k-r)}(0) \\ = Q(s)\tilde{\varepsilon}(s) - \sum_{k=1}^N q_k \sum_{r=1}^N s^r \varepsilon^{k-r}(0). \end{aligned} \quad (6.94)$$

The Laplace transform of (6.5) is

$$\tilde{\sigma}(s) = s\tilde{G}(s)\tilde{\varepsilon}(s), \quad (6.87\text{-rep})$$

illustrating that the two forms are equal if

$$s\tilde{G}(s) = \frac{\tilde{Q}(s)}{\tilde{P}(s)} \quad (6.95)$$

and

$$\sum_{r=k}^N p_r \sigma_{ij}^{r-k}(0) = \sum_{r=k}^N q_r \varepsilon^{r-k}(0), \quad k = 1, 2, \dots, N, \quad (6.96)$$

which is a restriction on the initial conditions. The integral equation formulation and the differential equation formulation are therefore functionally equivalent.

### 6.5.1 Rheological Models

One interesting way to visualize and develop differential forms of a linear viscoelastic model is to start with a one-dimensional rheological representation. This representation uses springs and dashpots to visualize the material model structure. Rheological representations can also be used to convert a linear viscoelastic model to a *non-linear* viscoelastic model. This will be discussed in detail in Chapter 8.

To initiate the discussion about rheological models we will first study two common model structures.

#### **Maxwell Model**

The Maxwell rheological model consists of a linear spring ( $\sigma = E\varepsilon_1$ ) and a linear dashpot ( $\dot{\varepsilon}_2 = \sigma/\eta$ ) in series, see [Figure 6.12](#). Here  $E$  is the Young's modulus, and  $\eta$  is the viscosity.

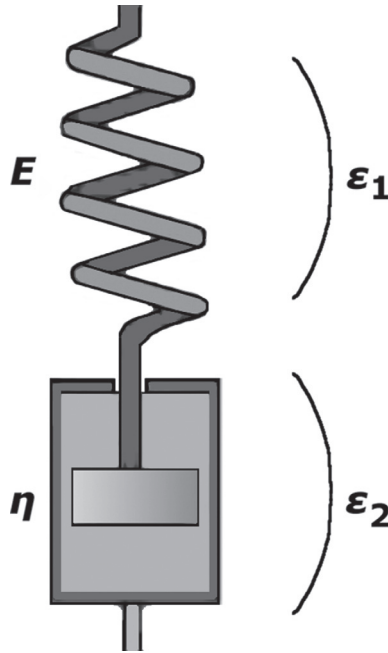
The rate of change in strain of the system is given by the equilibrium equation

$$\dot{\varepsilon} = \frac{d}{dt} (\varepsilon_1 + \varepsilon_2) = \frac{\dot{\sigma}}{E} + \frac{\sigma}{\eta}, \quad (6.97)$$

which is a differential equation representation of the Maxwell model.

Now consider a stress relaxation experiment with an applied strain step:  $\varepsilon(t) = \varepsilon_0 H(t)$ . Inserting this into Equation (6.97) gives

$$\frac{d\sigma}{dt} + \frac{E}{\eta} \sigma = 0 \quad (\text{for } t > 0).$$



**Figure 6.12** Rheological representation of the Maxwell model.

This differential equation can be solved in closed form

$$\sigma(t) = \sigma_0 \exp\left[\frac{-t}{\eta/E}\right]. \quad (6.98)$$

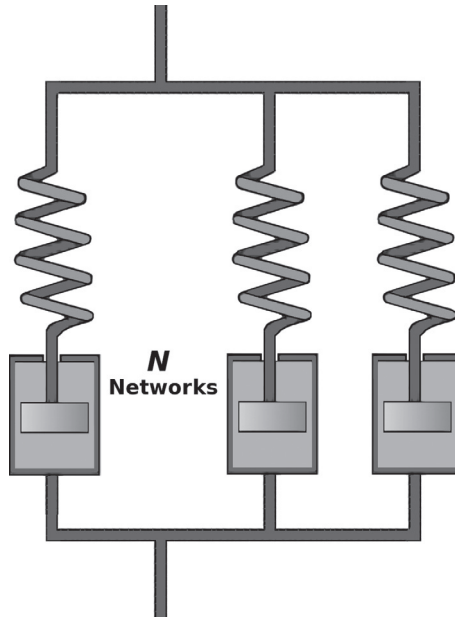
Hence, the stress relaxation function for the Maxwell model is given by an exponentially decaying stress relaxation modulus

$$E(t) = E_0 \exp\left[\frac{-t}{\eta/E}\right],$$

which is a single Prony series term. Or equivalently, a single Prony series term is equivalent to a Maxwell rheological model.

### **Multi-Network Maxwell Model**

A multi-network Maxwell model consists of a number of parallel Maxwell models, see [Figure 6.13](#).



**Figure 6.13** A multi-network Maxwell model is equivalent to a Prony series.

From the results in the previous section it is clear that the effective stress relaxation modulus for this multi-network model is

$$E_R(t) = E_0 + \sum_{i=1}^N E_i e^{-t/\tau_i}, \quad (6.99)$$

which is equivalent to the Prony series in Equation (6.14). Hence:

*A linear viscoelastic model with a given Prony series is identical to a multi-network Maxwell model.*

This equivalence can be of use when formulating strategies for solving the constitutive equations, and when developing extensions to linear viscoelasticity.

## 6.6 The Use of Shift Functions to Generalize Linear Viscoelasticity Theory

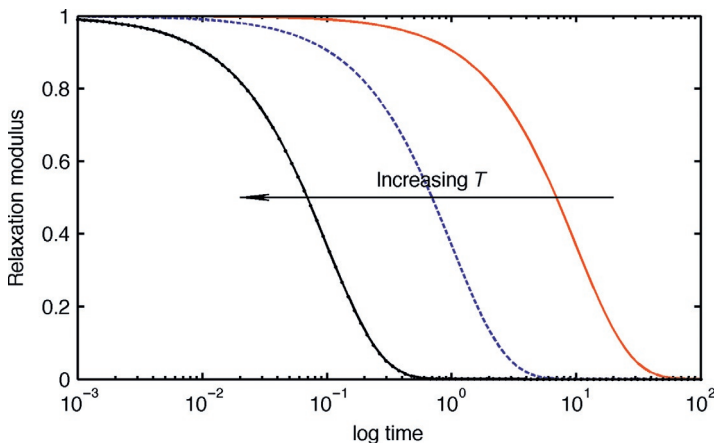
For many polymeric materials the domain in which linear viscoelasticity theory gives good predictions is unfortunately

relatively small. Based on experimental observations it turns out that the influence of variations in external parameters (such as temperature and aging time), which contribute to the limited applicability of the theory, can be accounted for by using a shift function approach. The reason for the success of this simple idea is that when the material functional is plotted using appropriate log-scales, changes in these external parameters do not change the shape of the plotted curve to any significant degree, only shift it.

### 6.6.1 Time-Temperature Equivalence

The discussion has so far been for a general viscoelastic material and can therefore be applied also to a polymer. But when considering polymers a number of complications become apparent. One of the more important issues that need to be recognized is the strong temperature dependence of polymeric materials. It has been shown experimentally that in many cases the temperature dependence can be modeled by a scaling of time using what has been termed the *time-temperature equivalence*.

The basis for this principle is shown in [Figure 6.14](#) illustrating that if the experimentally determined stress relaxation modulus is plotted as a function of logarithmic time, the shape of the resulting curves is the same for a wide interval of temperatures.



**Figure 6.14** Dependence of stress relaxation modulus on time and temperature.



In fact, the only significant difference between the curves is a horizontal shift. This observation suggests that if the relaxation modulus is known at one temperature (i.e. the *master curve* is known) then the relaxation modulus at any other temperature can be obtained if the horizontal shift factor  $a_T = a_T(T)$  is known. A material that has this property is called a *rheologically simple material*.

The time shifts can be defined by

$$\log t_{T_0} - \log t_T = \log a_T, \quad (6.100)$$

where  $t_T$  is the time at temperature  $T$ , and  $t_{T_0}$  the time at temperature  $T_0$ . Equation (6.100) gives  $a_T = t_{T_0}/t_T$ , or  $t_T = t_{T_0}/a_T$ , hence the behavior at a temperature of  $T$  becomes exactly the same as the behavior at the reference temperature  $T_0$  if the time is *accelerated* by the factor  $a_T$ .

In general, the temperature can be a function of time  $T(t_{T_0})$ , so instead of  $t_T = t_{T_0}/a_T$  it is necessary to write

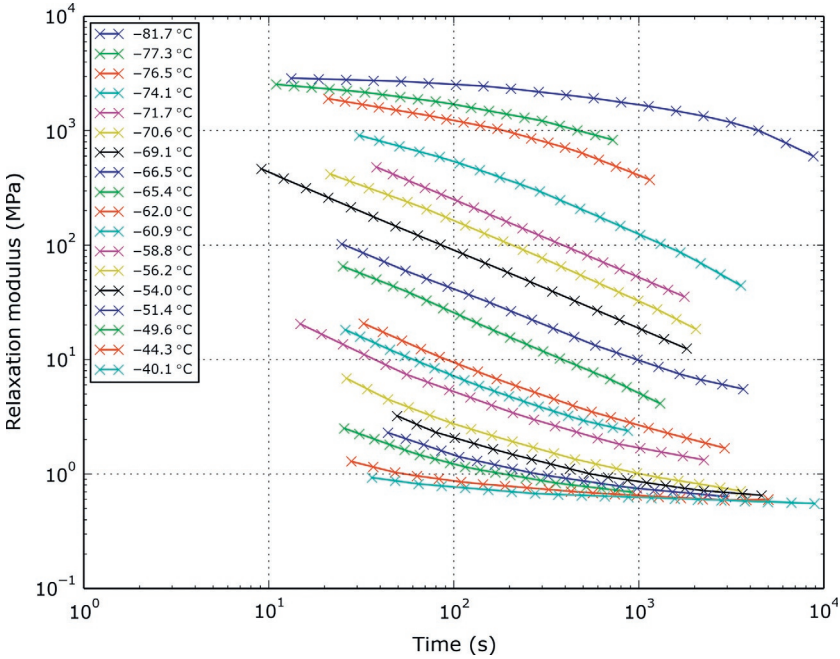
$$dt_T = \frac{dt_{T_0}}{a_T(T(t_{T_0}))} \quad (6.101)$$

giving

$$t_T = \int_0^{t_{T_0}} \frac{dt'_{T_0}}{a_T(T(t'_{T_0}))}. \quad (6.102)$$

The effective time experienced by the material, the material time, is a function of temperature and wall clock time. For a rheologically simple material, the scaling of time with temperature occurs in all viscoelastic quantities such as  $G$ ,  $J$ ,  $J'$ ,  $J''$ ,  $\tan \delta$ , etc. And the scaling constant must be the same for all quantities for the material to be rheologically simple.

Another common way to use time-temperature equivalence is to use experimental data obtained at different temperatures to estimate the stress-relaxation behavior at longer times than what was experimentally tested. As an example, consider the experimental data for butyl rubber [10] shown in Figure 6.15. The data shows the stress relaxation behavior for times between 10 s and 3 hours and for temperatures between  $-81.7^\circ\text{C}$  and



**Figure 6.15** Experimental stress relaxation data at different temperatures.

$-40.1\text{ }^{\circ}\text{C}$ . As shown in the figure the relaxation modulus is very temperature sensitive under these conditions.<sup>5</sup>

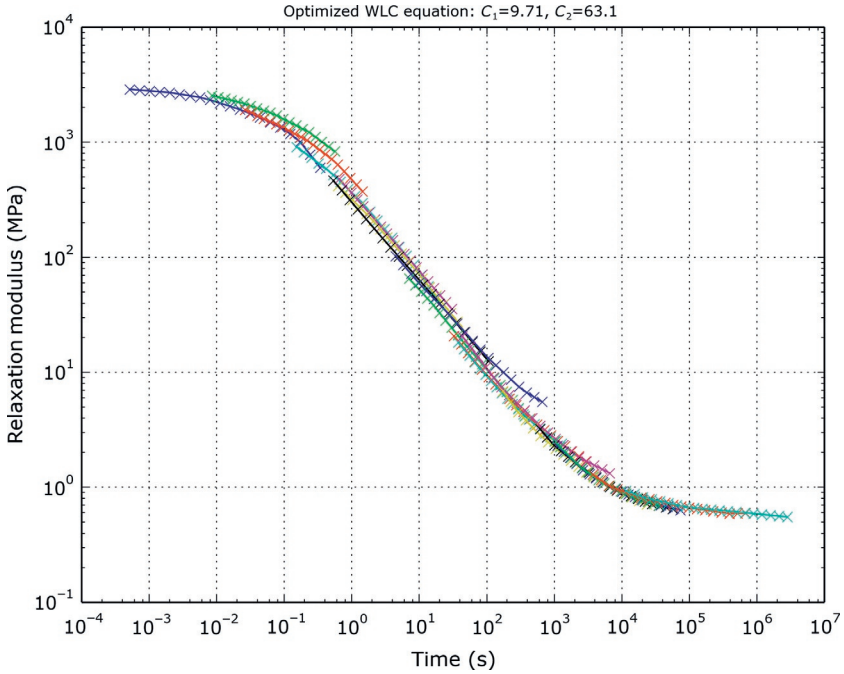
The stress relaxation behavior can be horizontally shifted using the William-Landel-Ferry (WLF) equation using the parameters:  $C_1 = 9.71$ ,  $C_2 = 63.1\text{ K}$ , and  $T_0 = -62\text{ }^{\circ}\text{C}$ , see Figure 6.16. By applying the time-temperature equivalence the experimental data now covers stress relaxation times from  $10^{-4}\text{ s}$  to  $10^6\text{ s}$ .

### Example

To exemplify this idea consider a simple Maxwell element:  $G(t) = G_0 \exp(-t)$ . If the material is rheologically simple the relaxation modulus becomes

$$G(t, T) = G_0 \exp[-a_T(T)t].$$

<sup>5</sup>The glass transition temperature is approximately  $-62\text{ }^{\circ}\text{C}$  for butyl rubber.



**Figure 6.16** Experimental stress relaxation data after horizontal shifting using the WLF equation.

The influence of  $a_T(T)$  on the stress-strain behavior was examined in [Section 6.2.1](#).

One commonly used representation of the shift factor  $a_T$  is the WLF equation [[11](#)]:

$$\log a_T(T) = \frac{C_1(T - T_0)}{C_2 + T - T_0}, \quad (6.103)$$

where  $C_1 = 17.4$  and  $C_2 = 51.6$  K. This relationship is often used [[2](#)] for amorphous polymers in the temperature range  $T \in [T_g - 50 \text{ K}, T_g + 50 \text{ K}]$ .

Another common equation for the shift factor  $a_T$  is the Arrhenius model:

$$\ln a_T(T) = -\frac{E_0}{R} \left( \frac{1}{T_0} - \frac{1}{T} \right), \quad (6.104)$$

where  $E_0$  is the apparent activation energy for viscoelastic relaxation,  $R \approx 8.314 \text{ J}/(\text{mol K})$  is the gas constant,  $T_0$  is a reference temperature, and  $T$  is the current temperature.

### 6.6.2 Vertical Shifts

As will be discussed in more detail below, the stress relaxation modulus (and also the creep compliance) curves when plotted as a function of logarithmic time often turn out to have the same shape not only for different temperatures but also for variations in other parameters (such as aging time). But to create a master curve in these cases it is often necessary to also use vertical shifts (on a log-scale):

$$\log G^\alpha - \log G^{\alpha_0} = \log b \quad (6.105)$$

giving

$$b = \frac{G^\alpha}{G^{\alpha_0}}. \quad (6.106)$$

In summary, the integral formulation

$$\sigma(t) = \int_{-\infty}^t G(t - \tau) \frac{d\varepsilon(\tau)}{d\tau} d\tau \quad (6.5\text{-rep})$$

becomes

$$\sigma(t) = \int_0^t b(t) G(\hat{t}(t) - \hat{t}(\tau)) \frac{d\varepsilon(\tau)}{d\tau} d\tau \quad (6.107)$$

when both vertical shift  $b(\theta_1(t))$  and horizontal shift  $\hat{t}(t) = \hat{t}(\theta_2(t))$ , where  $\theta_1$  and  $\theta_2$  are external parameters such as temperature, are considered.

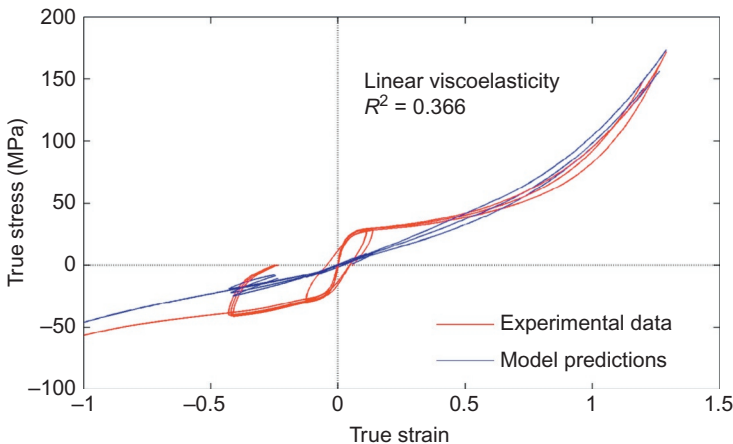
## 6.7 Use of Linear Viscoelasticity in Polymer Modeling

In this chapter the key aspects of linear viscoelasticity theory have been presented. It has been shown that to characterize the material only the instantaneous elastic (or the long-term elastic)

response and the Prony series relaxation modulus<sup>6</sup> function need to be determined. The Prony series terms can be determined through one creep, stress relaxation or oscillatory experiment. Once the material dependent Prony series has been determined the response due to any arbitrary imposed deformation or loading can be directly calculated.

Linear viscoelasticity is mainly used for predicting the response of elastomer-like materials, and to a lesser degree, the very small strain response of thermoplastic materials. Since linear viscoelasticity cannot predict plastic or viscoplastic deformations, it should not be used to predict the response of glassy polymers for strains that are similar in magnitude to the yield strain, or larger.

One example of what can happen if linear viscoelasticity is attempted to be used for ultra-high molecular weight polyethylene (UHMWPE) is shown in Figure 6.17. This figure shows that linear viscoelasticity is completely unable to predict the response of thermoplastics beyond yielding.

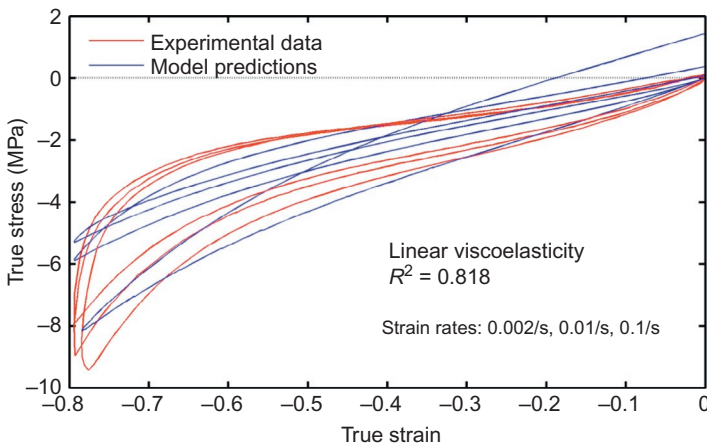


**Figure 6.17** Application of LVE to the large strain behavior of UHMWPE.

<sup>6</sup>Only one material relaxation function is required for an incompressible isotropic material subjected to a one-dimensional loading situation. For an isotropic material subjected to a general loading situation two material functionals need to be determined, and for an anisotropic material the number of functionals are dependent on the material symmetry.

Linear viscoelasticity is also not always able to accurately predict the viscoelastic response of elastomers. Figure 6.18 shows experimental data for a chloroprene rubber tested in uniaxial compression, followed by unloading. The experiment was repeated at three different strain rates. The figure also shows the best predictions from linear viscoelasticity. In this case linear viscoelasticity does not accurately predict the material response. The reason for the poor predictions of this material behavior is that the material is not *linearly* viscoelastic. The strains in this case are large enough that the material response is non-linear viscoelastic. Better material models for predicting this data set is discussed in Section 8.2 of Chapter 8.

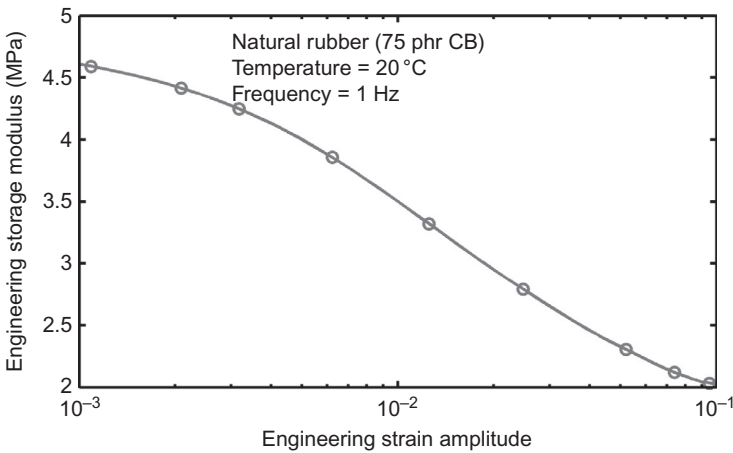
The strain magnitude below which linear viscoelasticity is an effective tool can be experimentally determined by running dynamic mechanical analysis (DMA) tests where the strain amplitude is gradually ramped from small to large values. Due to the Boltzmann superposition principle, the predictions from linear viscoelasticity will be independent of the strain amplitude. So, the results from this type of DMA tests can be used to determine the strain level below which the material response is *linearly* viscoelastic.



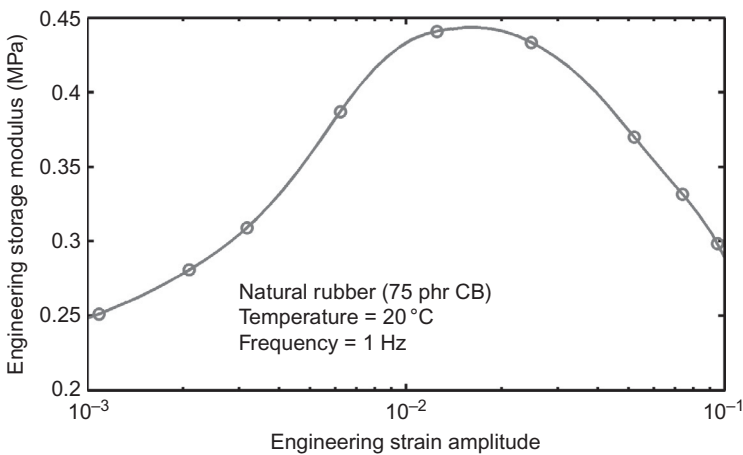
**Figure 6.18** Application of LVE to the large strain behavior of elastomers.

Dynamic storage and loss modulus data for natural rubber and silicone rubber as a function of strain amplitude are shown in Figures 6.19–6.21. These figures show that response of these materials is only linear viscoelastic at very small applied strains (less than 1%).

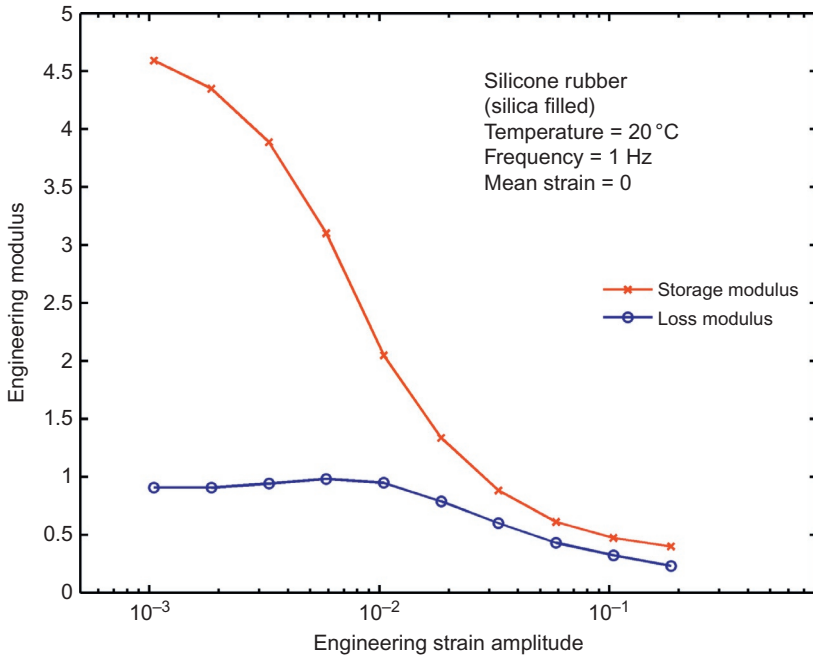
The storage modulus as a function of strain amplitude, for many elastomers, is a decreasing function of the strain amplitude.



**Figure 6.19** Experimentally determined storage modulus as a function of strain amplitude for a natural rubber [12].



**Figure 6.20** Experimentally determined loss modulus as a function of strain amplitude for a natural rubber [12].



**Figure 6.21** Experimentally determined loss modulus as a function of strain amplitude for a silicone rubber [12].

This effect, which is called the *Payne effect*, cannot be captured using linear viscoelasticity, but can be captured using non-linear viscoelasticity. To capture the Payne effect requires non-linear viscoelastic, a topic that will be discussed in Chapter 8.

## 6.8 Exercises

1. Describe the Boltzmann superposition principle.
2. Explain why linear viscoelasticity in uniaxial loading only requires an elastic modulus and the stress relaxation behavior.
3. How many Prony series terms should be experimentally determined before using linear viscoelasticity? What controls the required number of Prony series terms?
4. Which of the following sets of experimental data are *sufficient* for calibrating a linear viscoelastic model?



- stress relaxation from a given strain level
  - creep from a given stress level
  - monotonic uniaxial tension at a given strain rate
  - monotonic uniaxial tension at two different strain rates
  - a DMA temperature sweep
  - a DMA frequency sweep
5. Which of the following sets of experimental data are *recommended* for calibrating a linear viscoelastic model?
- stress relaxation from a given strain level
  - creep from a given stress level
  - monotonic uniaxial tension at a given strain rate
  - monotonic uniaxial tension at two different strain rates
  - a DMA temperature sweep
  - a DMA frequency sweep
6. Why is the Prony series written as a sum of exponentially decaying terms?
7. What is the definition of viscoelastic spectra?
8. Draw a schematic rheological model for a linear viscoelastic model with a 2-term Prony series.
9. What is a rheologically simple material?
10. What is the definition of the WLF equation?
11. What experiments would you run to determine if the response of a material is linear viscoelastic?
12. What is the graphical representation of a Maxwell element?
13. Describe the time-temperature superposition principle.

## References

- [1] R.M. Christensen. *Theory of Viscoelasticity; An Introduction*. Academic Press, New York, 1982.
- [2] M.T. Shaw, W.J. MacKnight. *Introduction to Polymer Viscoelasticity*. Wiley-Interscience, New York, 2005.
- [3] D.I. Bower. *An Introduction to Polymer Physics*. Cambridge University Press, Cambridge, 2002.
- [4] L. Boltzmann. *Pogg. Ann. Phys. U. Chem.*, 7 1876 624.

- [5] HKS, Inc.. ABAQUS, Pawtucket, RI, ver. 6.14, 2014.
- [6] F. Kohlrausch. Pogg. Ann. Phys., 12 1847 393.
- [7] G. Williams, D.C. Watts. Trans. Faraday Soc., 66 1970 80.
- [8] Private communications with Ted Diehl, DuPont., 2013.
- [9] Veryst Engineering. <http://PolyUMod.com>.
- [10] E. Castiff, A.V. Tobolsky. Stress-relaxation of polyisobutylene in the transition region. J. Colloid Sci., 10 1955 395-392.
- [11] M.L. Williams, R.F. Landel, J.D. Ferry. J. Am. Chem. Soc., 77 1955 3701.
- [12] L. Chazeau, J.D. Brown, L.C. Yanyo, S.S. Sternstein. Modulus recovery kinetics and other insights into the Payne effect for filled elastomers. Polym. Compos., 21 (2) 2000 202-222.

# 7 Plasticity Models

---

## Chapter Outline

7.1 Introduction	353
7.2 $J_2$ -Plasticity with Isotropic Hardening	354
7.2.1 Cyclic Loading	355
7.2.2 Matlab Implementation	357
7.2.3 Python Implementation	359
7.2.4 Application to Thermoplastics	361
7.3 Plasticity with Kinematic Hardening	362
7.4 Johnson-Cook Plasticity	365
7.5 Drucker Prager Plasticity	366
7.6 Use of Plasticity Models in Polymer Modeling	367
7.7 Exercises	368
References	369

## 7.1 Introduction

One historically common way to represent the mechanical behavior of a thermoplastic material in a finite element analysis is to use a metal plasticity model. This type of material model is typically easy to use, and is available in all commercial finite element programs. The main problem with this class of material models is that they were developed for metals and can be inaccurate if used incorrectly for polymers. The purpose of this chapter is to list the most commonly used plasticity models and to highlight both their usefulness and their limitations.

The theory of metal plasticity is well developed and described in numerous textbooks [1–5]. This chapter gives a brief introduction to the theory and how it applies to different types of polymer materials.

## 7.2 $J_2$ -Plasticity with Isotropic Hardening

$J_2$ -plasticity with isotropic hardening is perhaps the most commonly used type of plasticity theory, and is often simply called *plasticity theory*. In Abaqus [6], this model is created using the following commands:

```
*Elastic
[E], [nu]
*Plastic
[sigY0], 0
[sigY1], [epsY1]
...
*Rate Dependent, type=power law
[D], [n]
```

In this example, some of the needed values are simply listed as variables within square brackets. Also, the yield stress was made dependent on the plastic strain rate (which is optional).

In ANSYS the model can be selected using the following commands: Also in this example some of the needed values are

```
! Isotropic hardening elastic-plastic model
! with Mises isotropic yield and rate dependence
MP, EX, [matid],[E]
MP, NUXY, [matid],[nu]

TB, MISO, [matid], 1, [N]
TBPT, [, [strain1],[stress1]
...

TB, RATE, [matid],1, 2, PERZYNA
TBDATA, 1, [m]
TBDATA, 2, [gamma]
```

simply listed as variables within square brackets, and the rate-dependence is optional.

The theory and implementation of this model has been extensively covered [3, 6], here a very simplified implementation will be presented. The theory will be presented for one-dimensional loading in order to demonstrate how the theory works, and also to illustrate strength and limitations to the model.

Start by dividing the total strain into elastic ( $\varepsilon^e$ ) and plastic ( $\varepsilon^p$ ) contributions:  $\varepsilon = \varepsilon^e + \varepsilon^p$ . Only the elastic strain contribute to the stress:  $\sigma = E(\varepsilon - \varepsilon^p)$ .

The yield condition can be written as a scalar function:

$$f(\sigma) = |\sigma| - \sigma_y(\alpha) \leq 0, \quad (7.1)$$

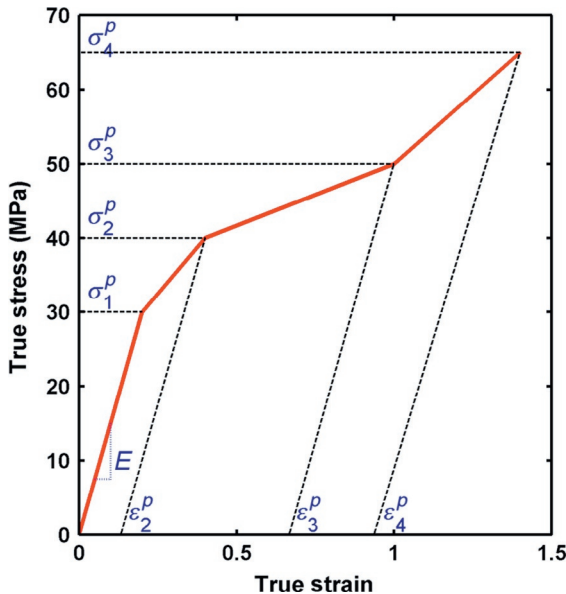
where  $\sigma_y(\alpha)$  is the yield stress at the accumulated plastic strain magnitude  $\alpha$ , which is given by  $\dot{\alpha} = |\dot{\varepsilon}^p|$ .

If  $f(\sigma) < 0$ , then the stress magnitude is less than the current yield stress and plastic flow is not active. If  $f(\sigma) = 0$ , then the stress magnitude is equal to the current yield stress and plastic flow can occur if the loading continues. Note that the yield function  $f(\sigma)$  cannot be positive since the stress magnitude cannot be greater than the current yield stress.

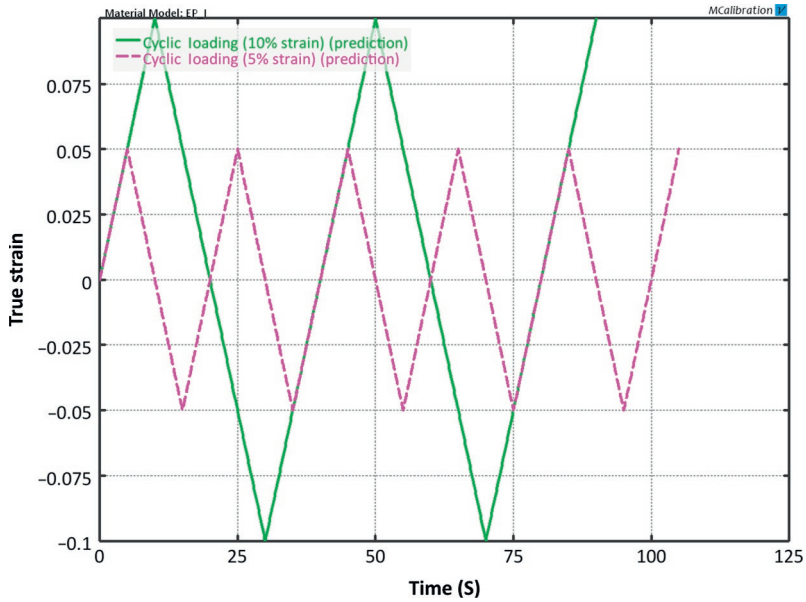
This material model can be graphically represented using a set of piecewise linear segments as shown in Figure 7.1. This means that the model can be made to fit almost any monotonic stress-strain data in a single loading mode. Note, however, that the model cannot be made to fit the large-strain behavior of elastomers since these materials often have a tangent modulus at large strains that is larger than the initial small-strain modulus.

## 7.2.1 Cyclic Loading

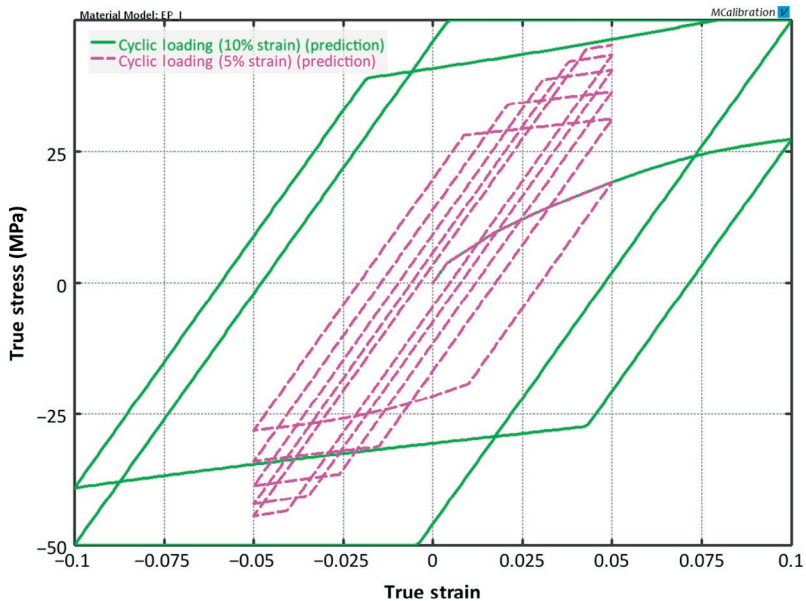
All isotropic hardening plasticity models that are based on (non-zero) isotropic hardening will exhibit a gradual increase in the yield stress with increasing plastic strain. As illustrated in Figures 7.2 and 7.3, this behavior will cause each stress-strain



**Figure 7.1** Stress-strain representation of the  $J_2$ -plasticity model with isotropic hardening. The model is defined by the Young's modulus and the pairs of  $(\epsilon_i^p, \sigma_i^p)$ .



**Figure 7.2** In this example two different cyclic strain histories are examined. The first case is using a strain amplitude of 5%, and the second case is using a strain amplitude of 10%.



**Figure 7.3** Predicted stress-strain response during cyclic loading with a strain amplitude of 5% and 10%.

loop to grow in amplitude during cyclic loading. These figures show the predicted stress-strain response due to two different strain histories. In the first case the strain is cycled between +5% and -5%, and in the second case the strain is cycled between +10% and -10%. The predicted stress-strain response shows that the predicted stress becomes almost bilinear in response, and that the stress amplitude gradually increases during the cyclic loading. These inherent features of the model response are not in agreement with experimental data for either rubbers or thermoplastics for which the stress-strain loops during cyclic loading typically reach a steady-state response, or a response that is close to steady state.

If the goal of the finite element study is to predict the response during cyclic loading then it is typically not appropriate to use a plasticity model based on *isotropic hardening*. In this case it is better to use a plasticity model based on kinematic hardening, or a viscoplastic material model.

### 7.2.2 Matlab Implementation

The isotropic hardening plasticity model, for the case of uniaxial loading, can be implemented using a short Matlab script. In the implementation shown here, the stress in a given strain increment is obtained from the radial return mapping algorithm which can be written:

1. Initialize variables:  $\sigma_0 = 0$ ,  $\alpha_0 = 0$ ,  $i = 1$ ,  $\varepsilon_0^p = 0$ .  
Where  $\sigma_0$  is the initial stress,  $\alpha_0$  is the initial plastic strain, and  $\varepsilon_0^p$  is the initial plastic strain.
2. Take an elastic trial step:

$$\sigma_i^{\text{trial}} = \sigma_{i-1} + E(\varepsilon_i - \varepsilon_{i-1}). \quad (7.2)$$

3. Check if plastic flow is active

$$f = \sigma_i^{\text{trial}} - \sigma_y(\alpha_i).$$

If  $f < 0$  then accept the trial step:

$$\sigma_i = \sigma_i^{\text{trial}}.$$

If  $f \geq 0$  then let  $h$  be the tangent modulus (i.e. hardness) at the current plastic strain magnitude, then perform the following updates:

$$\Delta\gamma = \frac{f}{E + h}, \quad (7.3)$$

$$\sigma_i = \sigma_i^{\text{trial}} - \Delta\gamma E \text{sign}(\sigma_i^{\text{trial}}), \quad (7.4)$$

$$\alpha_i = \alpha_{i-1} + \Delta\gamma. \quad (7.5)$$

In order to make the theory for the  $J_2$ -plasticity model with isotropic hardening more concrete it is useful to study a numerical implementation of the governing equations. A convenient way to do this is to use Matlab, as illustrated below.

Matlab Code “mat\_PlasticityJ2Iso.m”:

```
function [stress] = mat_PlasticityJ2Iso(time, strain, params)
%mat_PlasticityJ2Iso Elastic-Plastic model with isotropic hardening
%Uniaxial loading
%Params = [E, sigY0, sigY1, eps1, sigY2, eps2, ... sigYn, epsn]

N = length(params);
E = params(1);
sigmaY = [params(2) params(3:2:N)];
epsP = [0 params(4:2:N)];

stress = 0 * strain;
alpha = 0; % plastic strain magnitude
for i = 2 : length(strain)
    stressTrial = stress(i-1) + E * (strain(i) - strain(i-1));
    stressTrialMag = abs(stressTrial);
    sigmaYcurr = interp1(epsP, sigmaY, alpha, 'linear', sigmaY(end));
    fTrial = stressTrialMag - sigmaYcurr;
    if fTrial < 0
        stress(i) = stressTrial; % elastic loading
    else
        h = (interp1(epsP, sigmaY, alpha+1e-4, 'linear', sigmaY(end)) ...
            - sigmaYcurr) / 1e-4;
        dgamma = fTrial / (E+h); % plastic with radial return mapping
        stress(i) = stressTrial - dgamma * E * sign(stressTrial);
        alpha = alpha + dgamma;
    end
end
end
```

The code is written in the form of a Matlab function. The function takes as input a vector of time, a vector of strain, and a vector of material parameters. The time vector is not used since the model is strain-rate independent, but kept as an argument to the function in order to make all material model functions presented in this book have the same input arguments.

The material parameters that are required for the Matlab implementation are:  $[E, \sigma_y^0, \sigma_y^1, \varepsilon_1^p, \sigma_y^2, \varepsilon_2^p, \dots]$ . That is, the Young's



modulus followed by pairs of yield stress and the corresponding plastic strain. Note that the initial plastic strain is automatically set to be zero.

The example Matlab file presented here can be made strain rate dependent by incorporating a yield function that depends on the plastic strain rate. This modification is left as an exercise.

### 7.2.3 Python Implementation

A slightly more rigorous implementation of isotropic hardening plasticity is presented in [Figure 7.4](#). In this case the Python code includes a search for the plastic strain increment that is consistent

Python Code "mat\_PlasticityJ2Iso.py":

```

from pylab import *
import scipy.optimize

def errfunc(dgamma, alpha0, epsP_vec, sigmaY_vec, E, stressTrial):
    sigmaY1 = interp(alpha0+dgamma, epsP_vec, sigmaY_vec)
    return abs(stressTrial - dgamma * E * sign(stressTrial) \
              - sigmaY1)

def plasticity_J2iso(strain, params):
    """Uniaxial loading. [E, sigY0, sigY1, eps1, ...]"""
    N = len(params)
    E = params[0]
    sigmaY_vec = [params[1]]
    sigmaY_vec.extend(params[2::2])
    epsP_vec = [0.0]
    epsP_vec.extend(params[3::2])
    stress = zeros(len(strain))
    alpha0 = 0.0
    for i in range(1, len(strain)):
        stressTrial = stress[i-1] + E * (strain[i] - strain[i-1])
        sigmaY0 = interp(alpha0, epsP_vec, sigmaY_vec)
        fTrial = abs(stressTrial) - sigmaY0
        if fTrial < 0:
            stress[i] = stressTrial
        else:
            sigmaY0 = interp(alpha0, epsP_vec, sigmaY_vec)
            h0 = (interp(alpha0+1.0e-4, epsP_vec, sigmaY_vec) \
                  - sigmaY0) / 1.0e-4
            dgamma = fTrial / (E+h0)
            res = scipy.optimize.fmin(errfunc, x0=[dgamma], \
                                     xtol=1e-9, ftol=1e-9, maxfun=9999, \
                                     full_output=1, disp=0, \
                                     args=(alpha0, epsP_vec, sigmaY_vec, E, stressTrial))
            dgamma = res[0][0]
            stress[i] = stressTrial - dgamma * E * sign(stressTrial)
            alpha0 = alpha0 + dgamma
    return stress

strain = linspace(0.0, 0.8, 10)
params = [50.0, 10.0, 15.0, 0.1]

stress = plasticity_J2iso(strain, params)

plot(strain, stress, 'b.-')
```

**Figure 7.4** Python implementation of the  $J_2$ -plasticity model with isotropic hardening. File name: mat\_PlasticityJ2Iso.py.

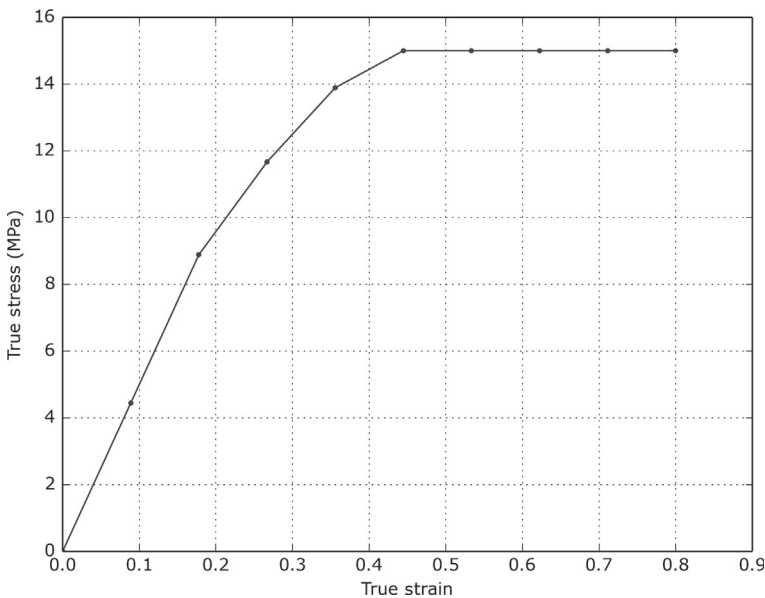
with the yield condition. This can be necessary in cases where the applied strain increment is large so that the hardening response becomes non-linear during the increment. The traditional way to search for the correct plastic strain increment is to use the Newton method, but the example here is simply using a generic search algorithm in order to keep the code clean.

The code in this case also contain the commands to generate and plot the complete stress-strain response during monotonic tension. The resulting figure showing the predicted stress-strain behavior is shown in [Figure 7.5](#).

Also in this case the plasticity model can be made rate-dependent using an equation like the following:

$$\sigma_{\text{yield}} = \sigma_{\text{yield}}^0 \cdot \left[ 1 + \left( \frac{\dot{\epsilon}^{\text{P}}}{D} \right)^{1/n} \right].$$

The implementation of this is left as an exercise.

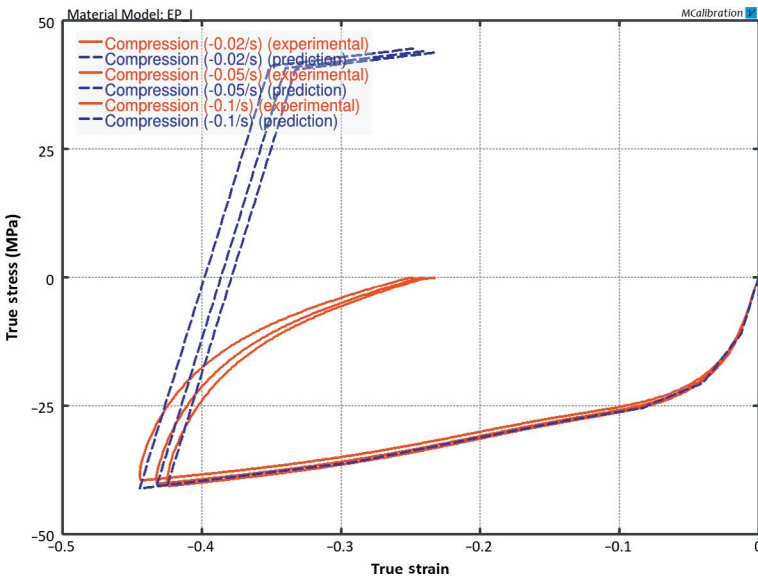


**Figure 7.5** Predicted stress-strain response obtained from the file `mat_PlasticityJ2Iso.py`.

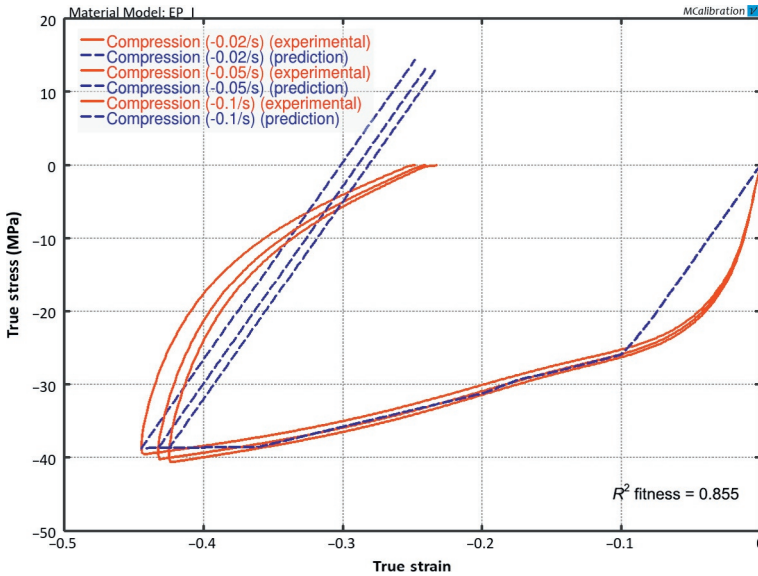
## 7.2.4 Application to Thermoplastics

The applicability of the isotropic hardening plasticity model is in this section examined by direct comparison with experimental data for a ultra-high molecular weight polyethylene (UHMWPE) thermoplastic material. The experimental data in this case was obtained in uniaxial compression at three different engineering strain rates ( $-0.02/s$ ,  $-0.05/s$ , and  $-0.10/s$ ) to a true strain of about  $-0.45$ , followed by unloading to zero stress.

The isotropic hardening plasticity model was calibrated to the experimental data in two ways. First the model was calibrated to the monotonic compression to the minimum strain using the intermediate strain rate. The results from that calibration are shown in Figure 7.6. This figure shows that, as expected, the model is very accurate at predicting the monotonic compressive response, but the model prediction of the unloading response is quite inaccurate since it severely underestimates the amount of recovery of the material response during unloading.



**Figure 7.6** Comparison between experimental data for UHMWPE and predictions from the isotropic hardening plasticity model calibrated the monotonic compressive material response.



**Figure 7.7** Comparison between experimental data for UHMWPE and predictions from the isotropic hardening plasticity model calibrated all the experimental data.

A different way to calibrate the material model is use all compressive data. [Figure 7.7](#) illustrates that when calibrated this way the overall predictions from the model may be better on average, but this calibrated model is likely less useful for most practical applications since the calibrated material model is not accurate at any section of the experimental stress-strain curve.

To accurately predict this material response requires a more advanced viscoplastic material model such as the Three Network (TN) model. This model is discussed in more detail in Section 8.6 of Chapter 8.

### 7.3 Plasticity with Kinematic Hardening

The stress for a kinematic hardening plasticity model of Chaboche type [7] is given by the following hardening model:

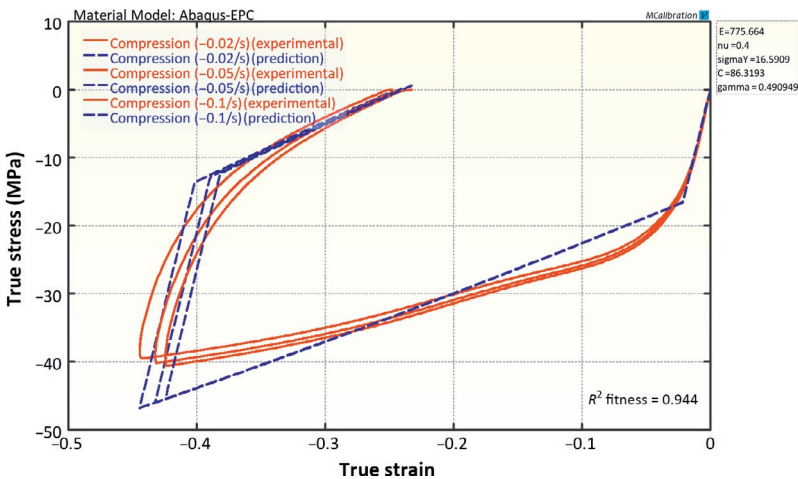
$$\dot{\sigma}_{\text{back}} = \frac{2\alpha}{3\beta} \sigma_y \dot{\epsilon}^p - \frac{\|\dot{\epsilon}^p\|}{\beta} \sigma_{\text{back}}, \quad (7.6)$$

where  $\dot{\sigma}_{\text{back}}$  is the time-derivative of the back stress, and  $\dot{\epsilon}^p$  the time-derivative of the plastic strain. The parameter  $\sigma_y$  is the initial yield stress,  $\alpha\sigma_y$  is the final yield stress, and  $\beta$  the transition strain for the yield evolution.

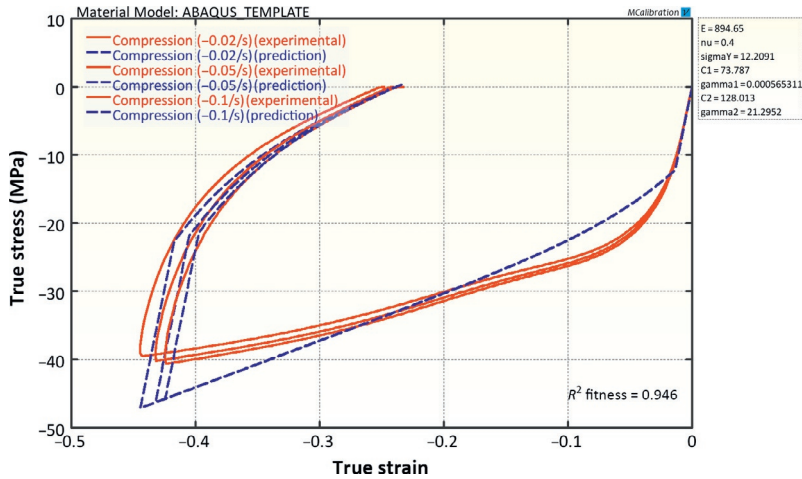
The applicability of the kinematic hardening plasticity model was examined by direct comparison with experimental data for a UHMWPE thermoplastic material. The experimental data in this case was obtained in uniaxial compression at three different engineering strain rates ( $-0.02/\text{s}$ ,  $-0.05/\text{s}$ , and  $-0.10/\text{s}$ ) to a true strain of about  $-0.45$ , followed by unloading to zero stress. The kinematic hardening plasticity model was calibrated to the experimental data using the MCalibration software [8]. The results from that calibration are shown in Figures 7.8–7.10.

The first of these figures compare the predictions of the kinematic hardening model with one backstress network to the experimental data. The Abaqus material definition that was used in this figure is shown below.

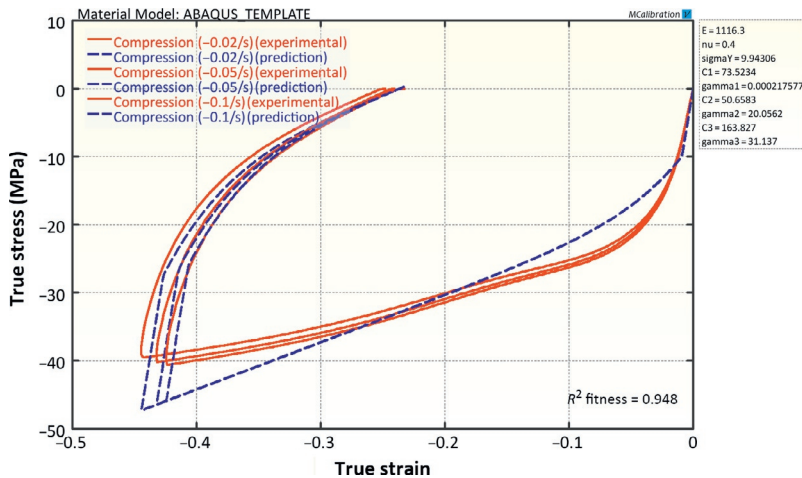
```
*Material, name=mat
*Elastic
775.664, 0.4
*Plastic, hardening=combined, data type=parameters
16.5909, 86.3193, 0.490949
```



**Figure 7.8** Calibration of an elastic-plastic material model with combined kinematic hardening and one backstress network.



**Figure 7.9** Calibration of an elastic-plastic material model with combined kinematic hardening and *two* backstress networks.



**Figure 7.10** Calibration of an elastic-plastic material model with combined kinematic hardening and *three* backstress networks.

As is shown in [Figure 7.9](#), adding a second backstress network slightly improves the model predictions. The Abaqus material definition that was used in this figure is shown below.

```
*Material, name=mat
*Elastic
894.65, 0.4
*Plastic, hardening=combined, data type=parameters, number backstresses=2
12.2091, 73.787, 0.000565311, 128.013, 21.2952
```

As is shown in [Figure 7.10](#), adding a third backstress network slightly further improves the model predictions. The Abaqus material definition that was used in this figure is shown below.

```
*Material, name=mat
*Elastic
1116.3, 0.4
*Plastic, hardening=combined, data type=parameters, number backstresses=3
9.94306, 73.5234, 0.000217577, 50.6583, 20.0562, 163.827, 31.137
```

One of the main limitations of this material model is that the tangent modulus is monotonically decreasing, making it impossible to capture the large strain tensile response of many ductile thermoplastics that are known to have a stress-strain response that stiffens at large strains.

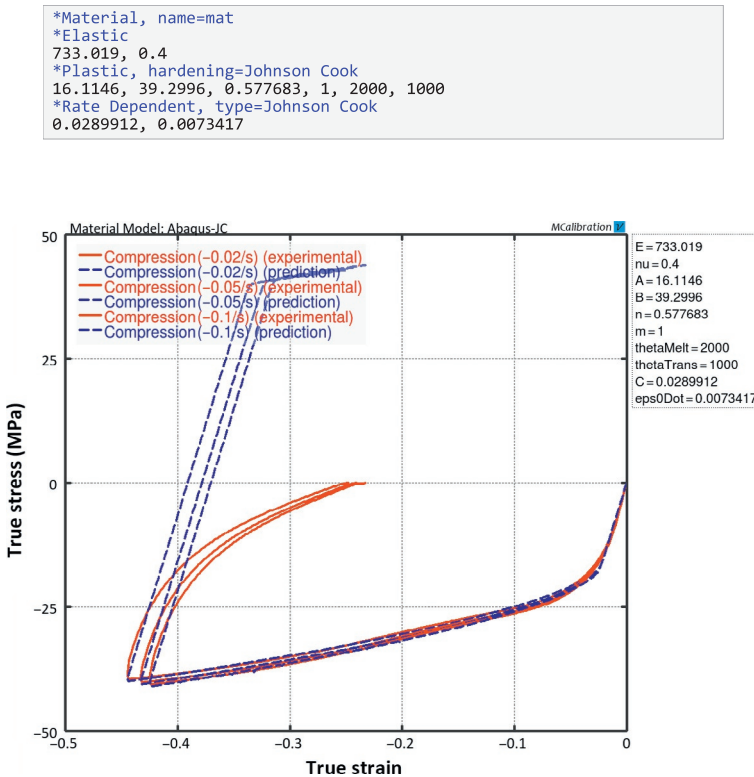
## 7.4 Johnson-Cook Plasticity

The Johnson-Cook model is a plasticity model that is based on Mises plasticity with closed-form analytical equations specifying the hardening behavior and the strain-rate dependence of the yield stress. In this model the yield stress is given by the following equation:

$$\sigma_{\text{yield}} = [A + B(\varepsilon^{\text{P}})^n] \left[ 1 + C \ln \left( \frac{\dot{\varepsilon}^{\text{P}}}{\dot{\varepsilon}_0} \right) \right] (1 - \hat{\theta}^m), \quad (7.7)$$

where  $[A, B, n, C, \dot{\varepsilon}_0, m]$  are material parameters that need to be determined from experimental data. In this equation  $\varepsilon^{\text{P}}$  is the effective plastic strain, and  $\dot{\varepsilon}^{\text{P}}$  is the time derivative of the effective plastic strain.

Since this material model is based on isotropic hardening, it cannot accurately predict the unloading response of many thermoplastics. [Figure 7.11](#) shows the predicted stress-strain response when calibrated to cyclic uniaxial compression data for UHMWPE. The figure shows that the predicted response under monotonic loading is in good agreement with the experimental behavior, but that the predicted unloading behavior significantly overpredicts the residual deformation after unloading. The calibrated material parameters that were used to generate the model predictions are listed in Abaqus inp-file format below.



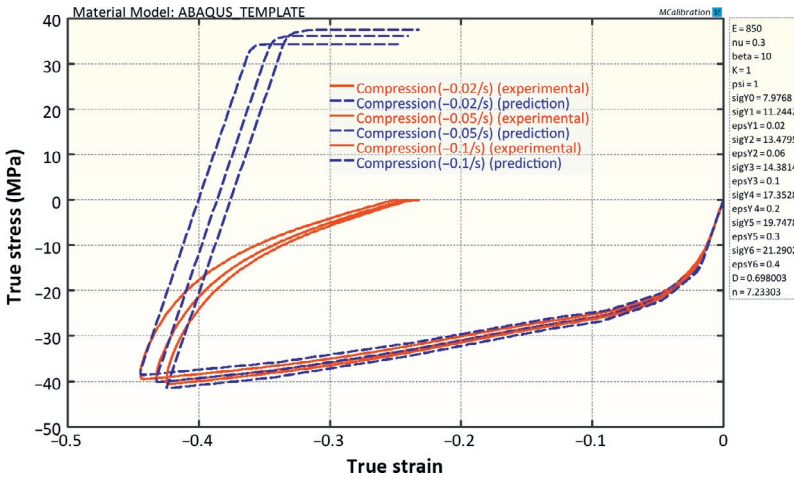
**Figure 7.11** Comparison between experimental data for UHMWPE and predictions from a Johnson-Cook plastic material.

## 7.5 Drucker Prager Plasticity

Drucker Prager plasticity is an isotropic hardening plasticity model specifically developed for frictional materials like granular solids. One interesting feature of this model is that it can have a yield stress that depends on the pressure, and hence can have different yield stresses in tension and compression. This is an attractive feature since many thermoplastics are known to have a pressure dependent yield stress. The model framework can also incorporate strain-rate dependence and progressive failure predictions.

One of the main limitations of the Drucker Prager model is that it is based on isotropic hardening, which means that it cannot accurately predict the unloading response of many thermoplastics. [Figure 7.12](#) shows the predicted stress-strain response when





**Figure 7.12** Comparison between experimental data for UHMWPE and predictions from a Drucker Prager plastic material.

calibrated to cyclic uniaxial compression data for UHMWPE. The figure shows that the predicted response under monotonic loading is in good agreement with the experimental behavior, but that the predicted unloading behavior significantly overpredicts the residual deformation after unloading. The calibrated material parameters that were used to generate the model predictions are listed in Abaqus inp-file format below.

```

*Material, name=mat
*Elastic
850, 0.3
*Drucker Prager
10, 1, 1
*Drucker Prager Hardening, type=tension
7.9768, 0.0
11.2442, 0.02
13.4795, 0.06
14.3814, 0.1
17.3528, 0.2
19.7478, 0.3
21.2902, 0.4
*Rate Dependent, type=power law
0.698003, 7.23303
  
```

## 7.6 Use of Plasticity Models in Polymer Modeling

Plasticity models have a long history of use in finite element simulations, and have also been extensively used to represent the behavior of thermoplastic materials. As was shown in this

chapter, however, using a plasticity model to predict the non-linear viscoplastic response of polymers can give poor predictions of the material response, particularly during cyclic loading or during unloading.

There are many different plasticity models that have been developed and that are available in finite element software. In practice, the different models can be divided into two groups: *isotropic hardening* plasticity and *kinematic hardening* plasticity. The isotropic hardening plasticity models are often easier to use and can provide accurate predictions all the way to failure (under monotonic loading). The kinematic hardening models are often based on non-linear equations with material parameters that need to be determined from experimental tests, and hence are often more difficult to calibrate. The kinematic hardening models often have a restriction that the tangent modulus decreases with increasing plastic strain, a restriction that is contradicting experimental behaviors of many polymers.

Plasticity models can still be important for practical use in a finite element setting since they are numerically very efficient. But plasticity models are never the most accurate material modeling framework. The viscoplastic models presented in the next chapter will always be more accurate.

## 7.7 Exercises

1. Show how to specify an isotropic hardening plasticity model in your favorite finite element program.
2. Why is the isotropic hardening plasticity model so easy to calibrate?
3. What experimental tests would you run in order to calibrate an isotropic hardening plasticity model?
4. What is the main limitation of isotropic hardening plasticity?
5. Describe the difference between the Matlab and the Python implementations of the isotropic hardening plasticity model.

6. What are some of the limitations of the kinematic hardening plasticity model?
7. Create a Matlab or Python implementation of the combined kinematic plasticity model.
8. What experimental tests would you run in order to calibrate a kinematic hardening plasticity model?
9. What are some features of the Johnson-Cook plasticity model that make it interesting?
10. What are some differences between the Johnson-Cook and the Drucker Prager plasticity models?

## References

- [1] R. Hill. *The Mathematical Theory of Plasticity*. Oxford Classic Texts in the Physical Sciences, Oxford, 1998.
- [2] J. Lubliner. *Plasticity Theory*. Dover Books on Engineering, New York, 2008.
- [3] J.C. Simo, T.J.R. Hughes. *Computational Inelasticity*. Springer, New York, 2000.
- [4] F. Dunne, N. Petrinic. *Introduction to Computational Plasticity*. Oxford University Press, Oxford, 2005.
- [5] A.S. Khan, S. Huang. *Continuum Theory of Plasticity*. Wiley Interscience, New York, 1995.
- [6] HKS, Inc.. ABAQUS, Pawtucket, RI, ver. 6.14, 2014.
- [7] J. Lemaitre, J.-L. Chaboche. *Mechanics of Solid Materials*. Cambridge University Press, Cambridge, 1990.
- [8] Veryst Engineering. <http://PolyUMod.com>.

# 8 Viscoplasticity Models

---

## Chapter Outline

8.1	Introduction	372
8.2	Bergström-Boyce Model	372
8.2.1	Matlab Implementation of the BB-Model	382
8.2.2	Python Implementation of the BB-Model	384
8.2.3	Generic Numerical Implementation	386
8.2.4	Dynamic Loading Predictions	387
8.2.5	Use of the BB-Model for Polymer Modeling	392
8.3	Arruda-Boyce Model	393
8.4	Dual Network Fluoropolymer Model	397
8.4.1	Matlab Implementation of the DNF Model	404
8.4.2	Use of the DNF Model for Polymer Modeling	404
8.5	Hybrid Model	409
8.5.1	Matlab Implementation of the Hybrid Model	413
8.5.2	Use of the Hybrid Model for Polymer Modeling	414
8.6	Three Network Model	417
8.6.1	Matlab Implementation of the Three Network Model	422
8.6.2	Python Implementation of the Three Network Model	422
8.6.3	Use of the Three Network Model for Polymer Modeling	426
8.7	Parallel Network Model	427
8.8	Use of Viscoplasticity in Polymer Modeling	431
8.9	Python Code Examples	432
8.10	Exercises	434
	References	435

## 8.1 Introduction

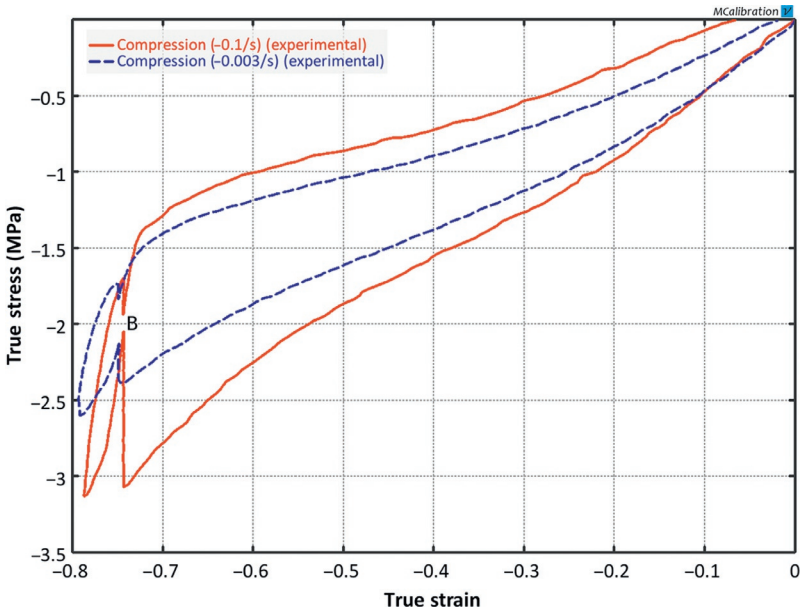
Viscoplasticity models are the ultimate models for polymer modeling, both in terms of accuracy and complexity. Some of the models of this category are truly impressive at predicting the non-linear, time- and temperature-dependent response of various polymers. The downsides of these models are that they typically require more experimental data for proper calibration, they can be numerically expensive, and typically require additional software components (user subroutines, like UMATs and VUMATs, see Chapter 10).

The field of viscoplastic constitutive modeling is evolving, and new models are developed every year by different research groups. This chapter attempts to present some of the most useful models that are available for different polymers, and the trends of where this field is going in the future.

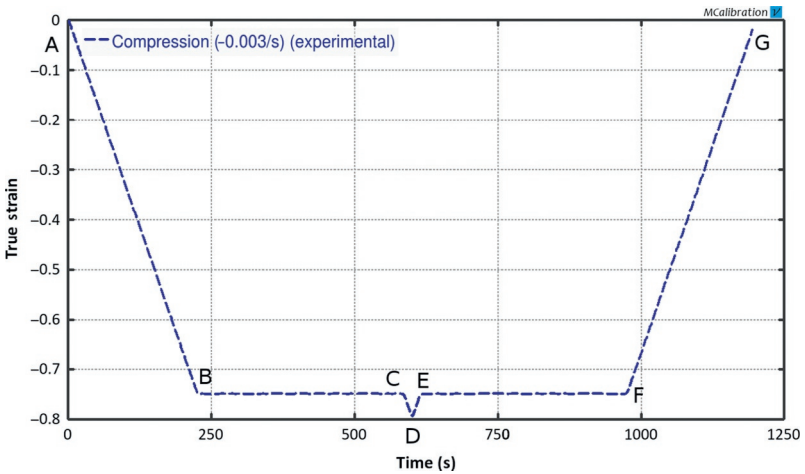
## 8.2 Bergström-Boyce Model

The Bergström-Boyce (BB) model [1–5] is an advanced constitutive model for predicting the non-linear time-dependent, large-strain behavior of elastomer-like materials. The model has been shown to be accurate for both traditional engineering rubbers, and soft biomaterials [1–5]. The BB-model can also be considered a powerful generalization of linear viscoelasticity, and as will be shown in this section can overcome some of the main issues with linear viscoelasticity related to both large strain deformations and strain amplitude dependence of the dynamic properties ( $E'$  and  $E''$ ).

The motivation for the BB-model can be described using [Figure 8.1](#). This figure shows experimental data for a chloroprene rubber tested in uniaxial compression at two different strain rates ( $-0.1/s$ , and  $-0.03/s$ ), although the qualitative response shown in this figure is true for all elastomers. The red solid curve shown in the figure is the experimentally determined stress-strain response after being exposed to the applied strain history shown in [Figure 8.2](#). The dashed blue line is the experimental data from



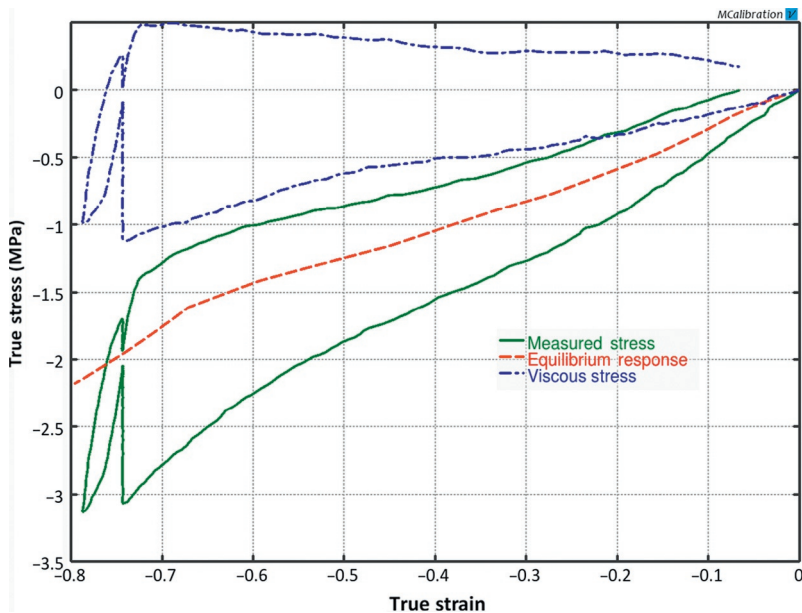
**Figure 8.1** Experimental data for a chloroprene rubber tested at two strain rates. The strain was held constant for 10 min at a true strain of  $-0.75$ .



**Figure 8.2** Applied strain history used to test the chloroprene rubber. The strain was held constant for 10 min at a true strain of  $-0.75$ .

a similar experiment performed at a 50 times slower applied strain-rate. What makes the results in [Figure 8.1](#) so interesting is the stress relaxation that occurs during the hold segments (i.e. B-C and E-F in [Figure 8.2](#)). In this case the stress relaxes significantly and the stress relaxation during loading and during unloading appears to asymptotically approach the same stress value. Note that the stress relaxation during unloading is negative. This is, even though the specimen has been compressed and has a negative stress, the stress actually grows larger in magnitude during the stress relaxation segment.

The stress value that is approached at Point *B* in [Figure 8.1](#) can therefore be considered the *equilibrium stress* at a strain value of about  $-0.75$ . By performing a set of experiments of this type, where stress relaxation segments have been inserted, makes it possible to experimentally establish an equilibrium response for the elastomer, see the red curve in [Figure 8.3](#). The equilibrium curve is the hypothetical stress-strain curve that would be



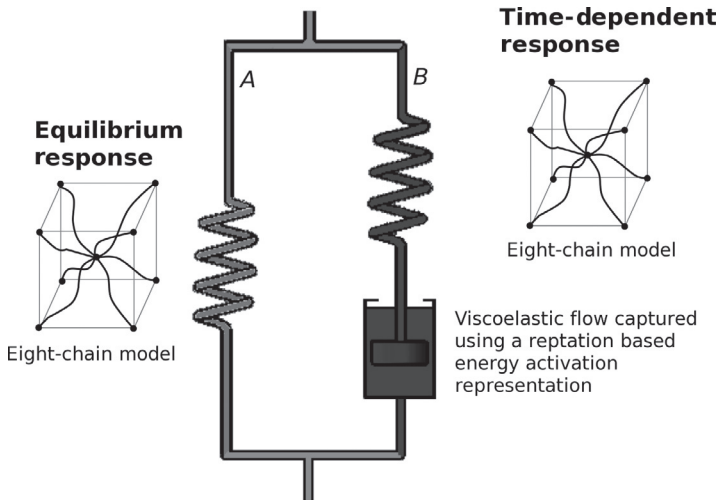
**Figure 8.3** The experimentally measured stress (red solid line) can be decomposed into an equilibrium stress (green dashed line) and a viscoelastic stress (blue dash-dot line).

obtained if the experiment was run in infinite time (i.e. if the strain-rate was zero). This observation makes it possible to divide the experimentally determined stress response into an equilibrium response (green dashed line in Figure 8.3), and a time-dependent deviation from the equilibrium response (see the blue dash-dot curve in Figure 8.3).

In other words, the true response of the elastomer can be represented using two parallel networks *A* and *B*, see Figure 8.4. Network *A* is a non-linear hyperelastic network, and network *B* consists of a non-linear hyperelastic component in series with a non-linear viscoelastic flow element.

Note that the viscoelastic flow response has to be non-linear since otherwise the model framework will be the same as linear viscoelasticity with a one-term Prony series, which was shown in Chapter 6 to be insufficient for capturing the general response of elastomers.

The idea to decompose the total stress into an elastic and a history dependent component was proposed by Green and Tobolsky [6], and the approach to model elastomers as two interacting networks has been used in different variations by Johnson and



**Figure 8.4** Constitutive representation of the elements of the BB model.



Quigley [7], Johnson and Stacer [8], Johnson et al. [9], Roland [10], and Roland and Warzel [11].

In the BB model the applied deformation gradient is acting on two parallel macromolecular networks:  $\mathbf{F} = \mathbf{F}_A = \mathbf{F}_B$ . The deformation gradient acting on the time-dependent network  $B$  is further decomposed into elastic and viscoelastic components:

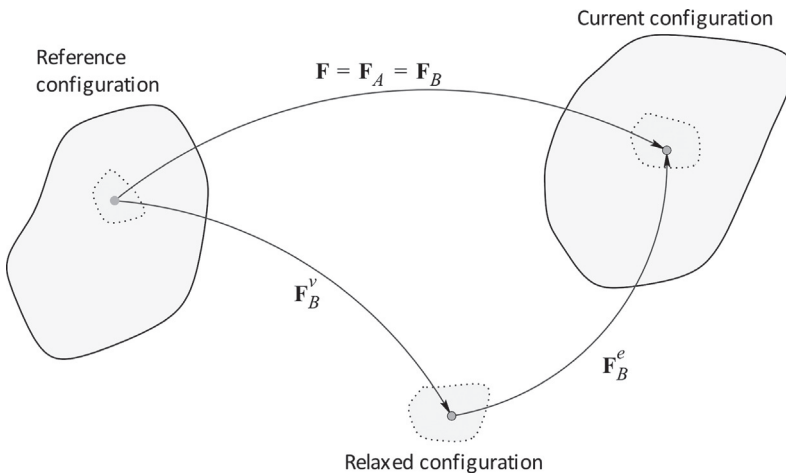
$$\mathbf{F}_B = \mathbf{F}_B^e \mathbf{F}_B^v. \quad (8.1)$$

The decomposition of the deformation gradient into elastic and viscoelastic components is shown in the deformation map in Figure 8.5.

The response of network  $A$  is given by the eight-chain model (see Section 5.3.10):

$$\boldsymbol{\sigma}_A = \frac{\mu}{J\lambda^*} \frac{\mathcal{L}^{-1}(\bar{\lambda}^*/\lambda^{\text{lock}})}{\mathcal{L}^{-1}(1/\lambda^{\text{lock}})} \text{dev}[\mathbf{b}^*] + \kappa(J - 1)\mathbf{I}, \quad (8.2)$$

where  $[\mu, \lambda^{\text{lock}}, \kappa]$  are material parameters,  $\mathcal{L}^{-1}(\cdot)$  the inverse Langevin function,  $J = \det[\mathbf{F}]$ , and  $\mathbf{b}^*$  the distortional part of the left Cauchy-Green tensor. The stress on network  $B$  is also given by the eight-chain model, but with a different effective shear modulus:



**Figure 8.5** Multiplicative decomposition of the deformation.

$$\boldsymbol{\sigma}_B = \frac{s \mu}{J_B \overline{\lambda}_B^{e*}} \frac{\mathcal{L}^{-1} \left( \overline{\lambda}_B^{e*} / \lambda^{\text{lock}} \right)}{\mathcal{L}^{-1} \left( 1 / \lambda^{\text{lock}} \right)} \text{dev}[\mathbf{b}_B^{e*}] + \kappa (J_B^e - 1) \mathbf{I}, \quad (8.3)$$

where  $s$  is a dimensionless material parameter specifying how much stiffer the shear modulus of network  $B$  is relative to network  $A$ , and  $\overline{\lambda}_B^{e*}$  is the chain stretch in the elastic part of Network  $B$ . Using this representation the total Cauchy stress is given by

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_A + \boldsymbol{\sigma}_B. \quad (8.4)$$

Note that the presentation here is based on the original BB-model in which the hyperelastic response is given by the Arruda-Boyce (AB) eight-chain model. This approach is also used in the ANSYS [12], MSC.Marc [13], and LS-DYNA [14] native implementations of the BB-model. The Abaqus [15] implementation (called \*HYSTERESIS), is more general and allows for any of the built-in hyperelastic models to be used.

Equations (8.1)–(8.3) completely specify how the stress can be calculated in any given deformation state (with known internal state  $\mathbf{F}_B^v$ ). To complete the model it is also necessary to specify the rate of change in the internal state of the material through viscoelastic flow, that is, the rate kinematics of the model. The velocity gradient on network  $B$ ,  $\mathbf{L}_B = \dot{\mathbf{F}}_B \mathbf{F}_B^{-1}$ , can be decomposed into elastic and viscous components as follows:

$$\begin{aligned} \mathbf{L}_B &= \left[ \frac{d}{dt} (\mathbf{F}_B^e \mathbf{F}_B^v) \right] (\mathbf{F}_B^e \mathbf{F}_B^v)^{-1} \\ &= [\dot{\mathbf{F}}_B^e \mathbf{F}_B^v + \mathbf{F}_B^e \dot{\mathbf{F}}_B^v] (\mathbf{F}_B^v)^{-1} (\mathbf{F}_B^e)^{-1} \\ &= \dot{\mathbf{F}}_B^e (\mathbf{F}_B^e)^{-1} + \mathbf{F}_B^e \dot{\mathbf{F}}_B^v (\mathbf{F}_B^v)^{-1} (\mathbf{F}_B^e)^{-1} \\ &= \mathbf{L}_B^e + \mathbf{F}_B^e \mathbf{L}_B^v \mathbf{F}_B^e \\ &= \mathbf{L}_B^e + \tilde{\mathbf{L}}_B^v, \end{aligned} \quad (8.5)$$

where

$$\mathbf{L}_B^v = \dot{\mathbf{F}}_B^v (\mathbf{F}_B^v)^{-1} = \mathbf{D}_B^v + \mathbf{W}_B^v, \quad (8.6)$$

$$\tilde{\mathbf{L}}_B^v = \tilde{\mathbf{D}}_B^v + \tilde{\mathbf{W}}_B^v. \quad (8.7)$$

To make the unloading unique [16], prescribe  $\tilde{\mathbf{W}}_B^v \equiv 0$ . The rate of viscous deformation of network  $B$  is constitutively prescribed by:

$$\tilde{\mathbf{D}}_B^v = \dot{\gamma}_B(\boldsymbol{\sigma}_B, \mathbf{b}_B^{e*}) \mathbf{N}_B^v, \quad (8.8)$$

where

$$\mathbf{N}_B^v = \frac{\text{dev}[\boldsymbol{\sigma}_B]}{\tau} = \frac{\text{dev}[\boldsymbol{\sigma}_B]}{\|\text{dev}[\boldsymbol{\sigma}_B]\|_F} \quad (8.9)$$

provides the direction of the viscoelastic flow, and  $\tau$  is the effective (scalar) stress driving the viscous flow.

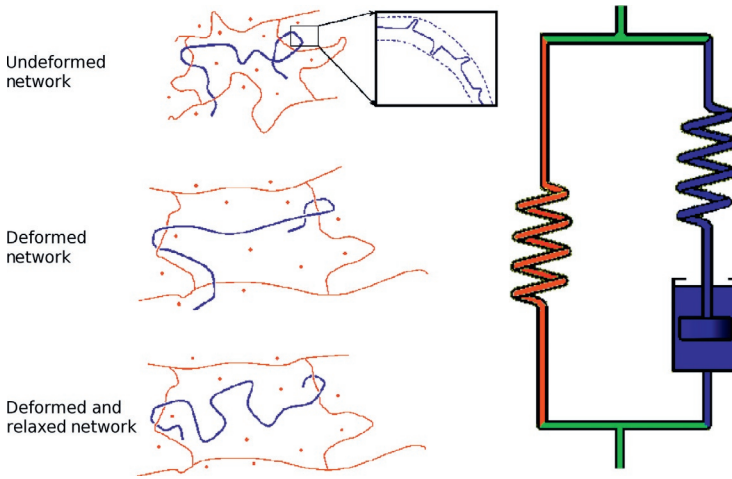
The time derivative of  $\mathbf{F}_B^v$  can be derived as follows:

$$\begin{aligned} \tilde{\mathbf{L}}_B^v &= \dot{\gamma}_B^v \mathbf{N}_B^v, & (8.10) \\ \Rightarrow \mathbf{F}_B^e \dot{\mathbf{F}}_B^v (\mathbf{F}_B^v)^{-1} (\mathbf{F}_B^e)^{-1} &= \dot{\gamma}_B^v \mathbf{N}_B^v, \\ \Rightarrow \dot{\mathbf{F}}_B^v &= \dot{\gamma}_B^v (\mathbf{F}_B^e)^{-1} \frac{\text{dev}[\boldsymbol{\sigma}_B]}{\|\text{dev}[\boldsymbol{\sigma}_B]\|_F} \mathbf{F}_B^e \mathbf{F}_B^v. & (8.11) \end{aligned}$$

These equations for the time-derivative of the deformation gradient are based on pure kinematics, and only embody the physics of a specific material through a scalar equation for the effective flow rate  $\dot{\gamma}_B$ . The effective flow rate  $\dot{\gamma}_B$  of Network  $B$  must be constitutively prescribed. Here, a micromechanism-inspired model is proposed on the assumption that the mechanism responsible for the time-dependent behavior is the reptation of macromolecules that are “elastically inactive” (i.e. molecules that carry less load and have the capability to significantly change conformation during creep loading.)

To illustrate this view consider first an over-simplified model with one free chain located in a network of chains as shown in [Figure 8.6](#).

If the network is deformed at a high enough rate then the free chain will also deform more or less affinely with the network. Hence the entropy of the free chain is decreased and the free chain contributes additional deformation resistance. If the applied strain is then held constant in the deformed state the free chain will slowly, by Brownian motion, return to a more relaxed



**Figure 8.6** Schematic representation of the molecular deformation that occurs during deformation and stress relaxation of crosslinked polymer.

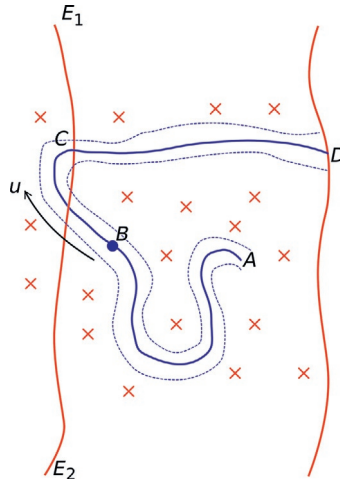
configuration. The rate of return toward a relaxed configuration is given by the governing reptation motion of the free chain [17].

Let us now turn our attention to a more realistic situation in which free chains do not exist. In a real polymer network, however, there are almost always free chain-ends which behave as the free chain described above. A reasonable extension to the free chain and free chain-end models is to consider also inactive chain segments such as  $A - B - C$  illustrated in Figure 8.7.

Qualitatively, the same behavior is exhibited by the inactive chain segment in Figure 8.7 and the free chain in Figure 8.6. The loop  $A - B - C$  in Figure 8.7 undergoes Brownian motion and has an equilibrium position at a finite distance from the constraining chain  $DD'$ . Hence,  $DD'$  behaves as an obstacle which imposes an energy barrier to the relaxation process. The relaxation toward equilibrium can therefore be considered to be energy activated.

To develop the constitutive equation for the time-dependent element consider a free chain-end of the type illustrated in Figure 8.7, but bear in mind that the presented arguments also hold for the more general situation shown in Figure 8.6.

The chain segment at  $B$  is constrained to travel back and forth along the constraining tube by Brownian motion in a combination



**Figure 8.7** The stress relaxation is driven by molecular reptation.

of reptation motion and contour length fluctuations. Call the displacement of  $B$  along the tube  $u$ . The average displacement of  $B$  can be shown by the theory of reptational motion of chain molecules [18] to scale as  $\langle u \rangle = C_3 \langle u^2 \rangle^{1/2} \equiv C_3 \sqrt{\phi(t)}$ . The effective length of the chain in a creep experiment can consequently be written

$$l(t) = l_0 + C_3 \sqrt{\phi(t)}, \quad (8.12)$$

where the mean square displacement of the chain segment  $\phi(t)$  has been derived [18] by reptation dynamics considerations to scale as

$$\phi(t) \propto \begin{cases} t^{1/2}, & t \leq \tau_e, \\ t^{1/4}, & \tau_e \leq t \leq \tau_R, \\ t^{1/2}, & \tau_R \leq t \leq \tau_d, \\ t, & \tau_d \leq t, \end{cases} \quad (8.13)$$

where  $\tau_e$  is the time at which the tube constraint is first felt,  $\tau_R$  is the Rouse relaxation time, and  $\tau_d$  is the tube disengagement time. Equations (8.12) and (8.13) can be simplified to

$$\lambda_{\text{chain}}^{Bp}(t) = \frac{l(t)}{l_0} = 1 + C_4 t^{C_5}, \quad (8.14)$$

where  $C_4 > 0$  and  $C_5 \in [0.5, 1.0]$ . Taking the time derivative of (8.14) gives

$$\dot{\lambda}_{\text{chain}}^{Bp} = C_4 C_5 t^{C_5-1}. \quad (8.15)$$

It is now possible to eliminate  $t$  between (8.14) and (8.15) giving

$$\dot{\lambda}_{\text{chain}}^{Bp} = C_6 \left[ \lambda_{\text{chain}}^{Bp} - 1 \right]^{C_7}, \quad (8.16)$$

where  $C_6 > 0$  and  $C_7$  is about  $-1$ . This equation shows how the *effective* creep rate depends on the chain stretch, where the chain stretch is correlated to the principal macroscopic stretch state by the 8-chain assumption [19], i.e.

$$\lambda_{\text{chain}}^{Bp} = \sqrt{\frac{I_1^{Bp}}{3}} = \sqrt{\frac{(\lambda_1^{Bp})^2 + (\lambda_2^{Bp})^2 + (\lambda_3^{Bp})^2}{3}}. \quad (8.17)$$

Equation (8.16) gives the creep rate at a constant stress level, but the creep rate also depends on the level of the applied stress. And as discussed above, this stress dependence is assumed to be energy activated. The microstructural connection to the activation parameters is complicated; it is possible, however, to use a generic expression of the form

$$\dot{\gamma}_B = C_1 \left[ \lambda_{\text{chain}}^{Bp} - 1 + \xi \right]^{C_2} \left( \frac{\tau_B}{\hat{\tau}_B} \right)^m, \quad (8.18)$$

where  $\tau_B$  is the effective stress measure introduced in Equation (8.18),  $\hat{\tau}_B$  is a material constant, and  $\xi$  is small positive constant that is introduced to eliminate a removable singularity in the flow rate in the undeformed state [4].

Note that in the proposed model the constants  $\hat{C}_1 \equiv C_1/\hat{\tau}_B^m$  and  $m$  are positive; and  $C_2$  is a constant that is restricted by reptational dynamics to be negative.

In summary, the rate-equation for viscous flow is given by [1]:

$$\dot{\gamma}_B^v = \dot{\gamma}_0 \left( \overline{\lambda}_B^v - 1 + \xi \right)^C \left[ R \left( \frac{\tau}{f_v \tau_{\text{base}}} - \hat{\tau}_{\text{cut}} \right) \right]^m, \quad (8.19)$$

where:

- $[\xi, C, \tau_{\text{base}}, \hat{\tau}_{\text{cut}}, m]$  are material parameters.
- $\dot{\gamma}_0 \equiv 1/s$  is a constant introduced to ensure dimensional consistency.
- $R(x) = (x + |x|)/2$  is the ramp function.
- $f_v = 1 + \alpha \boldsymbol{\epsilon}$  :  $\boldsymbol{\epsilon}_e$  is a network interaction factor. This is a modification of the original BB flow model [3] that makes the flow resistance stress dependent on the network state through a factor  $f_v$  that depends on the two strains  $\boldsymbol{\epsilon} = \ln[\mathbf{v}]$ , and  $\boldsymbol{\epsilon}_e = \ln[\mathbf{v}_e]$ . The reason for the modification is that most elastomer-like materials are experimentally shown to have less strain-rate dependence during unloading than during loading. This flow model introduces the ability to capture this response.
- $\hat{\tau}_{\text{cut}}$  is a cut-off stress below which no flow will occur. The ramp function is introduced in order to increase the numerical efficiency of the material model for cases when regions of the FE mesh is not undergoing significant deformations, and to allow for predictions of true plastic deformation.
- $\bar{\lambda}_B^v = \sqrt{\frac{\text{tr}[\mathbf{b}_B^v]}{3}}$  is the viscoelastic chain stretch.
- The effective stress driving the viscous flow is:

$$\tau = \|\text{dev}[\boldsymbol{\sigma}_B]\|_F = \sqrt{\text{tr}[\boldsymbol{\sigma}'_B \boldsymbol{\sigma}'_B]}, \quad (8.20)$$

where  $\|\cdot\|_F$  is the Frobenius norm.

### 8.2.1 Matlab Implementation of the BB-Model

As was discussed in Section 8.2, the BB material model is formulated as a set of differential equations that need to be solved for each time increment. One way to solve this set of equations is to use the following algorithm:

1. Known values at time  $t_i$ :
  - Deformation gradient:  $\mathbf{F}$
  - State variables:  $\mathbf{F}_B^v$

2. Known values at time  $t_{i+1}$ :
  - Deformation gradient:  $\mathbf{F}$
3. Calculate  $\mathbf{F}_B^V$  at  $t_{i+1}$  using an ODE solver and Equation (8.11) for  $\dot{\mathbf{F}}_B^V(t, \mathbf{F}_B^V)$ .
4. Calculate  $\mathbf{F}^e$  using Equation (8.1), and  $\sigma_B$  using Equation (8.3) at  $t_{i+1}$ .

For the case of incompressible, uniaxial loading this algorithm can be implemented in Matlab as shown in [Figure 8.8](#).

Matlab File Name: mat\_BB.m

```
function [stress] = mat_BB(time, strain, params)
%mat_BB Bergstrom-Boyce model
%Incompressible uniaxial loading
%This function is using true stress and strain
muA = params(1);
lambdaL = params(2);
sB = params(3);
xi = params(4);
C = params(5);
tauHat = params(6);
m = params(7);
stress = 0 * strain;
lambdaP = 1;
for i = 2 : length(strain)
    stressA = mat_EC(time(i), strain(i), [muA lambdaL]);
    [t,lambdaP] = ode45(@(t,y) lambdaPDot(t, y, time, strain, sB*muA, lambdaL, ...
        xi, C, tauHat, m), time(i-1:i), lambdaP); % find lambdaP at t+dt
    lambdaP = lambdaP(end);
    strainP = log(lambdaP);
    strainE = strain(i) - strainP;
    stressB = mat_EC(time(i), strainE, [muA*sB lambdaL]);
    stress(i) = stressA + stressB;
end
end

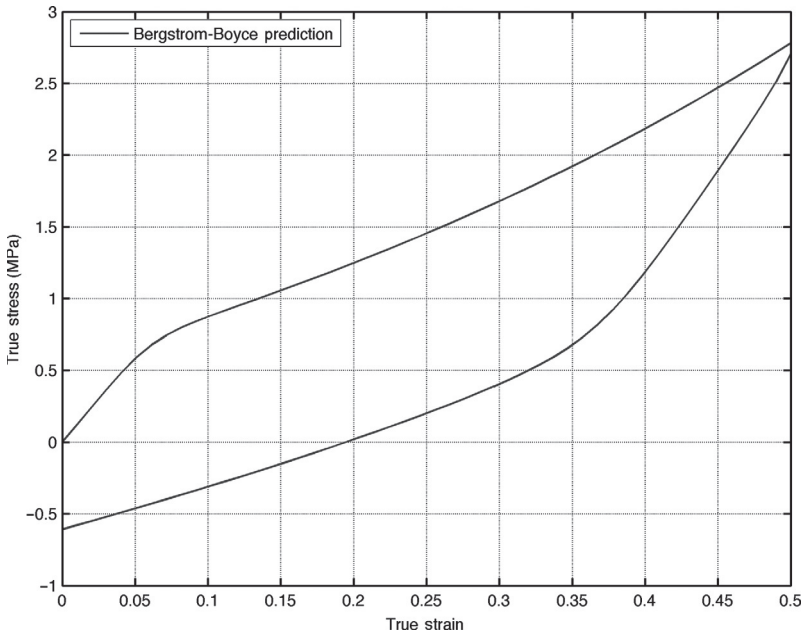
function lambdaPDot = lambdaPDot(time, lambdaP, timeVec, strainVec, muB, lambdaL, ...
    xi, C, tauHat, m)
lambda = exp( interp1(timeVec, strainVec, time) ); % stretch at the specified time
lambdaE = lambda / lambdaP;
lambdaPChain = sqrt((lambdaP^2 + 2 / lambdaP) / 3);
stressB = mat_EC(time, log(lambdaE), [muB lambdaL]);
gamDot = (lambdaPChain - 1 + xi)^C * (abs(stressB) / tauHat)^m;
lambdaPDot = gamDot * lambdaE * sign(stressB) * lambda;
end
```

Matlab File Name: test\_mat\_BB.m

```
time = linspace(0,2)';
strain = [linspace(0, 0.5,50)'; linspace(0.5, 0, 50)'];
stress = mat_BB(time, strain, [1 4 3 0.05 -0.5 1.0 4.0]);
plot(strain,stress)
```

**Figure 8.8** (a) Matlab implementation of the Bergström-Boyce model. (b) Exemplar Matlab code that calls the Matlab implementation of the BB model.





**Figure 8.9** Predictions from the Matlab file `test_mat_BB.m`.

The results from running the test file `test_mat_BB` are shown in [Figure 8.9](#).

## 8.2.2 Python Implementation of the BB-Model

For incompressible uniaxial loading the BB model can be implemented into Python code as shown below. The function `uniaxial_stress_visco()` is a generic function for calculating the stress for any viscoplastic material model specified by the function `model()`. This function searches for the transverse strain, at each time increment, that gives the correct stress boundary condition ( $\sigma_{22} = \sigma_{33} = 0$ ).

The implementation of the BB-model is provided in two functions `BB_timeDer_3D()` and `BB_3D()`. The time-derivative function is used by the Python function `scipy.integrate.odeint()`.

Code to "Polymer\_Mechanics\_Chap09.py":

```

from Polymer_Mechanics_Chap05 import *
import scipy.integrate

def ramp(x):
    return (x + abs(x)) / 2.0

def toVec(A):
    """Convert a 3x3 matrix to vector"""
    return array([A[0][0], A[1][1], A[2][2]])

def uniaxial_stress_visco(model, timeVec, trueStrainVec, params):
    """Compressible uniaxial loading. Returns true stress."""
    stress = zeros(len(trueStrainVec))
    lam2_1 = 1.0
    FBv1 = array([1.0, 1.0, 1.0])
    for i in range(1, len(trueStrainVec)):
        print 'uniaxial stress: i=', i, ' of ', len(trueStrainVec)
        time0 = timeVec[i-1]
        time1 = timeVec[i]
        lam1_0 = exp(trueStrainVec[i-1])
        lam1_1 = exp(trueStrainVec[i])
        lam2_0 = lam2_1
        F0 = array([lam1_0, lam2_0, lam2_0])
        F1 = array([lam1_1, lam2_1, lam2_1])
        FBv0 = FBv1.copy()
        calcS22Abs = lambda x: abs(model(F0, array([lam1_1,x,x]), \
            FBv0, time0, time1, params)[0][1])
        # search for transverse stretch that gives S22=0
        lam2_1 = scipy.optimize.fmin(calcS22Abs, x0=lam2_0, \
            xtol=1e-9, ftol=1e-9, disp=False)
        res = model(F0, array([lam1_1, lam2_1, lam2_1]), FBv0, \
            time0, time1, params)
        stress[i] = res[0][0]
        FBv1 = res[1]
    return stress

```

Additional Code to "Polymer\_Mechanics\_Chap09.py":

```

def BB_timeDer_3D(Fv, t, params, time0, time1, F0, F1):
    """Returns FvDot"""
    mu, lamL, kappa, s, xi, C, tauBase, m, tauCut = params[:9]
    F = F0 + (t-time0) / (time1-time0) * (F1-F0)
    Fe = F / Fv
    Stress = toVec(EC_3D(Fe, [s*mu, lamL, kappa]))
    devStress = Stress - sum(Stress)/3
    tau = norm(devStress)
    lamCh = sqrt(sum(Fv*Fv)/3.0)
    lamFac = lamCh - 1.0 + xi
    gamDot = lamFac**C * (ramp(tau/tauBase-tauCut)**m)
    prefac = 0.0
    if tau > 0: prefac = gamDot / tau
    FeInv = array([1.0, 1.0, 1.0]) / Fe
    FvDot = prefac * (FeInv * devStress * F)
    return FvDot

def BB_3D(F0, F1, FBv0, time0, time1, params):
    """BB-model. 3D loading specified by principal stretches.
    params: [muA, lamL, kappa, s, xi, C, tauHat, m, tauCut].
    Returns: (true stress, FBv1)"""
    muA, lamL, kappa, s = params[:4]
    StressA = toVec(EC_3D(F1, [muA, lamL, kappa]))
    FBv1 = scipy.integrate.odeint(BB_timeDer_3D, FBv0, \
        [time0, time1], args=(params, time0, time1, F0, F1))[1]
    FBe1 = F1 / FBv1
    StressB = toVec(EC_3D(FBe1, [s*muA, lamL, kappa]))
    Stress = StressA + StressB
    return (Stress, FBv1)

```

The actual commands that sets everything up and calls the main function are listed in the file `BB_Compressible_Uniaxial.py`.

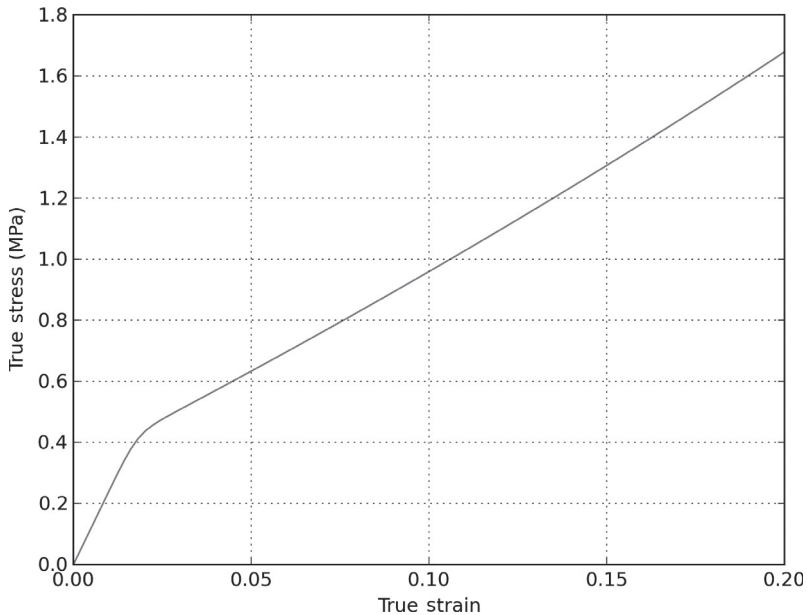
Code "BB\_Compressible\_Uniaxial.py":

```
from Polymer_Mechanics_Chap05 import *
from Polymer_Mechanics_Chap09 import *

N = 100
timeVec = linspace(0, 10.0, N)
trueStrain = linspace(0, 0.2, N)
params = [2.0, 3.5, 500.0, 3.0, 0.05, -0.5, 0.5, 8.0, 0.01]
trueStress = uniaxial_stress_visco(BB_3D, timeVec, trueStrain, \
    params)

plot(trueStrain, trueStress, 'r-', label='Python Calculation')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
show()
```

The results from running this code are shown in [Figure 8.10](#).



**Figure 8.10** Predictions from the Python file `BB_Compressible_Uniaxial.py`.

### 8.2.3 Generic Numerical Implementation

The theory for the BB model, presented above, used the viscoelastic deformation gradient ( $\mathbf{F}_B^V$ ) as a state variable for amount of

viscoelastic flow. This is not the only possible state variable. In fact, you can write the evolution equation for the viscoplastic deformation in many different and alternative ways:

$$\dot{\mathbf{F}}_v = \frac{\dot{\gamma}}{||\text{dev}[\mathbf{b}_e^*]||} \mathbf{F}_v^{-1} \text{dev}[\mathbf{b}_e^*] \mathbf{F}_v, \quad (8.21)$$

$$\dot{\mathbf{C}}_v^* = \frac{2\dot{\gamma}}{||\text{dev}[\mathbf{b}_e^*]||} \left[ \mathbf{C}^* - \frac{1}{3} \text{tr}[\mathbf{b}_e^*] \mathbf{C}_v^* \right], \quad (8.22)$$

$$\dot{\mathbf{E}}_v = \frac{\dot{\gamma}}{||\text{dev}[\mathbf{b}_e^*]||} \left[ \mathbf{C} - \frac{1}{3} \text{tr}[\mathbf{b}_e^*] \mathbf{C}_v \right], \quad (8.23)$$

$$\frac{d}{dt} \left[ \mathbf{F}_v^{-1} \right] = -\dot{\gamma} \mathbf{F}_v^{-1} \frac{\text{dev}[\mathbf{F}_v^{-\top} \mathbf{U}^{*2} \mathbf{F}_v^{-1}]}{||\text{dev}[\mathbf{b}_e^*]||}, \quad (8.24)$$

$$\frac{d}{dt} \left[ \mathbf{c}_v^{-1} \right] = \frac{-2\dot{\gamma}}{||\text{dev}[\mathbf{b}_e^*]||} \left( \mathbf{c}_v^{-1} \mathbf{U}^{*2} - \overline{\lambda_e^{*2}} \mathbf{I} \right) \mathbf{c}_v^{-1}. \quad (8.25)$$

In these equations  $\mathbf{C}_v^*$  is the viscoelastic right Cauchy-Green tensor,  $\mathbf{E}_v$  is the Green strain,  $\mathbf{F}_v^{-1}$  is the inverse of the viscoelastic deformation gradient, and  $\mathbf{C}_v^{-1}$  is called the Finger tensor. All of these equations are equally valid, and the most appropriate one can be selected based on personal preference and the desired reference frame to use for the calculations.

### 8.2.4 Dynamic Loading Predictions

The dynamic response of elastomers and other soft materials is an important characteristic in many industrial applications. One common way to experimentally measure the dynamic response is to apply a sinusoidal strain with a constant strain amplitude ( $\varepsilon_a$ ) and a constant mean strain ( $\varepsilon_m$ ):

$$\varepsilon(t) = \varepsilon_m + \varepsilon_a \sin(\omega t). \quad (8.26)$$

If the strains are sufficiently small then the measured stress response will also be sinusoidal, but it will be out-of-phase with the applied strain:

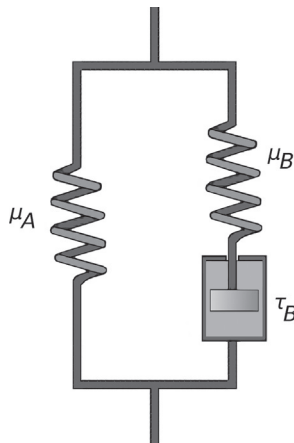
$$\sigma(t) = \sigma_m + \sigma_a \sin(\omega t + \delta), \quad (8.27)$$

$$= \sigma_m + \sigma_a E' \sin(\omega t) + \sigma_a E'' \cos(\omega t). \quad (8.28)$$

From this the storage modulus ( $E'$ ) and the loss modulus ( $E''$ ) can be defined from:

$$\sigma(t) = \sigma_m + \varepsilon_a E' \sin(\omega t) + \varepsilon_a E'' \cos(\omega t). \quad (8.29)$$

Experimentally, the storage and loss moduli of most elastomers depend on both the applied strain amplitude and the applied frequency. It is therefore often desirable to have a material model that agrees with this experimental fact. If we define a *Linear Flow Model* as a two-network model where Network A consists of a Neo-Hookean spring, and Network B consists of a Neo-Hookean spring in series with a linear dashpot (see the following figure).

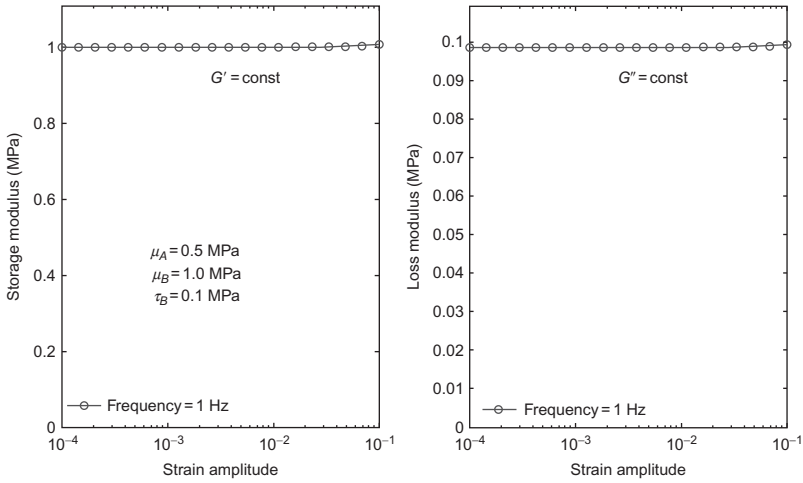


The flow of the linear dashpot is given by

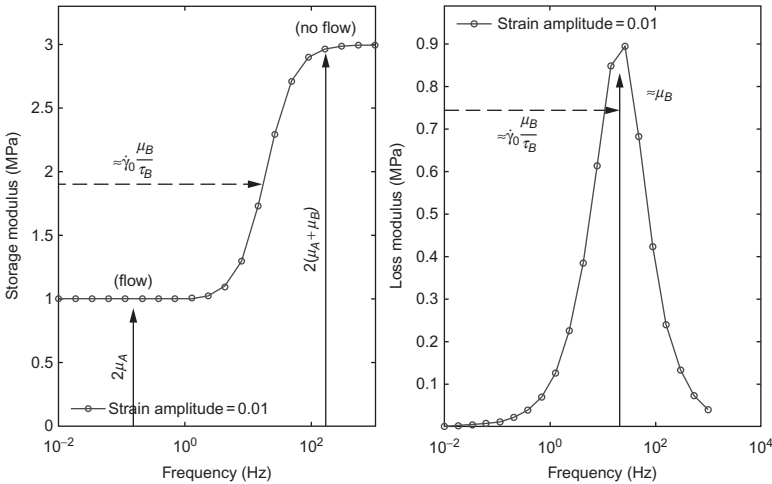
$$\dot{\gamma}_B = \dot{\gamma}_0 \left[ \frac{\tau}{\tau_B} \right], \quad (8.30)$$

where  $\dot{\gamma}_0 \equiv 0$ . For this model the characteristic relaxation time is given by:  $\tau_B / (\dot{\gamma}_0 \mu_B)$ , and the characteristic frequency is given by:  $\dot{\gamma}_0 \mu_B / \tau_B$ .

The predicted response for a strain amplitude sweep for this model is shown in [Figure 8.11](#). As expected, the predicted storage



**Figure 8.11** Predictions storage and loss moduli for a linear flow model when exposed to a strain amplitude sweep.



**Figure 8.12** Predictions storage and loss moduli for a linear flow model when exposed to a frequency sweep.

and loss moduli are both finite, but independent of the applied strain amplitude.

The predicted response for a frequency sweep for the linear flow model is shown in Figure 8.12. Similar to what was shown for linear viscoelasticity, the storage modulus increases with

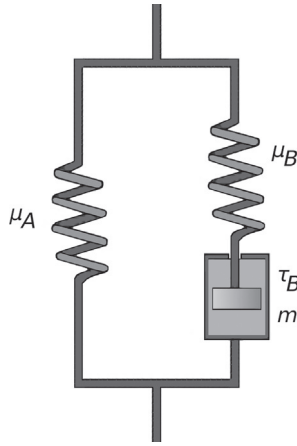
frequency, and the loss modulus has a peak at a characteristic frequency.

As is shown in [Figures 8.11](#) and [8.12](#), the linear flow model is not sufficiently advanced to capture the true viscoelastic response of many polymers that also depend on the strain amplitude.

An extension of this model, called the *Power Flow Model*, is exactly as the linear flow model except that the viscoelastic flow rate is given by:

$$\dot{\gamma}_B = \dot{\gamma}_0 \left[ \frac{\tau}{\tau_B} \right]^m. \quad (8.31)$$

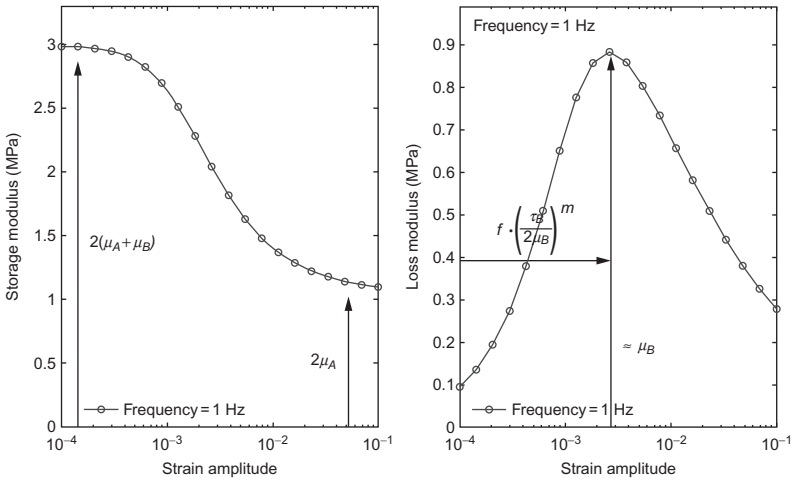
This equation can be represented in the following rheological representation.



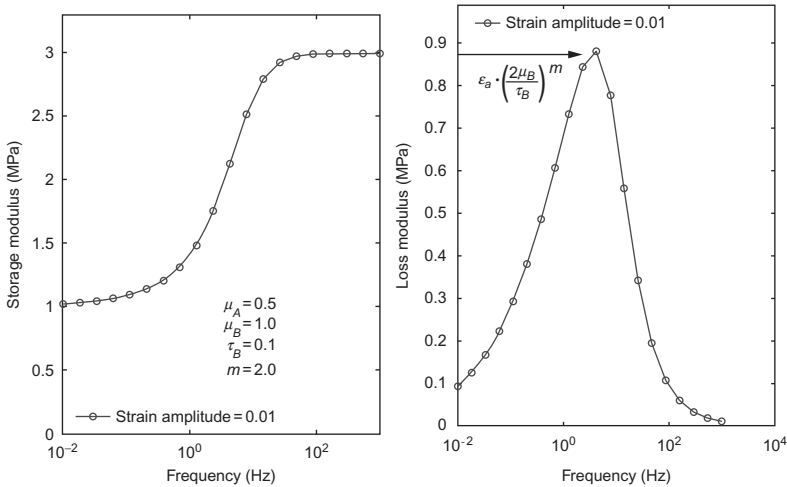
This model is a slightly simplified version of the BB model.

The predicted response for a strain amplitude sweep for this model is shown in [Figure 8.13](#). The predicted storage modulus decreases with increasing strain amplitude, this is what is called the *Payne effect*. Similarly, the loss modulus goes through a peak for a given strain amplitude. Both of these behaviors are consistent with what is experienced by typical materials.

The predicted response for a frequency sweep for the power flow model is shown in [Figure 8.14](#). Similar to what was shown for the linear flow model, the storage modulus increases with



**Figure 8.13** Predicted storage and loss moduli for a power flow model when exposed to a strain amplitude sweep.



**Figure 8.14** Predicted storage and loss moduli for a power flow model when exposed to a frequency sweep.

frequency, and the loss modulus has a peak at a characteristic frequency.

As is shown in [Figures 8.13](#) and [8.14](#), the power flow model is significantly more accurate compared to the linear flow model when it comes to predicting both the strain amplitude, and fre-

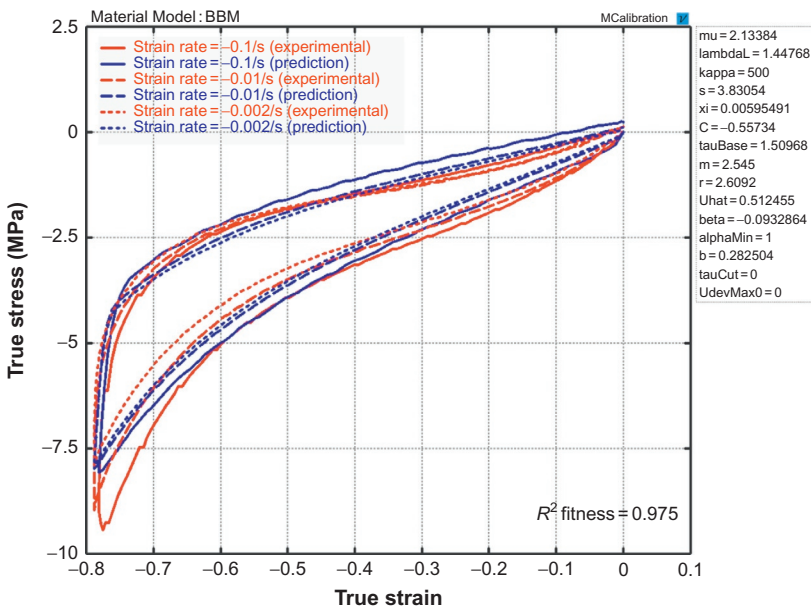


quency dependence. The power flow model (and therefore also the BB model) qualitatively captures all essential dynamic behaviors of elastomers. To capture the response also quantitatively can require multiple parallel networks of the type used in the Power Flow Model. Each network contributing a portion of the strain amplitude and frequency spectrum.

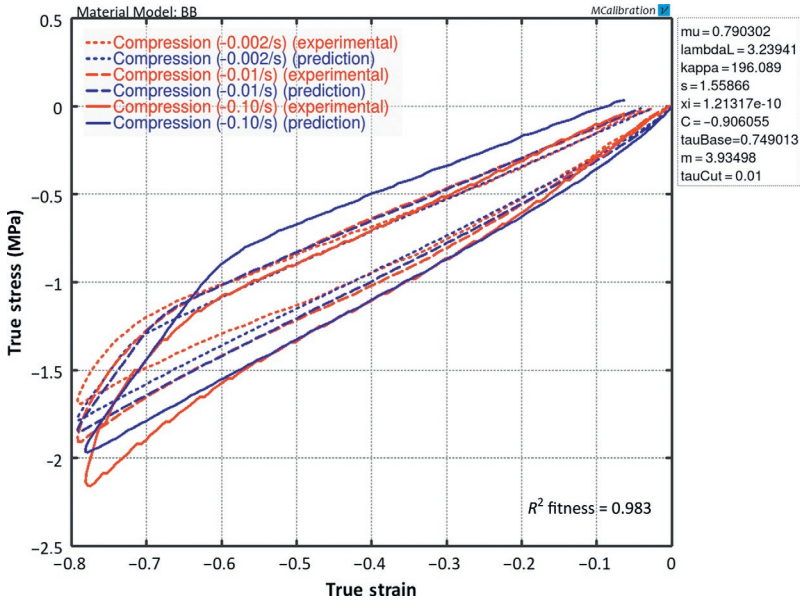
## 8.2.5 Use of the BB-Model for Polymer Modeling

The BB model is very good at predicting the large-strain time-dependent behavior of isotropic elastomer-like materials. The model is particularly suitable for predicting large-strain, cyclic, stress relaxation, and creep behaviors.

Figure 8.15 shows experimental data for a chloroprene rubber with 25 vol% carbon black. The experimental data consists of loading-unloading cycles at three different strain rates.



**Figure 8.15** Comparison between experimental data for a chloroprene rubber with 25 vol% carbon black, and predictions from the BB-model with Mullins softening.



**Figure 8.16** Comparison between experimental data for a chloroprene rubber with 7 vol% carbon black, and predictions from the BB-model.

The experiments were performed at room temperature. In the figure is also shown the best fit of the BB-model. It is shown that the model accurately captures the loading-unloading behavior at three strain rates.

Another example, this time for a chloroprene rubber with 7% volume is shown in [Figure 8.16](#).

Also in this case the material was tested in uniaxial compression at three different strain rates. The BB model was calibrated to the experimental data and the model predictions are indicated in the figure. The figure shows that the BB model captures the experimentally observed behavior.

### 8.3 Arruda-Boyce Model

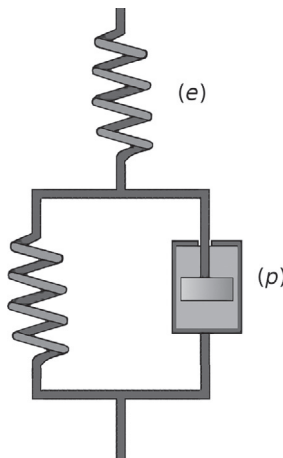
One of the first advanced models for predicting the response of glassy polymers is the AB model [16, 20, 21]. This model is not the same as the AB eight-chain model which is a hyperelastic model discussed in Section 5.3.10 of Chapter 5. This AB viscoplasticity model is interesting due to its simplicity

and structure, that is also used in more advanced models for thermoplastics.<sup>1</sup>

The AB model was developed [16, 20, 21] for predicting the large strain, time- and temperature-dependent response of glassy polymers. The behavior of this class of materials when subjected to gradually increasing loads is typically characterized by an initial linear elastic response followed by yielding and then strain hardening at large deformations.<sup>2</sup> This evolution in material response with applied loads is directly incorporated into the AB model.

In the AB framework, the total deformation gradient is decomposed into elastic and plastic components,  $\mathbf{F} = \mathbf{F}^e \mathbf{F}^p$ . As is shown in the one-dimensional rheological representation in Figure 8.17, this decomposition can be interpreted as two networks acting in series: one elastic network ( $e$ ) and one plastic network ( $p$ ).

Using this decomposition of the deformation gradient the Cauchy stress can be calculated from the linear elastic relationship:



**Figure 8.17** Rheological representation of the Arruda-Boyce model.

<sup>1</sup>The AB model is available in the PolyUMod library [22].

<sup>2</sup>Some thermoplastics, for example LDPE, does not have a linear elastic response even at very small strains. The material behaves in a non-linear viscoelastic manner even at small strain.

$$\mathbf{T} = \frac{1}{J^e} (2\mu^e \mathbf{E}^e + \lambda \operatorname{tr}[\mathbf{E}^e] \mathbf{I}), \quad (8.32)$$

where  $\mathbf{E}^e = \ln[\mathbf{V}^e]$  is the logarithmic true strain,  $J^e = \det[\mathbf{F}^e]$ , and  $\mu^e, \lambda^e$  are Lamé's constants. The stress driving the plastic flow is given by the tensorial difference between the total stress and the convected back stress

$$\mathbf{T}^* = \mathbf{T} - \frac{1}{J^e} \mathbf{F}^e \mathbf{T}^p (\mathbf{F}^e)^\top, \quad (8.33)$$

where the deviatoric back stress is given by the incompressible eight-chain model which can be written:

$$\mathbf{T}^p = \frac{\mu^p}{\bar{\lambda}^p} \frac{\mathcal{L}^{-1}(\bar{\lambda}^p / \lambda_{\text{lock}}^p)}{\mathcal{L}^{-1}(1 / \lambda_{\text{lock}}^p)} \operatorname{dev}[\mathbf{b}^p] \quad (8.34)$$

with  $\mu^p, \lambda_{\text{lock}}^p$  being physically motivated material constants,  $\mathbf{b}^p = \mathbf{F}^p (\mathbf{F}^p)^\top$ ,  $\bar{\lambda}^p = (\operatorname{tr}[\mathbf{b}^p] / 3)^{1/2}$  the effective chain stretch based on the eight-chain topology assumption, and  $\mathcal{L}^{-1}(x)$  the inverse Langevin function.

In the original work [16, 20, 21] the plastic flow rate was given by:

$$\dot{\gamma}^p = \dot{\gamma}_0 \exp \left[ -\frac{As}{k_B \theta} \left( 1 - \left( \frac{\tau}{s} \right)^{5/6} \right) \right], \quad (8.35)$$

where  $\dot{\gamma}_0, A, s$  are material constants,  $k_B$  is Boltzmann's constant, and  $\theta$  is the absolute temperature. It has been shown by Hasan and Boyce [23] that the difference in behavior between a stress exponent of 5/6 and 1 is very small. By taking the stress exponent to be 1 and grouping material constants together the expression for the plastic flow rate can be simplified to

$$\dot{\gamma}^p = \dot{\gamma}_i \exp \left[ \frac{\tau}{\tau_{\text{base}}} \right], \quad (8.36)$$

where  $\dot{\gamma}_i$  and  $\tau_{\text{base}}$  are material parameters. The focus of the current work is on isothermal deformation histories, to explicitly include temperature effects the parameter  $\tau_{\text{base}}$  can be replaced by  $k_B \theta / A$ . The scalar equivalent stress  $\tau$  is here taken as the Frobenius norm of the deviatoric part of the driving stress

$\tau = \|\text{dev}[\mathbf{T}^*]\|_F$ , where  $\|\mathbf{A}\|_F \equiv (A_{ij}A_{ij})^{1/2}$ . The rate of plastic deformation is given by

$$\mathbf{D}^P = \frac{\dot{\gamma}^P}{\tau} \text{dev}[\mathbf{T}^*] \quad (8.37)$$

and the plastic spin is taken to be zero [16], i.e.  $\mathbf{W}^P = 0$ , which uniquely specifies the rate kinematics. The time-derivative of  $\dot{\mathbf{F}}$  is given by

$$\dot{\mathbf{F}}^P = \mathbf{D}^P \mathbf{F}^P = \frac{\dot{\gamma}^P}{\tau} \text{dev}[\mathbf{T}^*] \mathbf{F}^P. \quad (8.38)$$

Note that the original Boyce model also allows for modeling of strain softening through an evolution equation of the athermal shear resistance,  $s$ :

$$\dot{s} = h \left[ 1 - \frac{s}{s_{ss}} \right] \dot{\gamma}^P, \quad (8.39)$$

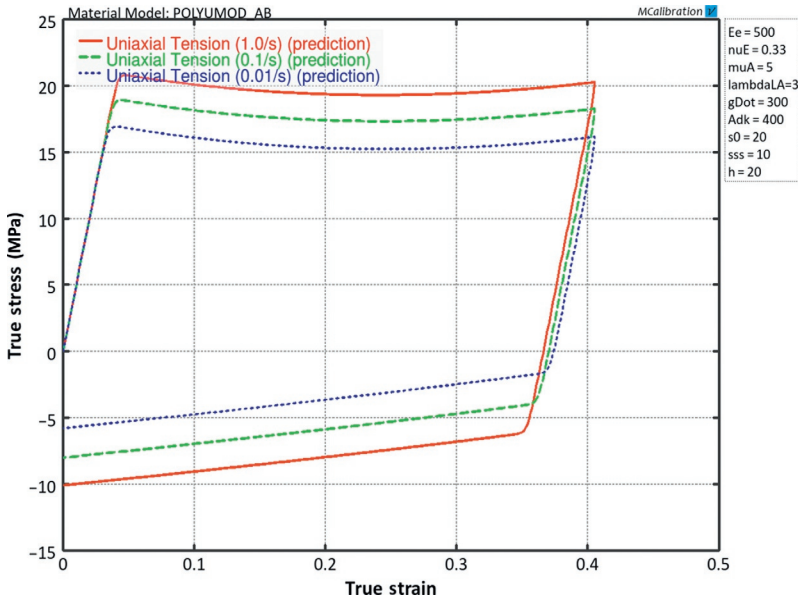
where  $h$  and  $s_{ss}$  are material parameters. This equation can also be written in terms of the effective shear strength:

$$\dot{\tau}_{\text{base}} = h \left[ 1 - \frac{\tau_{\text{base}}}{\tau_{\text{base.ss}}} \right] \dot{\gamma}^P, \quad (8.40)$$

where  $\tau_{\text{base},0}$  is the initial flow resistance, and  $\tau_{\text{base.ss}}$  is the steady-state flow resistance at large strains.

Figure 8.17 shows one exemplar set of predictions from the AB-model. In this case, the response in uniaxial tension followed by unloading is shown at three different strain rates. In this example the flow resistance evolves from an initial value of 20 MPa down to 10 MPa at steady state (Figure 8.18).

In summary, the AB model is based on a simple theoretical framework, and as a result is easy to use but not always as accurate as needed for analysis of engineering problems.



**Figure 8.18** Stress-strain predictions from the Arruda-Boyce model.

## 8.4 Dual Network Fluoropolymer Model

The dual network fluoropolymer (DNF) model is an advanced material model capable of predicting the large-strain, time- and temperature-dependent viscoplastic behavior of various types of fluoropolymers and other types of thermoplastics.<sup>3</sup>

Fluoropolymers, as well as other thermoplastics, exhibit a complicated non-linear response when subjected to external loads. At small applied strains, the material response is typically linear viscoelastic. At larger strains, the material undergoes distributed yielding that evolves with plastic strain, followed by large-scale viscoplastic flow, and finally, gradual material stiffening at large strain until ultimate failure occurs. It is also known that the material response is strongly dependent on applied strain-rate and temperature: higher strain rates and lower temperatures increase the stiffness of the material. The DNF model is a constitutive model aimed at predicting these experimentally observed char-

<sup>3</sup>The DNF model is available in the PolyUMod library [22].

acteristics. The proposed model is an extension of previous work by Bergström and Boyce [1, 4, 5] and Arruda and Boyce [21] for elastomers and glassy polymers.

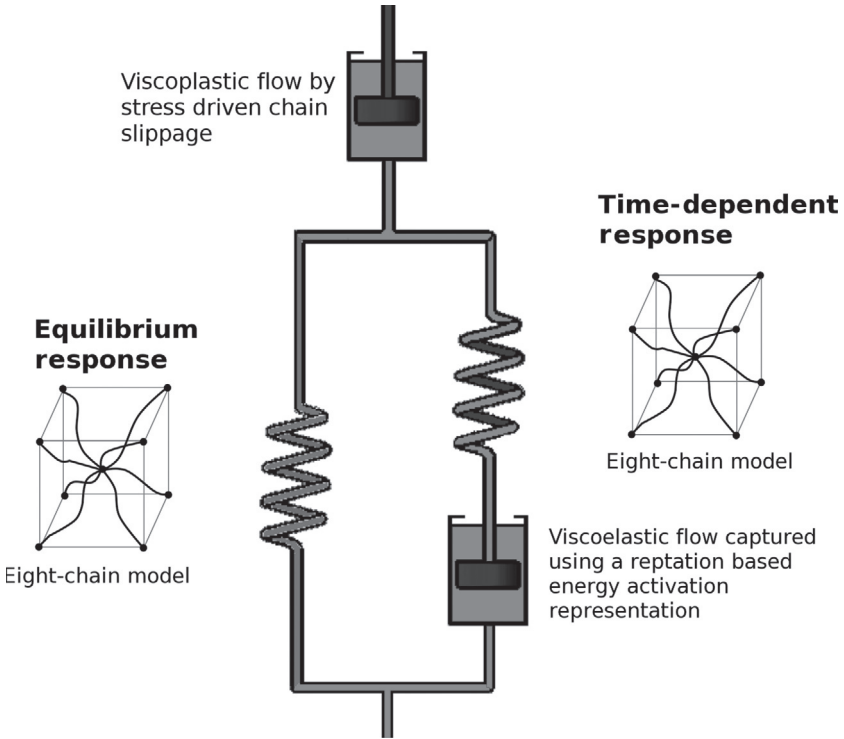
There are a number of different candidate material models that are documented in the literature for predicting the behavior of general thermoplastics (e.g. [21, 24–26]), and some of these models are summarized in this chapter. Development of advanced constitutive models for polymers is an active area of research that is evolving and improving. Up until about year 2000, there were no constitutive models specifically developed for fluoropolymers, and the most useful models were either classical isotropic plasticity, linear viscoelasticity models, or general models for thermoplastics [4, 21]. Since then constitutive models specifically developed for fluoropolymers have emerged [27, 28]. These models are typically better at predicting the experimentally observed characteristics of fluoropolymers than traditional isotropic plasticity or viscoelasticity models, but have limitations of isothermal conditions only.

The DNF model can be represented using the rheological model shown in Figure 8.19.

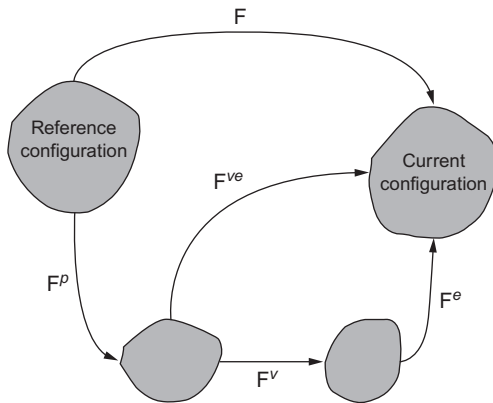
The DNF model incorporates experimental characteristics by using a decomposition of the material behavior into a viscoplastic response, corresponding to irreversible molecular chain sliding (due to the lack of chemical crosslinks in the material) and a time-dependent viscoelastic response. The viscoelastic response is further decomposed into the response of two molecular networks acting in parallel: a first network (*A*) captures the equilibrium (long term) of the viscoelastic response and a second network (*B*) the time-dependent (short term) deviation from the viscoelastic equilibrium state. A schematic illustrating the kinematics of deformation are shown in Figure 8.20.

The total deformation gradient  $\mathbf{F}^{\text{appl}}$  contains both a thermal expansion part  $\mathbf{F}^{\text{th}} = [1 + \alpha(\theta - \theta_0)] \mathbf{I}$ , and a mechanical deformation part  $\mathbf{F}$ :

$$\mathbf{F}^{\text{appl}} = \mathbf{F} \mathbf{F}^{\text{th}}.$$



**Figure 8.19** Structure of the dual network fluoropolymer model.



**Figure 8.20** Kinematics of deformation for the DNF model.



The deformation gradient  $\mathbf{F}$  is multiplicatively decomposed into viscoplastic and viscoelastic components:

$$\mathbf{F} = \mathbf{F}^{ve} \mathbf{F}^p. \quad (8.41)$$

The Cauchy stress acting on network  $A$  is given by the eight-chain representation [4, 19]:

$$\boldsymbol{\sigma}^{ve} = \mathbf{f}_{8\text{ch}}(\mathbf{F}^{ve}) = \frac{\mu_A^0(\theta)}{J^{ve} \bar{\lambda}^{*}} \frac{\mathcal{L}^{-1}(\bar{\lambda}^{ve*}/\lambda^{\text{lock}})}{\mathcal{L}^{-1}(1/\lambda^{\text{lock}})} \text{dev}[\mathbf{B}^{ve*}] + \frac{\kappa \ln[J^{ve}]}{J^{ve}} \mathbf{1}, \quad (8.42)$$

where  $J^{ve} = \det[\mathbf{F}^{ve}]$ ,  $\mu_A^0(\theta)$  is a temperature-dependent initial shear modulus,  $\lambda^{\text{lock}}$  is the chain locking stretch,  $\mathbf{b}^{ve*} = (J^{ve})^{-2/3} \mathbf{F}^{ve} (\mathbf{F}^{ve})^T$  is the Cauchy-Green deformation tensor,  $\bar{\lambda}^{ve*} = (\text{tr}[\mathbf{b}^{ve*}]/3)^{1/2}$  is the effective chain stretch based on the eight-chain topology assumption [19],  $\mathcal{L}^{-1}(x)$  is the inverse Langevin function, where  $\mathcal{L}(x) = \coth(x) - 1/x$ , and  $\kappa$  is the bulk modulus. By explicitly incorporating the temperature-dependence of the shear modulus it is possible to capture the stiffness variation of the material over a range of temperatures. The following expression is found to accurately capture the experimentally observed temperature dependence of the shear modulus for temperatures between 20 °C and 200 °C:

$$\mu_A(\theta) = \mu_A^0 \exp\left[\frac{\theta_0 - \theta}{\theta_{\text{base}}}\right], \quad (8.43)$$

where  $\theta$  is the current temperature, and  $\mu_A^0$ ,  $\theta_0$ , and  $\theta_{\text{base}}$  are material parameters.

The viscoelastic deformation gradient acting on network  $B$  is decomposed into elastic and viscous parts:

$$\mathbf{F}^{ve} = \mathbf{F}^e \mathbf{F}^v. \quad (8.44)$$

The Cauchy stress acting on network  $B$  is obtained from the same eight-chain network representation that was used for network  $A$ . For simplicity, the response of network  $B$  is taken as a scalar factor  $s_B$  (a specified material parameter) times the eight-chain ex-

pression that was used for network  $A$  applied on the deformation gradient  $\mathbf{F}^e$ :

$$\boldsymbol{\sigma}^e = s_B \cdot \mathbf{f}_{8\text{ch}}(\mathbf{F}^e), \quad (8.45)$$

where  $\mathbf{f}_{8\text{ch}}(\cdot)$  is the tensorial function defined in Equation (8.42). Using this framework, the total Cauchy stress in the system is given by  $\boldsymbol{\sigma} = \boldsymbol{\sigma}^{ve} + \boldsymbol{\sigma}^e$ .

The total velocity gradient of network  $B$ ,  $\mathbf{L}^{ve} = \dot{\mathbf{F}}^{ve}(\mathbf{F}^{ve})^{-1}$ , can similarly be decomposed into elastic and viscous components:  $\mathbf{L}^{ve} = \mathbf{L}^e + \mathbf{F}^e \mathbf{L}^e \mathbf{F}^{e-1} = \mathbf{L}^e + \tilde{\mathbf{L}}^v$ , where  $\mathbf{L}^v = \dot{\mathbf{F}}^v \mathbf{F}^{v-1} = \mathbf{D}^v + \mathbf{W}^v$  and  $\tilde{\mathbf{L}}^v = \mathbf{D}^v + \tilde{\mathbf{W}}^v$ . The unloading process relating the deformed state with the intermediate state is not uniquely defined, since an arbitrary rigid body rotation of the intermediate state still leaves the state stress free. The intermediate state can be made unique in different ways [16], one particularly convenient way that is used here is to prescribe  $\tilde{\mathbf{W}}^v = \mathbf{0}$ . This will, in general, result in elastic and inelastic deformation gradients both containing rotations.

The rate of viscoplastic flow of network  $B$  is constitutively prescribed by

$$\tilde{\mathbf{D}}^v = \dot{\gamma}_{\text{dev}}^v \mathbf{N}_{\text{dev}}^v + \dot{\gamma}_{\text{vol}}^v \mathbf{N}_{\text{vol}}^v, \quad (8.46)$$

where the first term gives the *deviatoric* viscoelastic flow and the second term gives the *volumetric* viscoelastic flow. The tensors  $\mathbf{N}_{\text{dev}}^v$  and  $\mathbf{N}_{\text{vol}}^v$  specify the directions of the driving deviatoric and volumetric stresses of the relaxed configuration convected to the current configuration, and the terms  $\dot{\gamma}_{\text{dev}}^v$  and  $\dot{\gamma}_{\text{vol}}^v$  specify the effective deviatoric and volumetric flow rates. Noting that  $\boldsymbol{\sigma}^e$  is computed in the loaded configuration, the driving deviatoric stress on the relaxed configuration convected to the current configuration is given by  $\boldsymbol{\sigma}^{e'} = \text{dev}[\boldsymbol{\sigma}^e]$ , and by defining an effective stress by the Frobenius norm  $\tau^e = \|\boldsymbol{\sigma}^{e'}\|_F \equiv (\text{tr}[\boldsymbol{\sigma}^{e'} \boldsymbol{\sigma}^{e'}])^{1/2}$ , the direction of the driving deviatoric stress becomes  $\mathbf{N}_{\text{dev}}^v = \boldsymbol{\sigma}^{e'} / \tau^e$ . The effective deviatoric flow rate is given by the reptation-inspired equation [4]:

$$\dot{\gamma}_{\text{dev}}^v = \dot{\gamma}_0 [\bar{\lambda}^v - 1 + \xi]^C \cdot \left( \frac{\tau^e}{\tau_{\text{base}} + \beta R(p^e)} \right)^m \cdot \left( \frac{\theta}{\theta_{\text{base}}} \right)^n, \quad (8.47)$$

where  $\bar{\lambda}^v = \sqrt{\text{tr}[\mathbf{B}^{v*}]/3}$  is an effective viscoelastic chain stretch,  $\mathbf{B}^{v*} = (J^v)^{-2/3} \mathbf{F}^v (\mathbf{F}^v)^T$  is the Cauchy-Green deformation tensor,  $R(\cdot)$  is the ramp function,  $p^e = -(\sigma_{11}^e + \sigma_{22}^e + \sigma_{33}^e)/3$  is the hydrostatic pressure,  $\dot{\gamma}_0$  is a constant taken as 1/s (1 reciprocal second) that is needed for dimensional consistency, and  $C, \beta, m, \tau_{\text{base}}, n,$  and  $\theta_{\text{base}}$  are specified material parameters.

In this framework, the temperature dependence of the flow rate is taken to follow a power law form. Due to the high bulk modulus of PTFE the effective volumetric flow rate is small and is here simply represented with a constant viscosity  $\eta_{\text{vol}}$ :

$$\dot{\gamma}_{\text{vol}}^v = -p^e / \eta_{\text{vol}}. \quad (8.48)$$

In summary, the velocity gradient of the viscoelastic flow can be written

$$\dot{\mathbf{F}}^v = \mathbf{F}^{e-1} \left( \dot{\gamma}_{\text{dev}}^v \frac{\text{dev}[\boldsymbol{\sigma}^e]}{\tau^e} + \dot{\gamma}_{\text{vol}}^v \mathbf{I} \right) \mathbf{F}^{ve}. \quad (8.49)$$

The rate of plastic flow is captured by a simple phenomenological representation:

$$\dot{\gamma}^p = \begin{cases} ab(\epsilon - \epsilon_0)^{b-1} \dot{\epsilon} & \text{if } \tau > \sigma_0, \\ 0 & \text{otherwise,} \end{cases} \quad (8.50)$$

where  $a > 0, b > 0$  and  $\sigma_0 > 0$  are material parameters,  $\tau = \|\text{dev}[\boldsymbol{\sigma}]\|_F$  is the Frobenius norm of the deviatoric portion of the Cauchy stress  $\boldsymbol{\sigma}$ , and  $\epsilon_0$  is the effective strain at which  $\tau$  becomes equal to  $\sigma_0$ . The effective strain in Equation (8.50) is obtained from  $\epsilon = \|\mathbf{E}_{\text{ln}}\|_F$ , where  $\mathbf{E}_{\text{ln}} = \ln[\mathbf{V}]$  is the logarithmic strain, and  $\dot{\epsilon}$  is the rate of change of the effective strain. The key feature of Equation (8.50) is that it predicts the rate of plastic flow to be proportional to the applied strain rate and the magnitude of the current strain. By inserting  $\mathbf{F} = \mathbf{F}^{ve} \mathbf{F}^p$  into  $\mathbf{L} = \dot{\mathbf{F}} \mathbf{F}^{-1}$ , the expression for the velocity gradient can be expanded to

$\mathbf{L} = \mathbf{L}^{ve} + \mathbf{F}^{ve} \mathbf{L}^p \mathbf{F}^{ve} = \mathbf{L}^{ve} \tilde{\mathbf{L}}^p$ . By taking  $\tilde{\mathbf{W}}^p = \mathbf{0}$ , the viscoplastic velocity gradient can be written  $\tilde{\mathbf{L}}^p = \dot{\gamma}^p \text{dev}[\boldsymbol{\sigma}]/\tau$  giving

$$\dot{\mathbf{F}}^p = \dot{\gamma}^p \mathbf{F}^{ve-1} \frac{\text{dev}[\boldsymbol{\sigma}]}{\tau} \mathbf{F}, \quad (8.51)$$

specifying the rate kinematics of the plastic flow.

The DNF model requires the material parameters in [Table 8.1](#).

**Table 8.1** Material Parameters Used by the DNF Model

Index	Symbol	Description
1	$\mu_A^0$	Shear modulus of network <i>A</i>
2	$\theta_0$	Reference temperature
3	$\theta_{\text{base}}$	Temperature factor
4	$\lambda^{\text{lock}}$	Locking stretch
5	$\kappa$	Bulk modulus
6	$s_B$	Relative stiffness of network <i>B</i>
7	$\xi$	Strain adjustment factor
8	$C$	Strain exponential
9	$\tau_{\text{base}}$	Flow resistance
10	$\beta$	Pressure dependence of flow
11	$m$	Stress exponential
12	$n$	Temperature exponential
13	$\eta_{\text{vol}}$	Volumetric flow viscosity
14	$a$	Plastic flow ratio
15	$b$	Plastic flow exponent
16	$\sigma_0$	Plastic flow strength
17	$\alpha$	Thermal expansion coefficient

### 8.4.1 Matlab Implementation of the DNF Model

The DNF material model is formulated as a set of differential equations that need to be solved for each time increment. One way to solve this set of equations is to use the following algorithm:

1. Known values at time  $t_i$ :
  - Deformation gradient:  $\mathbf{F}$
  - State variables:  $\mathbf{F}^v, \mathbf{F}^p$
2. Known values at time  $t_{i+1}$ :
  - Deformation gradient:  $\mathbf{F}$
3. Calculate  $\mathbf{F}^v$  at  $t_{i+1}$  using an ODE solver and Equation (8.49) for  $\dot{\mathbf{F}}^v$ .
4. Calculate  $\mathbf{F}^p$  at  $t_{i+1}$  using an ODE solver and Equation (8.51) for  $\dot{\mathbf{F}}^p$ .
5. Calculate  $\boldsymbol{\sigma}$  using Equation (8.42), and Equation (8.45) at  $t_{i+1}$ .

For the case of incompressible, uniaxial loading this algorithm can be implemented into Matlab code as shown in Figure 8.21.

Example code that can be used to calculate the stress-strain response is shown in Figure 8.22. The results from running this code are shown in Figure 8.23.

### 8.4.2 Use of the DNF Model for Polymer Modeling

One example of how the DNF model can be used to predict the response of a fluoropolymer with 15 vol% short glass fibers is shown in this section. The material model calibrated to the complete set of uniaxial tension and compression data, and all predictions shown in Figures 8.24–8.28 were obtained using the same set of material parameters. As is shown in these figures, the DNF model can accurately predict the characteristic material response.

Matlab File Name: mat\_DNF.m

```

function [stress] = mat_DNF(time, strain, params)
%mat_DNF Dual-Network Fluoropolymer Model
%Incompressible uniaxial loading (uses true strain & stress)
%Parameters: [muA lambdaL sB xi C tauBas beta m a b sigma0]
%State variables: [lambdaBv lambdaP]
muA = params(1); lambdaL = params(2); sB = params(3);
stress = 0 * strain;
state = [1 1]';
for i = 2 : length(strain)
    [t,stateAll] = ode45(@(t,y) flow(t, y, time, strain, params), time(i-1:i), state);
    state = stateAll(end,:);
    lambda = exp(strain(i));
    lambdaB = lambda / state(2);
    lambdaBe = lambdaB / state(1);
    stressA = mat_EC(time(i), log(lambdaB), [muA lambdaL]);
    stressB = mat_EC(time(i), log(lambdaBe), [sB*muA lambdaL]);
    stress(i) = stressA + stressB;
end
end

function res = flow(time, state, timeVec, strainVec, params)
% Calculates the time-derivative of the state variables
global eps0
muA = params( 1); lambdaL = params( 2); sB = params( 3);
xi = params( 4); C = params( 5); tauBas = params( 6);
beta = params( 7); m = params( 8); a = params( 9);
b = params(10); sigma0 = params(11);
lambda = exp( interp1(timeVec, strainVec, time) ); % stretch at the specified time
lambdaBv = state(1);
lambdaP = state(2);
lambdaB = lambda / state(2);
lambdaBe = lambdaB / lambdaBv;
% network B
lambdaBvChain = sqrt((lambdaBv^2 + 2 / lambdaBv) / 3);
stressB = mat_EC(time, log(lambdaBe), [sB*muA lambdaL]);
gammaBDot = (lambdaBvChain-1+xi)^C * (abs(stressB) / (tauBas+beta*stressB))^m;
res(1,1) = gammaBDot * sign(stressB) * lambdaBv;
% network P
strain = log(lambda);
dt = 1e-4; % simple estimation of the strain rate
strain1 = interp1(timeVec, strainVec, time+dt, 'linear', 'extrap');
strainDot = (strain1 - strain) / dt;
stressA = mat_EC(time, log(lambdaB), [muA lambdaL]);
stress = stressA + stressB;
if abs(stress) > sigma0
    if eps0 == 0
        eps0 = strain;
        res(2,1) = 0;
    else
        gamDotP = a*b*abs(strain - eps0)^(b-1) * strainDot;
        res(2,1) = gamDotP * sign(stress) * lambdaP;
    end
else
    eps0 = 0;
    res(2,1) = 0;
end
end
end

```

Figure 8.21 Matlab implementation of the DNF Model.

```

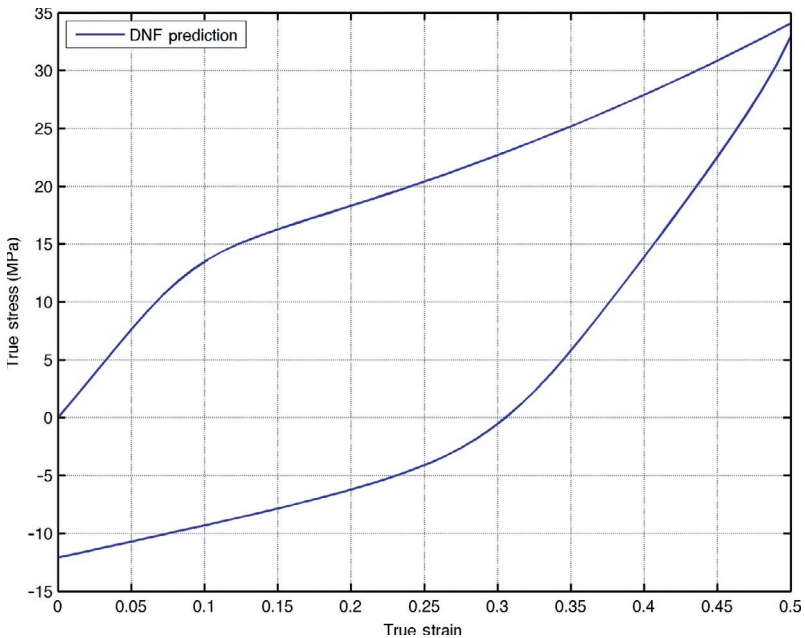
% Create the Dual Network Fluoropolymer Model prediction
time = linspace(0,2)';
strain = [linspace(0, 0.5,50)'; linspace(0.5, 0, 50)'];

muA = 10;
lambdaL = 3;
sB = 4;
xi = 0.05;
C = -0.5;
tauBas = 20;
beta = 0.00;
m = 4;
a = 0.02;
b = 1.10;
sigma0 = 10;
params = [muA lambdaL sB xi C tauBas beta m a b sigma0];

stress = mat_DNF(time, strain, params);

```

**Figure 8.22** Matlab test code for the DNF Model.



**Figure 8.23** Predicted stress-strain results from the Matlab implementation of the DNF model.

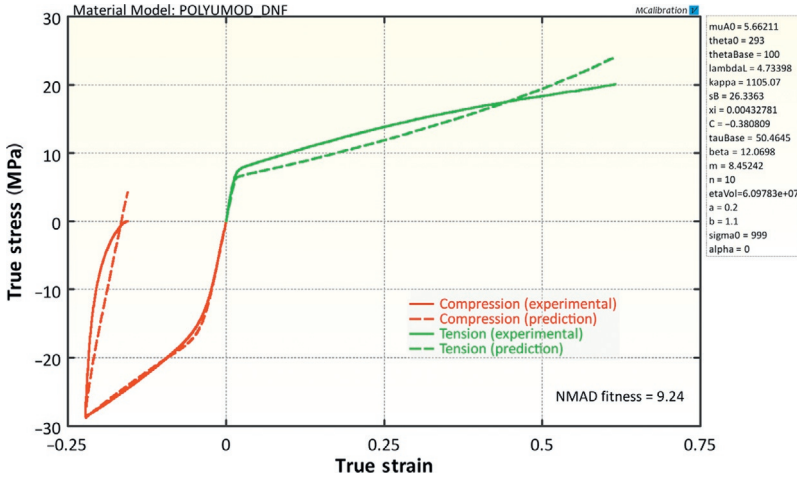


Figure 8.24 Comparison between experimental data in uniaxial tension and compression and model predictions from the DNF model.

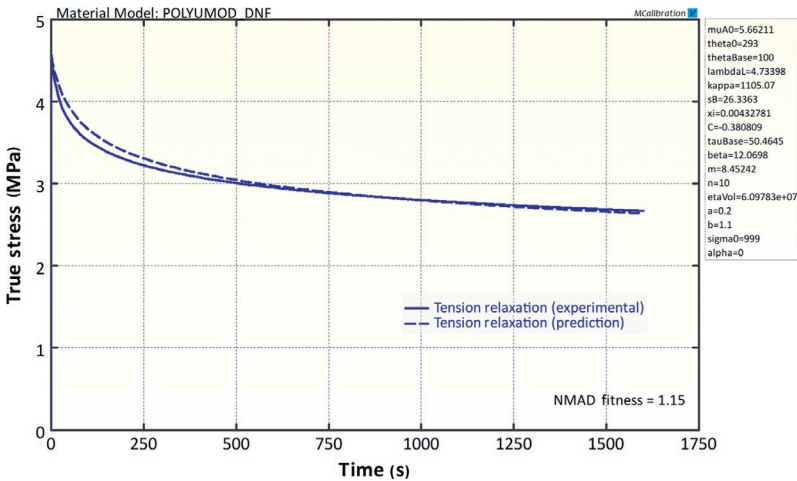
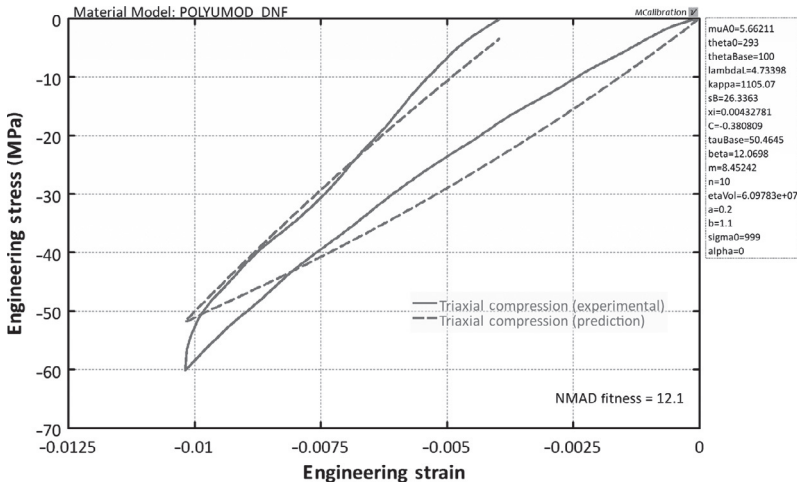
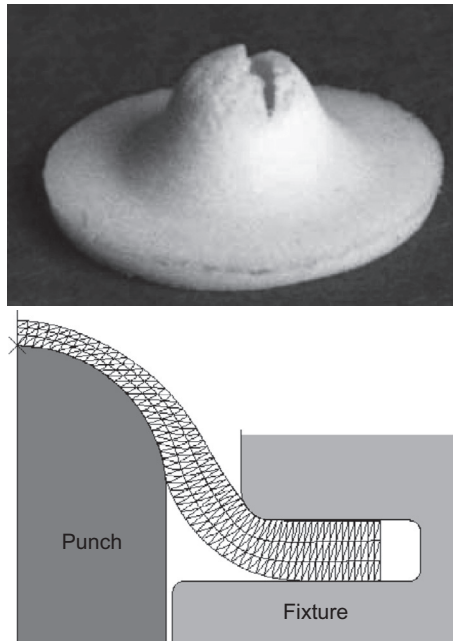


Figure 8.25 Comparison between experimental stress relaxation data and model predictions from the DNF model.

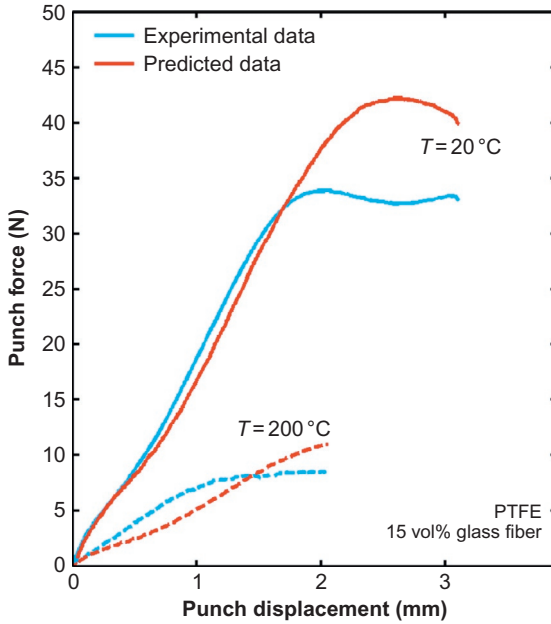




**Figure 8.26** Comparison between experimental volumetric compression data and model predictions from the DNF model.



**Figure 8.27** Deformed specimen used in the punch experiments.



**Figure 8.28** Comparison between experimental data for a fluoropolymer and model predictions from the DNF model.

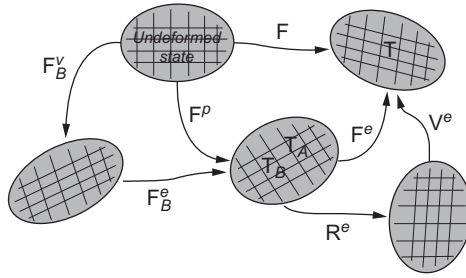
## 8.5 Hybrid Model

The Hybrid Model (HM) is an advanced material model specifically developed for predicting the large strain time-dependent behavior of ultra-high molecular weight polyethylene (UHMWPE) [24, 29]. This model can also be used to predict the response of many other types of thermoplastics.<sup>4</sup>

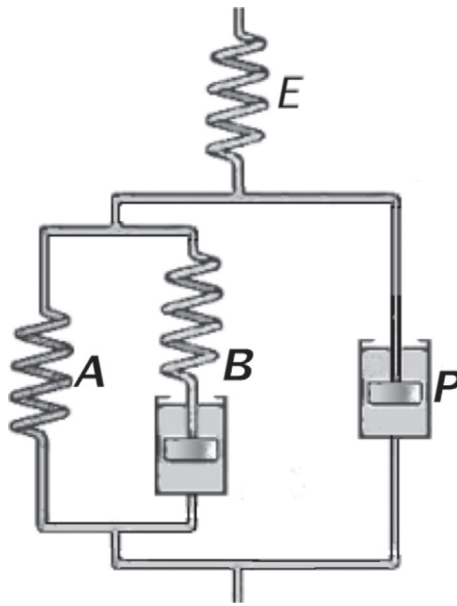
The kinematic framework used in the HM is based on a decomposition of the applied deformation gradient into elastic and viscoplastic components:  $\mathbf{F} = \mathbf{F}^e \mathbf{F}^p$ , see Figure 8.29.

The model can also be represented using the rheological representation shown in Figure 8.30.

<sup>4</sup>The HM model is available in the PolyUMod library [22].



**Figure 8.29** Deformation map used in the Hybrid Model.



**Figure 8.30** Rheological representation of the Hybrid Model.

The Cauchy stress for the HM at a given deformation state is given by the isotropic linear elasticity expression:

$$\sigma = \frac{1}{J^e} (2\mu \mathbf{E}^e + \lambda \text{tr}[\mathbf{E}^e] \mathbf{I}), \quad (8.52)$$

where  $\mathbf{V}^e$  is left stretch tensor,  $\mathbf{E}^e = \ln[\mathbf{V}^e]$  is the logarithmic strain,  $J^e = \det[\mathbf{F}^e]$  is the relative volume change, and  $\mu$  and  $\lambda$  are the Lamé constants that can be obtained from the Young's modulus ( $E$ ) and the Poisson's ratio ( $\nu$ ) from:

$$\mu = \frac{E}{2(1 + \nu)},$$

$$\lambda = \frac{E\nu}{(1 + \nu)(1 - 2\nu)}.$$

The stress acting on the equilibrium portion of the backstress network  $A$  is given by the eight-chain model (see Section 5.3.10):

$$\boldsymbol{\sigma}_A = \frac{\mu_A}{J^p \overline{\lambda}^{p*}} \frac{\mathcal{L}^{-1}(\overline{\lambda}^{p*}/\lambda^{\text{lock}})}{\mathcal{L}^{-1}(1/\lambda^{\text{lock}})} \text{dev}[\mathbf{b}^{p*}] + \kappa(J^p - 1)\mathbf{I}, \quad (8.53)$$

where  $J^p = \det[\mathbf{F}^p]$ ,  $\mathbf{b}^{p*} = (J^p)^{-2/3} \mathbf{F} \mathbf{F}^\top$  is the distortional part of the left Cauchy-Green deformation tensor,  $\overline{\lambda}^{p*} = (\text{tr}[\mathbf{b}^{p*}]/3)^{1/2}$  is the chain stretch in network  $A$ . This hyperelastic network requires three material parameters: a shear modulus  $\mu_A$ , a locking stretch  $\lambda^{\text{lock}}$ , and a bulk modulus  $\kappa$ .

The kinematics of the viscoelastic flow portion of the backstress network  $B$  is represented by an approach similar to the BB model (see Section 8.2). Specifically, the deformation gradient is decomposed into elastic and viscoelastic components:  $\mathbf{F}^p = \mathbf{F}_B^e \mathbf{F}_B^v$ . The stress driving the viscoplastic flow of the backstress network is obtained from the eight-chain model:

$$\boldsymbol{\sigma}_B = s_B \frac{\mu_A}{J_B^e \overline{\lambda}_B^{e*}} \frac{\mathcal{L}^{-1}(\overline{\lambda}_B^{e*}/\lambda^{\text{lock}})}{\mathcal{L}^{-1}(1/\lambda^{\text{lock}})} \text{dev}[\mathbf{b}_B^{e*}] + \kappa(J_B^e - 1)\mathbf{I}, \quad (8.54)$$

where  $J_B^e = \det[\mathbf{F}_B^e]$ ,  $\mathbf{b}_B^{e*} = (J_B^e)^{-2/3} \mathbf{F}_B^e (\mathbf{F}_B^e)^\top$ ,  $\overline{\lambda}_B^{e*} = (\text{tr}[\mathbf{b}_B^{e*}]/3)^{1/2}$ , and  $s_B$  is a dimensionless parameter specifying the relative stiffness of network  $B$  compared to network  $A$ .

At small deformations the stiffness of the backstress network is constant and the material response is linear elastic. At intermediate applied deformations viscoplastic flow is initiated by molecular chain sliding. With increasing amount of viscoplastic flow, the crystalline domains become distorted and start to provide additional molecular material to the backstress network. This is manifested by an initial reduction in the effective stiffness of the backstress network with imposed strain and is represented in the model by allowing the parameter  $s_B$  to evolve during the plastic deformation as follows

$$\dot{s}_B = -\alpha_B \cdot (s_B - s_{Bf}) \cdot \dot{\gamma}_P, \quad (8.55)$$

where  $\alpha_B$  is a material parameter specifying the transition rate of the distributed yielding, and  $s_B$  in the undeformed state is  $s_{Bi}$ , and  $s_{Bf}$  in the fully transformed state. The quantity  $\dot{\gamma}_P$  is the rate of viscoplastic flow and is given by Equation (8.57).

The time derivative of the viscoelastic deformation gradient of network  $B$  is given by

$$\begin{aligned} \dot{\mathbf{F}}_B^v &= \mathbf{L}_B^v \mathbf{F}_B^v = \dot{\gamma}_0 \cdot \left( \frac{\tau_B}{\tau_B^{\text{base}} [1 + p_B / \hat{p}]} \right)^{m_B} \\ &(\mathbf{F}_B^e)^{-1} \frac{\text{dev}[\boldsymbol{\sigma}_B]}{\tau_B} \mathbf{F}^P \equiv \dot{\gamma}_B^v \mathbf{N}_B^e, \end{aligned} \quad (8.56)$$

where  $\dot{\gamma}_0 \equiv 1/s$  is a constant that is introduced to maintain dimensional consistency,  $\tau_B = \|\text{dev}[\boldsymbol{\sigma}_B]\|_F$  is the effective shear stress driving the viscoelastic flow,  $p_B = -\text{tr}[\boldsymbol{\sigma}_B]/3$  is the hydrostatic pressure, and  $\tau_B^{\text{base}}$ ,  $m_B$ , and  $\hat{p}$  are material parameters.

The time rate of change of the plastic flow of network  $P$  is captured using a similar energy activation approach as for network  $B$

$$\begin{aligned} \dot{\mathbf{F}}^P &= \mathbf{L}^P \mathbf{F}^P = \dot{\gamma}_0 \cdot \left( \frac{\tau^P}{\tau_{\text{base}}^P [1 + p^P / \hat{p}]} \right)^{m^P} \\ &(\mathbf{R}^e)^\top \frac{\text{dev}[\mathbf{T}^P]}{\tau^P} \mathbf{R}^e \mathbf{F}^P \equiv \dot{\gamma}^P \mathbf{N}^P, \end{aligned} \quad (8.57)$$

where  $\dot{\gamma}_0 \equiv 1/s$  is a constant that is introduced to maintain dimensional consistency,  $\tau^P = \|\text{dev}[\mathbf{T}^P]\|_F$  is the effective shear stress driving the plastic flow,  $p^P = -\text{tr}[\mathbf{T}^P]/3$  is the hydrostatic pressure, and  $\tau_{\text{base}}^P$ ,  $m^P$ , and  $\hat{p}$  are material parameters.

In total, the augmented HM requires 13 material parameters (see Table 8.2): 2 small strain elastic constants ( $E_e$ ,  $\nu_e$ ), 3 hyperelastic constants for the back stress network ( $\mu_A$ ,  $\lambda^{\text{lock}}$ ,  $\kappa$ ), 6 flow constants for the backstress network ( $s_{Bi}$ ,  $s_{Bf}$ ,  $\alpha_B$ ,  $\tau_B^{\text{base}}$ ,  $m_B$ ,  $\hat{p}$ ), and two yield and viscoplastic flow parameters ( $\tau_{\text{base}}^P$ ,  $m^P$ ).

**Table 8.2** Material Parameters Used by the Augmented Hybrid Model

Index	Symbol	Description
1	$E$	Young's modulus
2	$\nu$	Poisson's ratio
3	$\mu_A$	Shear modulus $A$
4	$\lambda^{\text{lock}}$	Locking stretch
5	$\kappa$	Bulk modulus
6	$s_{Bi}$	Initial stiffness $B$
7	$s_{Bf}$	Final stiffness $B$
8	$\alpha_B$	Transition rate stiffness $B$
9	$\tau_{\text{base}}^B$	Flow resistance $B$
10	$m_B$	Stress exponent $B$
11	$\hat{p}$	Pressure dependence of flow
12	$\tau_{\text{base}}^p$	Flow resistance $p$
13	$m^p$	Stress exponent $p$

### 8.5.1 Matlab Implementation of the Hybrid Model

The HM material model is formulated as a set of differential equations that need to be solved for each time increment (Figure 8.31). One way to solve this set of equations is to use the following algorithm:

1. Known values at time  $t_i$ :
  - Deformation gradient:  $\mathbf{F}$
  - State variables:  $\mathbf{F}_B^v, \mathbf{F}^p$
2. Known values at time  $t_{i+1}$ :
  - Deformation gradient:  $\mathbf{F}$
3. Calculate  $\mathbf{F}_B^v$  at  $t_{i+1}$  using an ODE solver and Equation (8.56) for  $\dot{\mathbf{F}}_B^v$ .

Matlab File Name: mat\_HM.m

```
function [stress] = mat_HM(time, strain, params)
%mat_HM Hybrid model
%Incompressible uniaxial loading (true stress & strain)
%Params: [E, muA, lambdaL, sBi, sBf, alphaB, tauBasB, mB, ...
%        pHat, tauBasP, mP]
%State variables: [lambdaBv, lambdaP, sB]
stress = 0 * strain;
state = [1 1 params(4)]';
for i = 2 : length(strain)
    [t,stateAll] = ode45(@(t,y) flow(t, y, time, strain, params), time(i-1:i), state);
    state = stateAll(end,:);
    stress(i) = params(1) * (strain(i) - log(state(2)));
end
end

function res = flow(time, state, timeVec, strainVec, params)
% Calculates the time-derivative of the state variables
E = params( 1); % initialize variables
muA = params( 2);
lambdaL = params( 3);
sBf = params( 5);
alphaB = params( 6);
tauBasB = params( 7);
mB = params( 8);
pHat = params( 9);
tauBasP = params(10);
mP = params(11);
lambdaBv = state(1);
lambdaP = state(2);
sB = state(3);
% calculate stresses
lambda = exp( interp1(timeVec, strainVec, time) ); % stretch at the specified time
stress = E * log(lambda/lambdaP);
stressA = mat_EC(time, log(lambdaP), [muA lambdaL]);
stressB = mat_EC(time, log(lambdaP/lambdaBv), [sB*muA lambdaL]);
stressP = stress - (lambda/lambdaP)^2 * (stressA + stressB);
% calculate time-derivatives of state variables
res(1,1) = (abs(stressB) / (tauBasB * (1+stressB/pHat)))^mB * ...
    lambdaBv * sign(stressB); % lambdaBvDot
res(2,1) = (abs(stressP) / (tauBasP*(1+stressP/pHat)))^mP * ...
    lambdaP * sign(stressP); % lambdaPDot
res(3,1) = -alphaB * (sB - sBf) * ...
    (abs(stressP) / (tauBasP*(1+stressP/pHat)))^mP; % sBDot
end
```

Figure 8.31 Matlab implementation of the Hybrid Model.

4. Calculate  $\mathbf{F}^P$  at  $t_{i+1}$  using an ODE solver and Equation (8.57) for  $\dot{\mathbf{F}}^P$ .
5. Calculate the stress at  $t_{i+1}$  using Equation (8.52).

For the case of incompressible uniaxial loading, the results from running this function are shown in Figure 8.32.

## 8.5.2 Use of the Hybrid Model for Polymer Modeling

The HM is useful for predicting the mechanical response of thermoplastic materials below the glass transition temperature, or semi-crystalline polymers below the melting temperature.

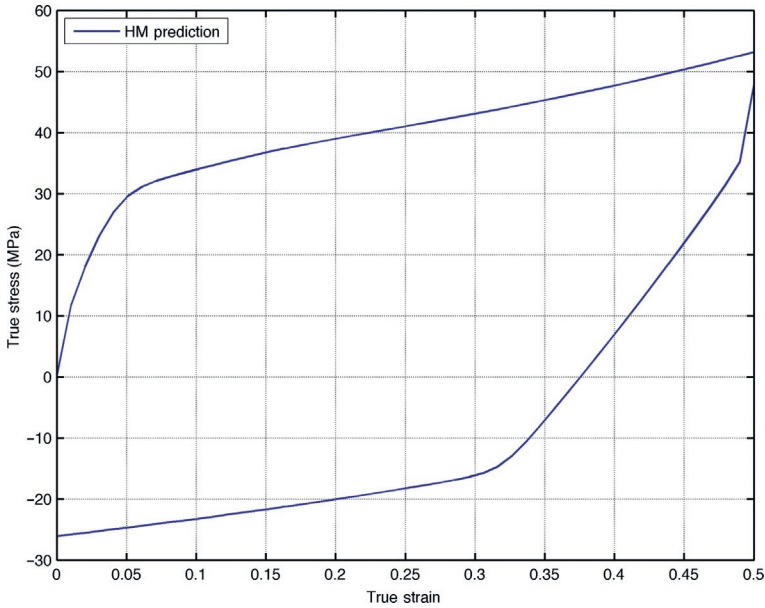


Figure 8.32 Matlab test of the Hybrid Model.

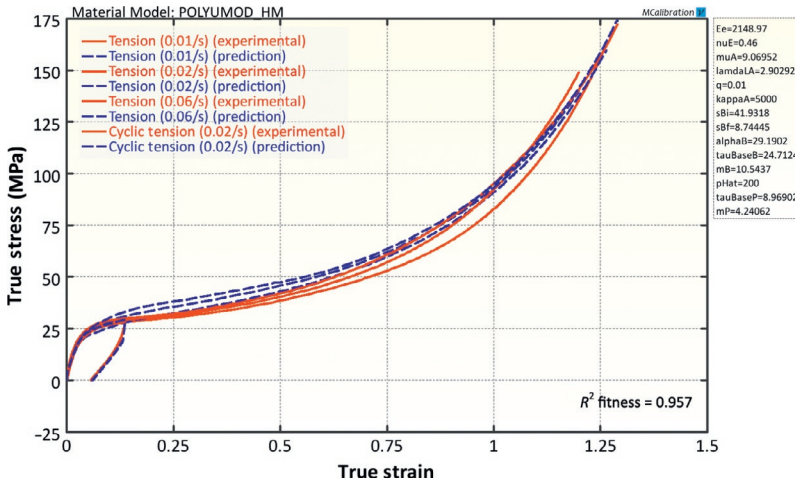
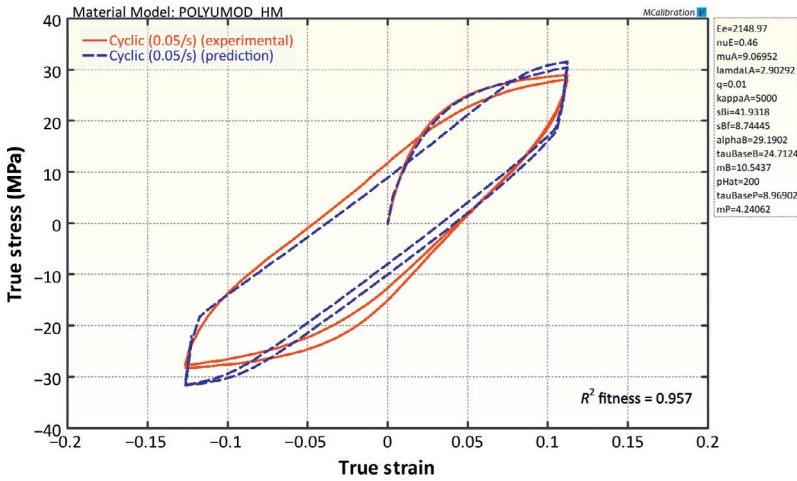


Figure 8.33 Comparison between experimental data in uniaxial tension for UHMWPE and predictions from the Hybrid Model.

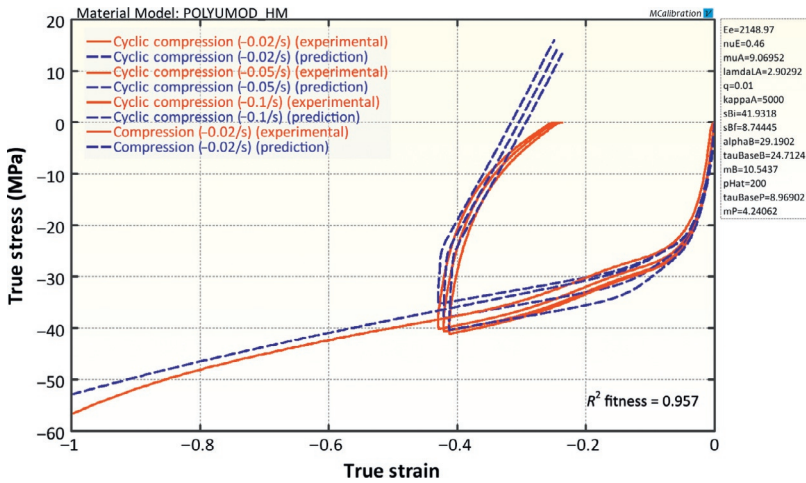
The HM was originally developed for predicting the response of UHMWPE in biomedical applications [24, 29], but the model is suitable also for other thermoplastics.

Figures 8.33–8.35 compare the predictions from the HM to experimental data for UHMWPE (crosslinked GUR 1050). The





**Figure 8.34** Comparison between cyclic experimental data for UHMWPE and predictions from the Hybrid Model.



**Figure 8.35** Comparison between experimental data in uniaxial compression for UHMWPE and predictions from the Hybrid Model.

material model was calibrated to all uniaxial tension and compression data simultaneously. The results are here shown in three separate images in order to better illustrate the predictive capabilities of the model. These images show that the HM can accurately predict the behavior of this material.

## 8.6 Three Network Model

The Three Network Model (TNM) is a material model specifically developed for thermoplastic materials. It has many features that are similar to the HM, but is designed to be more numerically efficient.<sup>5</sup>

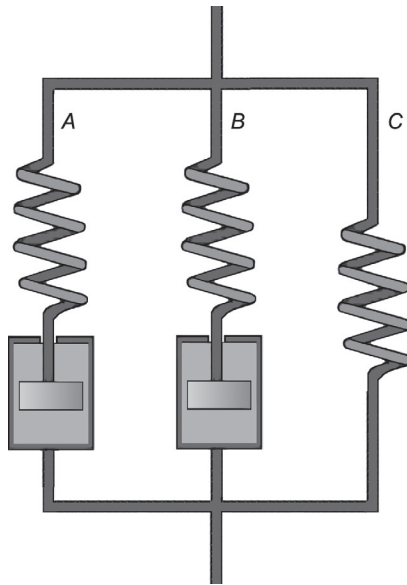
As specified by its name, the kinematics of the TNM consists of three parts, or molecular networks, acting in parallel, see the rheological representation in [Figure 8.36](#).

The total deformation gradient  $\mathbf{F}^{\text{appl}}$  contains both a thermal expansion part  $\mathbf{F}^{\text{th}} = [1 + \alpha(\theta - \theta_0)] \mathbf{I}$ , and a mechanical deformation part  $\mathbf{F}$ :

$$\mathbf{F}^{\text{appl}} = \mathbf{F} \mathbf{F}^{\text{th}}.$$

The deformation gradient acting on network  $A$  is multiplicatively decomposed into viscoplastic and viscoelastic components:

$$\mathbf{F} = \mathbf{F}_A^e \mathbf{F}_A^v. \quad (8.58)$$



**Figure 8.36** Rheological representation of the constitutive model.

<sup>5</sup>The TN model is available in the PolyUMod library [22].

The Cauchy stress acting on network  $A$  is given by the eight-chain representation [4, 19]:

$$\mathbf{T}_A = \frac{\mu_A}{J_A^e \overline{\lambda}_A^{e*}} \left[ 1 + \frac{\theta - \theta_0}{\hat{\theta}} \right] \frac{\mathcal{L}^{-1} \left( \frac{\overline{\lambda}_A^{e*}}{\lambda^{\text{lock}}} \right)}{\mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right)} \text{dev} [\mathbf{b}_A^{e*}] + \kappa (J_A^e - 1) \mathbf{1}, \quad (8.59)$$

where  $J_A^e = \det[\mathbf{F}_A^e]$ ,  $\mu_A$  is the initial shear modulus,  $\lambda^{\text{lock}}$  is the chain locking stretch,  $\theta$  is the current temperature,  $\theta_0 = 293 \text{ K}$  is a fixed reference temperature,  $\hat{\theta}$  is a material parameter specifying the temperature response of the stiffness,  $\mathbf{b}_A^{e*} = (J_A^e)^{-2/3} \mathbf{F}_A^e (\mathbf{F}_A^e)^\top$  is a Cauchy-Green deformation tensor,  $\overline{\lambda}_A^{e*} = (\text{tr}[\mathbf{b}_A^{e*}]/3)^{1/2}$  is the effective chain stretch based on the eight-chain topology assumption [19],  $\mathcal{L}^{-1}(x)$  is the inverse Langevin function, where  $\mathcal{L}(x) = \coth(x) - 1/x$ , and  $\kappa$  is the bulk modulus. By explicitly incorporating the temperature dependence of the shear modulus it is possible to capture the stiffness variation of the material over a wide range of temperatures.

The viscoelastic deformation gradient acting on network  $B$  is decomposed into elastic and viscous parts:

$$\mathbf{F} = \mathbf{F}_B^e \mathbf{F}_B^v. \quad (8.60)$$

The Cauchy stress acting on network  $B$  is obtained from the same eight-chain network representation that was used for network  $A$ .

$$\mathbf{T}_B = \frac{\mu_B}{J_B^e \overline{\lambda}_B^{e*}} \left[ 1 + \frac{\theta - \theta_0}{\hat{\theta}} \right] \frac{\mathcal{L}^{-1} \left( \frac{\overline{\lambda}_B^{e*}}{\lambda^{\text{lock}}} \right)}{\mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right)} \text{dev} [\mathbf{b}_B^{e*}] + \kappa (J_B^e - 1) \mathbf{1}, \quad (8.61)$$

where  $J_B^e = \det[\mathbf{F}_B^e]$ ,  $\mu_B$  is the initial shear modulus,  $\mathbf{b}_B^{e*} = (J_B^e)^{-2/3} \mathbf{F}_B^e (\mathbf{F}_B^e)^\top$  is a Cauchy-Green deformation tensor, and  $\overline{\lambda}_B^{e*} = (\text{tr}[\mathbf{b}_B^{e*}]/3)^{1/2}$  is the effective chain stretch based on the eight-chain topology assumption [19]. In Equation (8.61), the effective shear modulus is taken to evolve with plastic strain from an initial value of  $\mu_{Bi}$  according to:

$$\dot{\mu}_B = -\beta [\mu_B - \mu_{Bf}] \cdot \dot{\gamma}_A, \quad (8.62)$$

where  $\dot{\gamma}_A$  is the viscoplastic flow rate defined in Equation (8.64). This equation enables the model to better capture the distributed yielding that is observed in many thermoplastics.

Similarly the Cauchy stress acting on network  $C$  is given by

$$\mathbf{T}_C = \frac{\mu_C}{J\bar{\lambda}^*} \left[ 1 + \frac{\theta - \theta_0}{\hat{\theta}} \right] \frac{\mathcal{L}^{-1} \left( \frac{\bar{\lambda}^*}{\lambda^{\text{lock}}} \right)}{\mathcal{L}^{-1} \left( \frac{1}{\lambda^{\text{lock}}} \right)} \text{dev} [\mathbf{b}^*] + \kappa(J - 1)\mathbf{1}, \quad (8.63)$$

where  $J = \det[\mathbf{F}]$ ,  $\mu_C$  is the initial shear modulus,  $\mathbf{b}^* = J^{-2/3}\mathbf{F}(\mathbf{F})^\top$  is a Cauchy-Green deformation tensor, and  $\bar{\lambda}^* = (\text{tr}[\mathbf{b}^*]/3)^{1/2}$  is the effective chain stretch based on the eight-chain topology assumption [19].

Using this framework, the total Cauchy stress in the system is given by  $\mathbf{T} = \mathbf{T}_A + \mathbf{T}_B + \mathbf{T}_C$ .

The total velocity gradient of network  $A$ ,  $\mathbf{L} = \dot{\mathbf{F}}\mathbf{F}^{-1}$ , can be decomposed into elastic and viscous components:  $\mathbf{L} = \mathbf{L}_A^e + \mathbf{F}_A^e \mathbf{L}_A^v \mathbf{F}_A^{e-1} = \mathbf{L}_A^e + \tilde{\mathbf{L}}_A^v$ , where  $\mathbf{L}_A^v = \dot{\mathbf{F}}_A^v \mathbf{F}_A^{v-1} = \mathbf{D}_A^v + \mathbf{W}_A^v$  and  $\tilde{\mathbf{L}}_A^v = \tilde{\mathbf{D}}_A^v + \tilde{\mathbf{W}}_A^v$ . The unloading process relating the deformed state with the intermediate state is not uniquely defined since an arbitrary rigid body rotation of the intermediate state still leaves the state stress free. The intermediate state can be made unique in different ways [16], one particularly convenient way that is used here is to prescribe  $\tilde{\mathbf{W}}_A^v = \mathbf{0}$ . This will, in general, result in elastic and inelastic deformation gradients both containing rotations. The rate of viscoplastic flow of network  $A$  is constitutively prescribed by  $\tilde{\mathbf{D}}_A^v = \dot{\gamma}_A \mathbf{N}_A$ . The tensor  $\mathbf{N}_A$  specifies the direction of the driving deviatoric stress of the relaxed configuration convected to the current configuration, and the term  $\dot{\gamma}_A$  specifies the effective deviatoric flow rate. Noting that  $\mathbf{T}_A$  is computed in the loaded configuration, the driving deviatoric stress on the relaxed configuration convected to the current configuration is given by  $\mathbf{T}'_A = \text{dev}[\mathbf{T}_A]$ , and by defining an effective stress by the Frobenius norm  $\tau_A = \|\mathbf{T}'_A\|_F \equiv (\text{tr}[\mathbf{T}'_A \mathbf{T}'_A])^{1/2}$ , the direction of the driving deviatoric stress becomes  $\mathbf{N}_A = \mathbf{T}'_A / \tau_A$ . The effective deviatoric flow rate is given by the reptation-inspired equation [4]:

$$\dot{\gamma}_A = \dot{\gamma}_0 \cdot \left( \frac{\tau_A}{\hat{\tau}_A + aR(p_A)} \right)^{m_A} \cdot \left( \frac{\theta}{\theta_0} \right)^n, \quad (8.64)$$

where  $\dot{\gamma}_0 \equiv 1/s$  is a constant introduced for dimensional consistency,  $p_A = -[(\mathbf{T}_A)_{11} + (\mathbf{T}_A)_{22} + (\mathbf{T}_A)_{33}]/3$  is the hydrostatic pressure,  $R(x) = (x + |x|)/2$  is the ramp function, and  $\hat{\tau}_A$ ,  $\beta$ ,  $m_A$ ,  $n$ , and  $\theta_0$  are specified material parameters. In this framework, the temperature dependence of the flow rate is taken to follow a power law form. In summary, the velocity gradient of the viscoelastic flow of network  $A$  can be written

$$\dot{\mathbf{F}}_A^v = \dot{\gamma}_A \mathbf{F}_A^{e-1} \frac{\text{dev}[\mathbf{T}_A]}{\tau_A} \mathbf{F}. \quad (8.65)$$

The total velocity gradient of network  $B$  can be obtained very similarly as for network  $A$ . Specifically,  $\mathbf{L} = \dot{\mathbf{F}}\mathbf{F}^{-1}$  can be decomposed into elastic and viscous components:  $\mathbf{L} = \mathbf{L}_B^e + \mathbf{F}_B^e \mathbf{L}_B^v \mathbf{F}_B^{e-1} = \mathbf{L}_B^e + \tilde{\mathbf{L}}_B^v$ , where  $\mathbf{L}_B^v = \dot{\mathbf{F}}_B^v \mathbf{F}_B^{v-1} = \mathbf{D}_B^v + \mathbf{W}_B^v$  and  $\tilde{\mathbf{L}}_B^v = \tilde{\mathbf{D}}_B^v + \tilde{\mathbf{W}}_B^v$ . The unloading process relating the deformed state with the intermediate state is not uniquely defined since an arbitrary rigid body rotation of the intermediate state still leaves the state stress free. The intermediate state can be made unique in different ways [16], one particularly convenient way that is used here is to prescribe  $\tilde{\mathbf{W}}_B^v = \mathbf{0}$ . This will, in general, result in elastic and inelastic deformation gradients both containing rotations. The rate of viscoplastic flow of network  $B$  is constitutively prescribed by  $\tilde{\mathbf{D}}_B^v = \dot{\gamma}_B \mathbf{N}_B$ . The tensor  $\mathbf{N}_B$  specifies the direction of the driving deviatoric stress of the relaxed configuration convected to the current configuration, and the term  $\dot{\gamma}_B$  specifies the effective deviatoric flow rate. Noting that  $\mathbf{T}_B$  is computed in the loaded configuration, the driving deviatoric stress on the relaxed configuration convected to the current configuration is given by  $\mathbf{T}'_B = \text{dev}[\mathbf{T}_B]$ , and by defining an effective stress by the Frobenius norm  $\tau_B = \|\mathbf{T}'_B\|_F \equiv (\text{tr}[\mathbf{T}'_B \mathbf{T}'_B])^{1/2}$ , the direction of the driving deviatoric stress becomes  $\mathbf{N}_B = \mathbf{T}'_B / \tau_B$ . The effective deviatoric flow rate is given by the reptation-inspired equation [4]:

$$\dot{\gamma}_B = \dot{\gamma}_0 \cdot \left( \frac{\tau_B}{\hat{\tau}_B + aR(p_B)} \right)^{m_B} \cdot \left( \frac{\theta}{\theta_0} \right)^n, \quad (8.66)$$

where  $\dot{\gamma}_0 \equiv 1/s$  is a constant introduced for dimensional consistency,  $p_B = -[(\mathbf{T}_B)_{11} + (\mathbf{T}_B)_{22} + (\mathbf{T}_B)_{33}]/3$  is the hydrostatic pressure,  $R(x) = (x + |x|)/2$  is the ramp function, and  $\hat{\tau}_B$ ,  $\beta$ ,  $m_B$ ,  $n$ , and  $\theta_0$  are specified material parameters. In this framework, the temperature dependence of the flow rate is taken to follow a power law form. In summary, the velocity gradient of the viscoelastic flow of network  $B$  can be written

$$\dot{\mathbf{F}}_B^v = \dot{\gamma}_B \mathbf{F}_B^{e-1} \frac{\text{dev}[\mathbf{T}_B]}{\tau_B} \mathbf{F}. \quad (8.67)$$

The TNM model requires the material parameters in [Table 8.3](#).

**Table 8.3** Material Parameters Used by the Three Network Model

Index	Symbol	Description
1	$\mu_A$	Shear modulus of network $A$
2	$\hat{\theta}$	Temperature factor
3	$\lambda_L$	Locking stretch
4	$\kappa$	Bulk modulus
5	$\hat{\tau}_A$	Flow resistance of network $A$
6	$a$	Pressure dependence of flow
7	$m_A$	Stress exponential of network $A$
8	$n$	Temperature exponential
9	$\mu_{Bi}$	Initial shear modulus of network $B$
10	$\mu_{Bf}$	Final shear modulus of network $B$
11	$\beta$	Evolution rate of $\mu_B$
12	$\hat{\tau}_B$	Flow resistance of network $B$
13	$m_B$	Stress exponential of network $B$
14	$\mu_C$	Shear modulus of network $C$
15	$\alpha$	Thermal expansion coefficient
16	$\theta_0$	Thermal expansion reference temperature

### 8.6.1 Matlab Implementation of the Three Network Model

Like the other models in this chapter, the TNM is formulated as a set of differential equations. These equations can be quickly solved for uniaxial loading using the equations in the following function (Figures 8.37 and 8.38).

A Matlab function that exercises the `mat_TNM()` function is shown in the following code, and the results from running the code are shown in Figure 8.39.

### 8.6.2 Python Implementation of the Three Network Model

For incompressible uniaxial loading the Three Network (TN) model can be implemented into Python code as shown below. The code builds upon the code from Chapter 5, and previous examples from this chapter. First a few help functions are defined.

Additional Code to "Polymer\_Mechanics\_Chap09.py":

```
def Dev(A):
    """Deviatoric part of a tensor"""
    return A - sum(A)/3.0

def Inv(A):
    """Inverse of a tensor"""
    return array([1.0, 1.0, 1.0]) / A

def pressure(A):
    """Pressure of a stress tensor"""
    return -sum(A) / 3.0
```

The time-derivative of the state variables in the TN model are calculated in the function `TNM_timeDer_3D()`, listed below. The main function for the TN model is provided by the function `TNM_3D()`.

The actual commands that sets everything up and calls the main function are listed in the file `TNM_Compressible_Uniaxial.py`.

The results from running this code are shown in Figure 8.40.

**Additional Code to “Polymer\_Mechanics\_Chap09.py”:**

```
def TNM_timeDer_3D(statev, t, params, time0, time1, F0, F1):
    """Returns statevDot"""
    muA, lamL, kappa = params[0:3]
    tauHatA, a, mA = params[3:6]
    muBi, muBf, beta = params[6:9]
    tauHatB, mB, muC = params[9:12]
    res = zeros(len(statev))
    F = F0 + (t-time0) / (time1-time0) * (F1-F0)

    # Network A: FAV
    Fv = statev[0:3]
    muB = statev[6]
    Fe = F / Fv
    Stress = toVec(EC_3D(Fe, [muA, lamL, kappa]))
    tau = norm(Dev(Stress)) + 1.0e-9
    gamDot = (tau / (tauHatA + a * ramp(pressure(Stress))))**mA
    res[0:3] = gamDot/tau * (Inv(Fe) * Dev(Stress) * F)
    res[6] = -beta * (statev[6] - muBf) * gamDot

    # Network B: FBV
    Fv = statev[3:6]
    muB = statev[6]
    Fe = F / Fv
    Stress = toVec(EC_3D(Fe, [muB, lamL, kappa]))
    tau = norm(Dev(Stress)) + 1.0e-9
    gamDot = (tau / (tauHatB + a * ramp(pressure(Stress))))**mB
    res[3:6] = gamDot/tau * (Inv(Fe) * Dev(Stress) * F)
    return res
```

**Additional Code to “Polymer\_Mechanics\_Chap09.py”:**

```
def TNM_3D(F0, F1, statev0, time0, time1, params):
    """TN-model. 3D loading specified by principal stretches.
    params: [muA, lamL, kappa, tauHatA, a, mA, muBi, mBf,
            beta, tauHatB, mB, muC].
    Returns: (true stress, statev1)"""
    muA, lamL, kappa, s = params[0:4]
    muC = params[11]
    StressC = toVec(EC_3D(F1, [muC, lamL, kappa]))

    statev1 = scipy.integrate.odeint(TNM_timeDer_3D, statev0, \
        [time0, time1], args=(params, time0, time1, F0, F1))[1]
    FAv1 = statev1[0:3]
    FBv1 = statev1[3:6]
    muB = statev1[6]

    FAe1 = F1 / FAv1
    StressA = toVec(EC_3D(FAe1, [muA, lamL, kappa]))

    FBe1 = F1 / FBv1
    StressB = toVec(EC_3D(FBe1, [muB, lamL, kappa]))

    Stress = StressA + StressB + StressC
    return (Stress, statev1)
```

**Python Code “TNM\_compressible\_uniaxial.py”:**

```
from Polymer_Mechanics_Chap05 import *
from Polymer_Mechanics_Chap09 import *

N = 50
timeVec = append(linspace(0,10.0,N), linspace(10.0,20.0,N)[1:])
trueStrain = append(linspace(0,0.2,N), linspace(0.2,0,N)[1:])

params = [290.0, 5.0, 2000.0, 7.0, 0.0, 9.5, \
          130.0, 50.0, 10.0, 24.0, 9.0, 8.0]

# FAV, FBV, muB
statev0 = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, params[6]]

trueStress = uniaxial_stress_visco(TNM_3D, timeVec, trueStrain,
    params, statev0)

plot(trueStrain, trueStress, 'r-', label='Python Calculation')
xlabel('True Strain')
ylabel('True Stress (MPa)')
grid('on')
savefig('TNM_compressible_uniaxial.png', dpi=300)
show()
```



Matlab File Name: mat\_TNM.m

```
function [stress] = mat_TNM(time, strain, params)
%mat_TNM Three-Network Model
%Incompressible uniaxial loading (uses true strain & stress)
%Parameters: [muA lambdaL tauHatA a mA muBi muBf beta tauHatB mB muC]
%State variables: [lambdaAv lambdaBv muB]
muA = params(1);
lambdaL = params(2);
muBi = params(6);
muC = params(11);
stress = 0 * strain;
state = [1 1 muBi]';
for i = 2 : length(strain)
    [t,stateAll] = ode45(@(t,y) flow(t, y, time, strain, params), time(i-1:i), state);
    state = stateAll(end,:);
    lambda = exp(strain(i));
    stressA = mat_EC(time(i), log(lambda/state(1)), [muA lambdaL]);
    stressB = mat_EC(time(i), log(lambda/state(2)), [state(3) lambdaL]);
    stressC = mat_EC(time(i), strain(i), [muC lambdaL]);
    stress(i) = stressA + stressB + stressC;
end
end

function res = flow(time, state, timeVec, strainVec, params)
% Calculates the time-derivative of the state variables
muA = params(1);
lambdaL = params(2);
tauHatA = params(3);
a = params(4);
mA = params(5);
muBf = params(7);
beta = params(8);
tauHatB = params(9);
mB = params(10);
% network A
lambda = exp( interp1(timeVec, strainVec, time) ); % stretch at the specified time
lambdaAv = state(1);
stressA = mat_EC(time, log(lambda/lambdaAv), [muA lambdaL]);
gammaADot = (abs(stressA)/(tauHatA+a*stressA))^mA;
res(1,1) = gammaADot * sign(stressA) * lambdaAv;
% network B
lambdaBv = state(2);
muB = state(3);
stressB = mat_EC(time, log(lambda/lambdaBv), [muB lambdaL]);
res(2,1) = (abs(stressB)/(tauHatB+a*stressB))^mB * sign(stressB) * lambdaBv;
res(3,1) = -beta * (muB - muBf) * gammaADot;
end
```

**Figure 8.37** Matlab implementation of the Three Network Model.

Matlab File Name: test\_mat\_TNM.m

```
time = linspace(0,2)';
strain = [linspace(0, 0.5,50)'; linspace(0.5, 0, 50)'];
stress = mat_TNM(time, strain, [192 3.11 7.2 0.0001 9.7 131 48 11 25 10 8]);
hp = plot(strain, stress);
```

**Figure 8.38** Matlab code to test the implementation of the Three Network Model.

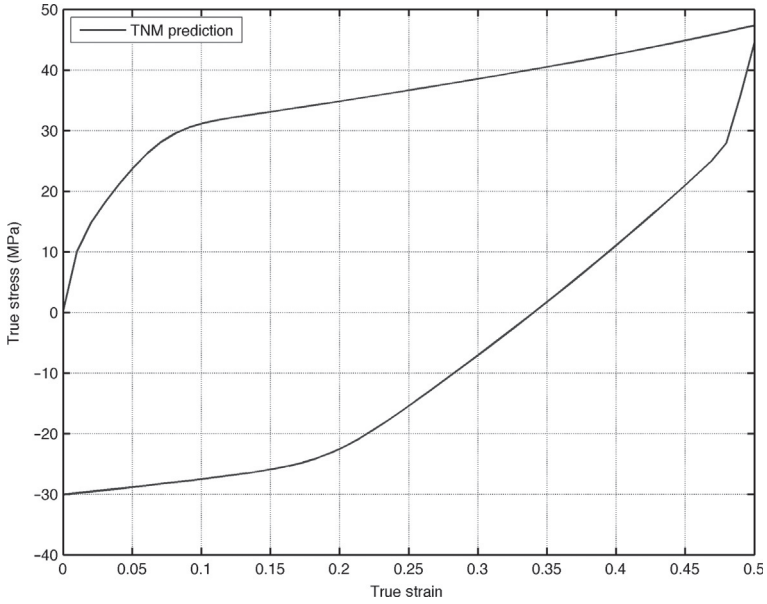


Figure 8.39 Matlab test of the Three Network Model.

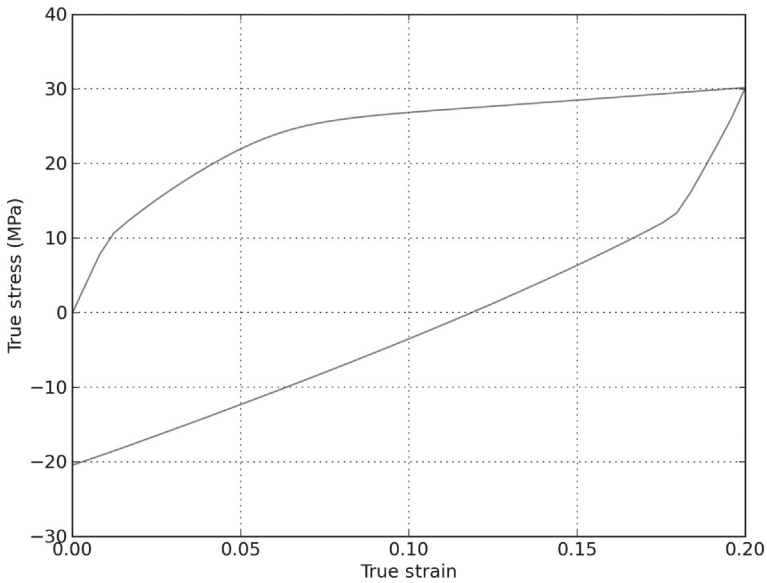


Figure 8.40 Predictions from the Python file `TNM_Compressible_Uniaxial.py`.

### 8.6.3 Use of the Three Network Model for Polymer Modeling

The TNM has many similarities to the HM in terms of target materials and behaviors. The TNM is specifically useful for predicting the mechanical response of thermoplastic materials below the glass transition temperature, or semi-crystalline polymers below the melting temperature.

Figures 8.41–8.43 compare the predictions from the TNM to experimental data for UHMWPE (crosslinked GUR 1050). As shown in the figures the TNM accurately predicts the behavior of the material.

Figure 8.41 shows the response in uniaxial tension at three different strain-rates. Figure 8.42 shows the response in cyclic loading at an intermediate strain, and Figure 8.43 shows the predictions in uniaxial compression.

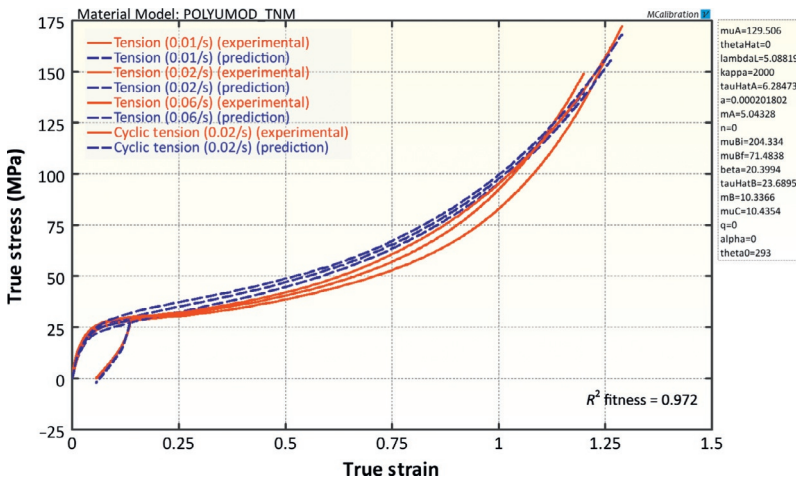


Figure 8.41 Uniaxial tensile results from the Three Network Model.

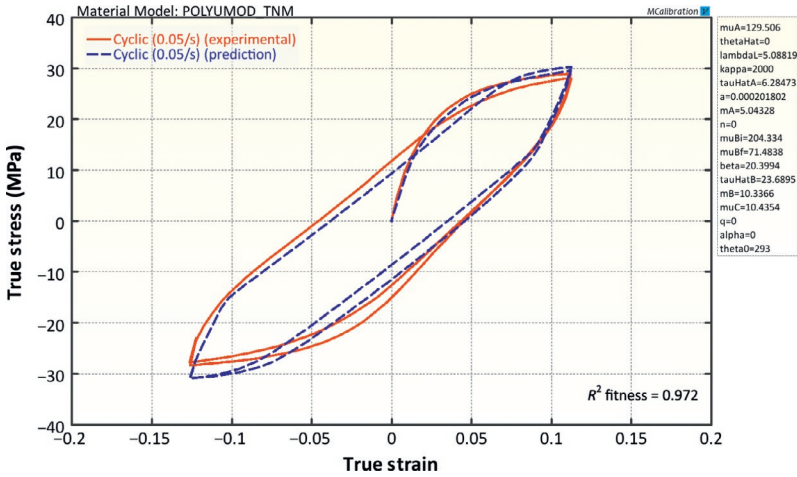


Figure 8.42 Uniaxial cyclic results from the Three Network Model.

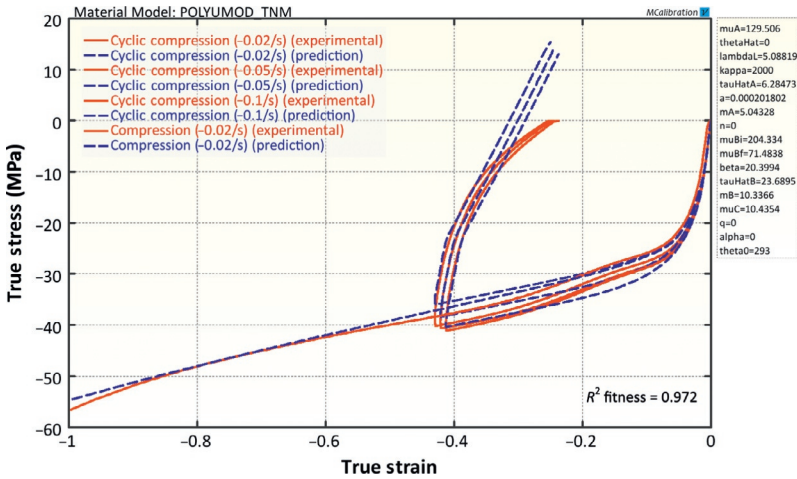
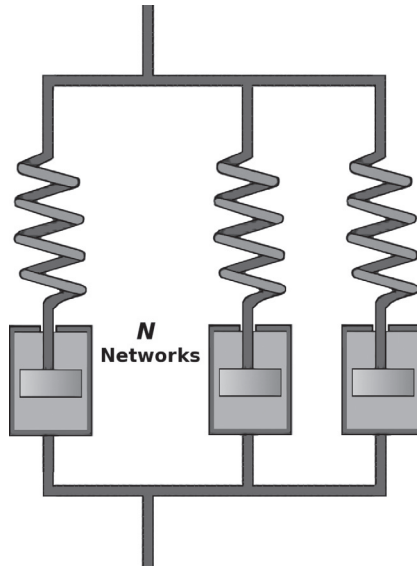


Figure 8.43 Uniaxial compressive results from the Three Network Model.

### 8.7 Parallel Network Model

The Parallel Network (PN) model is an extension of the TNM that allows for an arbitrary number of parallel networks, where each network consists of an elastic component and an optional flow component. The PN model is supported by both the

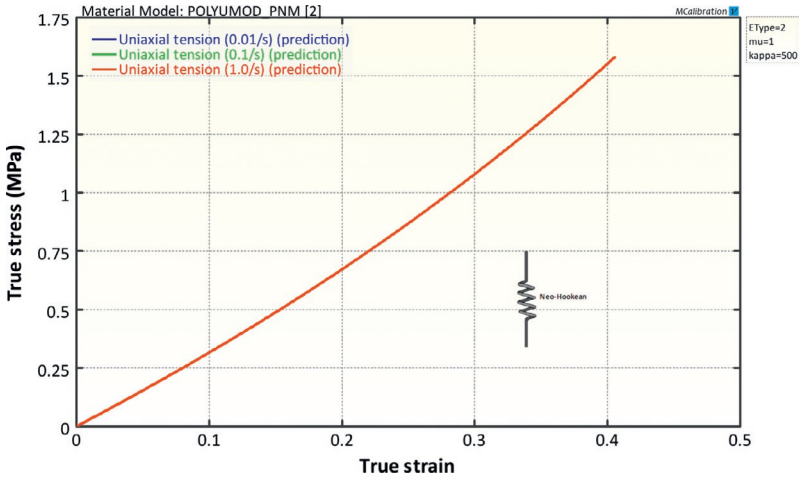


**Figure 8.44** Rheological representation of the parallel network model.

MCalibration [30] and the PolyUMod [22] software. [Figure 8.44](#) shows a rheological representation of the PN model.

The constitutive model framework supports a large number of different isotropic and anisotropic elastic behaviors, coupled with a large collection of isotropic and anisotropic viscoplastic behaviors. The model framework also supports many different damage and failure models. These features make this model framework one of the most advanced material model frameworks that have been developed, and it can capture the mechanical response of almost any polymeric material. This flexibility can make the model challenging to setup for someone not experienced in visualizing the response of rheological models. For that reason this section contains a selection of simple model frameworks and their corresponding responses. A complete description is provided in the PolyUMod User's Manual.

[Figure 8.45](#) shows the stress-strain response for a simple one-network model consisting of a Neo-Hookean element. As shown in this figure the material response is independent of the applied strain rate.



**Figure 8.45** Predictions from a simple PN model with hyperelastic network.

By adding a viscoelastic flow element with a flow rate that depends on the normalized shear stress raised to a power:

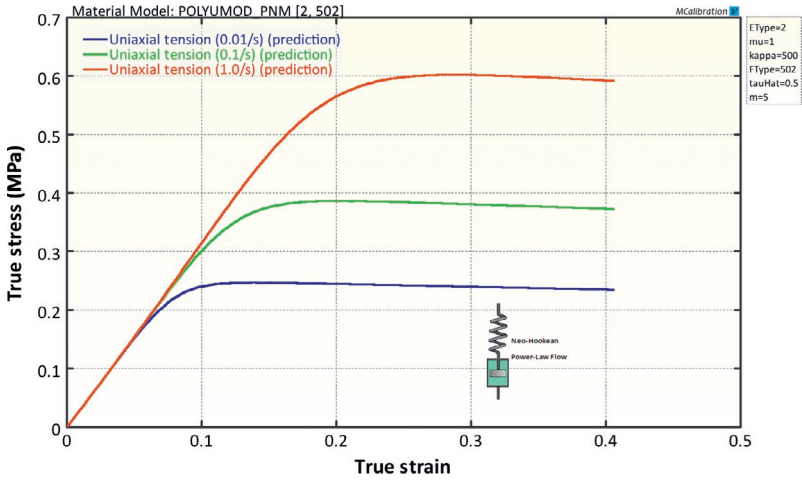
$$\dot{\gamma} = \left( \frac{\tau}{\hat{\tau}} \right)^m, \quad (8.68)$$

the material response becomes non-linear viscoelastic once the stress becomes sufficiently large. The predicted stress-strain response of this model framework is shown in [Figure 8.46](#).

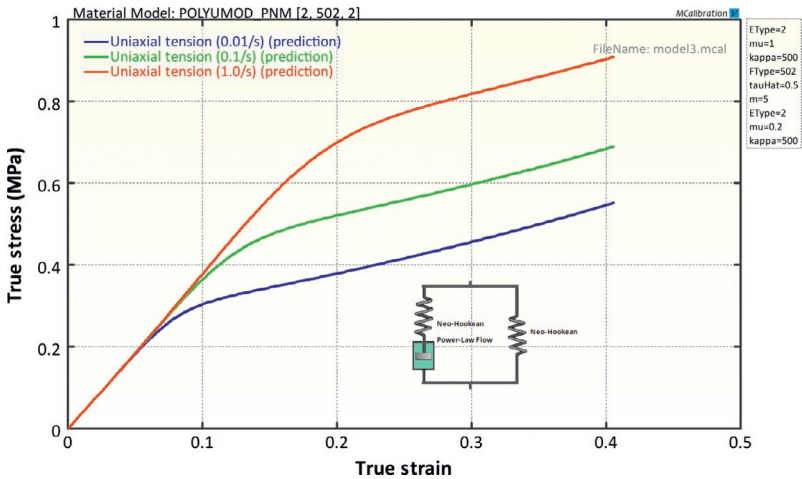
The stress-strain response after yielding can be modified by adding a second network (see [Figure 8.47](#)). This second network is typically less stiff than the first network, and makes the model similar to the BB model discussed in [Section 8.2](#).

Some polymers soften after yielding. The experimentally observed softening can often be captured using a yield evolution equation that control how the flow resistance evolves with plastic strain. [Figure 8.48](#) illustrates the predicted stress-strain response when the flow resistance ( $\hat{\tau}$ ) depends exponentially on the plastic strain:

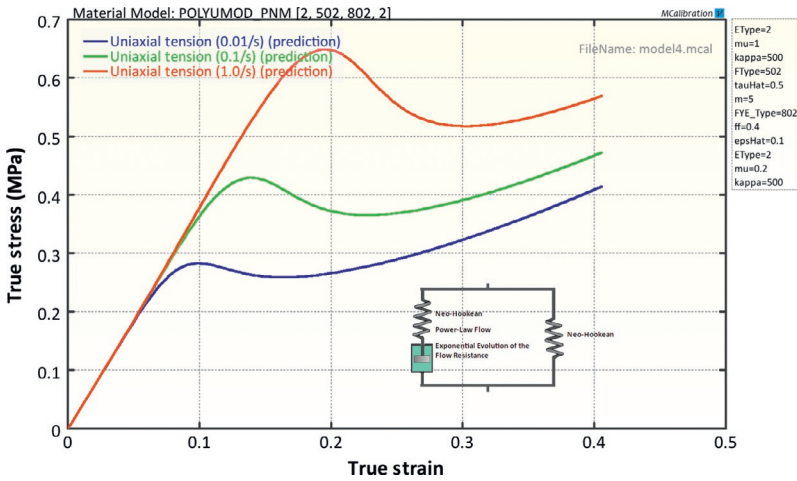
$$\hat{\tau} = \hat{\tau}_0 \left\{ f_f + (1 - f_f) \exp \left[ \frac{-\varepsilon_p}{\hat{\varepsilon}} \right] \right\}, \quad (8.69)$$



**Figure 8.46** Predictions from a simple PN model with a Neo-Hookean hyperelastic component in series with a power-law flow element.



**Figure 8.47** Predictions from a simple PN model with two parallel networks. The second network gives the stiffening beyond the onset of viscoplastic flow.



**Figure 8.48** Predictions from a simple PN model with two parallel networks. The flow element has an exponential evolution of the flow resistance.

where  $f_f$  is the final value that  $\hat{\tau}$  evolves to,  $\hat{\varepsilon}$  is the characteristic strain for the evolution, and  $\varepsilon_p$  is the effective plastic strain.

## 8.8 Use of Viscoplasticity in Polymer Modeling

Viscoplasticity is the most accurate material model framework available to represent the mechanical response of all polymers. A large collection of viscoplastic material models have been developed in the academic literature, and this chapter reviewed a collection of commonly used models that have been shown to accurately capture the behavior of elastomers, thermoplastics, and other polymers.

The material models presented here have been designed to be easy to use and calibrate. In almost all cases the hyperelastic portion of the response is limited to  $I_1$ -based energy functions, making it possible to calibrate the material models to only uniaxial tension and/or compression data. To capture the viscous time-dependent response also requires that the material be tested at different strain rates and preferably with loading—



hold—unloading cycles. Multiple examples of the practical use of viscoplastic material models are given in Chapter 11.

## 8.9 Python Code Examples

The behavior of a number of the viscoplastic material models presented in this chapter were examined using short Python functions. In each example, it was listed “Additional Code to Polymer\_Mechanics\_Chap09.py”. This section summarizes the file `Polymer_Mechanics_Chap09.py`. This file can also be downloaded from this web address:

[http://PolymerMechanics.com/Polymer\\_Mechanics\\_Chap09.zip](http://PolymerMechanics.com/Polymer_Mechanics_Chap09.zip).

```

from Polymer_Mechanics_Chap05 import *
import scipy.integrate

def ramp(x):
    return (x + abs(x)) / 2.0

def toVec(A):
    """Convert a 3x3 matrix to vector"""
    return array([A[0][0], A[1][1], A[2][2]])

def Dev(A):
    """Deviatoric part of a tensor"""
    return A - sum(A)/3.0

def Inv(A):
    """Inverse of a tensor"""
    return array([1.0, 1.0, 1.0]) / A

def pressure(A):
    """Pressure of a stress tensor"""
    return -sum(A) / 3.0

def uniaxial_stress_visco(model, timeVec, trueStrainVec, params):
    """Compressible uniaxial loading. Returns true stress."""
    stress = zeros(len(trueStrainVec))
    lam2_1 = 1.0
    FBv1 = array([1.0, 1.0, 1.0])
    for i in range(1, len(trueStrainVec)):
        print 'uniaxial_stress: i=', i, ' of ', len(trueStrainVec)
        time0 = timeVec[i-1]
        time1 = timeVec[i]
        lam1_0 = exp(trueStrainVec[i-1])
        lam1_1 = exp(trueStrainVec[i])
        lam2_0 = lam2_1
        F0 = array([lam1_0, lam2_0, lam2_0])
        F1 = array([lam1_1, lam2_1, lam2_1])
        FBv0 = FBv1.copy()
        calcS22Abs = lambda x: abs(model(F0, array([lam1_1,x,x]), \
            FBv0, time0, time1, params)[0][1])
        # search for transverse stretch that gives S22=0
        lam2_1 = scipy.optimize.fmin(calcS22Abs, x0=lam2_0,
            xtol=1e-9, ftol=1e-9, disp=False)
        res = model(F0, array([lam1_1, lam2_1, lam2_1]), FBv0, time0,

```

```

        time1, params)
        stress[i] = res[0][0]
        FBv1 = res[1]
    return stress

def BB_timeDer_3D(Fv, t, params, time0, time1, F0, F1):
    """Returns FvDot"""
    mu, lamL, kappa, s, xi, C, tauBase, m, tauCut = params[:9]
    F = F0 + (t-time0) / (time1-time0) * (F1-F0)
    Fe = F / Fv
    Stress = toVec(EC_3D(Fe, [s*mu, lamL, kappa]))
    devStress = Stress - sum(Stress)/3
    tau = norm(devStress)
    lamCh = sqrt(sum(Fv*Fv)/3.0)
    lamFac = lamCh - 1.0 + xi
    gamDot = lamFac**C * (ramp(tau/tauBase-tauCut)**m)
    prefac = 0.0
    if tau > 0: prefac = gamDot / tau
    FeInv = array([1.0, 1.0, 1.0]) / Fe
    FvDot = prefac * (FeInv * devStress * F)
    return FvDot

def BB_3D(F0, F1, FBv0, time0, time1, params):
    """BB-model. 3D loading specified by principal stretches.
    params: [muA, lamL, kappa, s, xi, C, tauHat, m, tauCut].
    Returns: (true stress, FBv1)"""
    muA, lamL, kappa, s = params[:4]
    StressA = toVec(EC_3D(F1, [muA, lamL, kappa]))
    FBv1 = scipy.integrate.odeint(BB_timeDer_3D, FBv0, \
        [time0, time1], args=(params, time0, time1, F0, F1))[1]
    FBe1 = F1 / FBv1
    StressB = toVec(EC_3D(FBe1, [s*muA, lamL, kappa]))
    Stress = StressA + StressB
    return (Stress, FBv1)

def TNM_timeDer_3D(statev, t, params, time0, time1, F0, F1):
    """Returns statevDot"""
    muA, lamL, kappa = params[0:3]
    tauHatA, a, mA = params[3:6]
    muBi, muBf, beta = params[6:9]
    tauHatB, mB, muC = params[9:12]
    res = zeros(len(statev))
    F = F0 + (t-time0) / (time1-time0) * (F1-F0)

    # Network A: FAv
    Fv = statev[0:3]
    muB = statev[6]
    Fe = F / Fv
    Stress = toVec(EC_3D(Fe, [muA, lamL, kappa]))
    tau = norm(Dev(Stress)) + 1.0e-9
    gamDot = (tau / (tauHatA + a * ramp(pressure(Stress))))**mA
    res[0:3] = gamDot/tau * (Inv(Fe) * Dev(Stress) * F)
    res[6] = -beta * (statev[6] - muBf) * gamDot

    # Network B: FBv
    Fv = statev[3:6]
    muB = statev[6]
    Fe = F / Fv
    Stress = toVec(EC_3D(Fe, [muB, lamL, kappa]))
    tau = norm(Dev(Stress)) + 1.0e-9
    gamDot = (tau / (tauHatB + a * ramp(pressure(Stress))))**mB
    res[3:6] = gamDot/tau * (Inv(Fe) * Dev(Stress) * F)
    return res

def TNM_3D(F0, F1, statev0, time0, time1, params):
    """TN-model. 3D loading specified by principal stretches.

```

```

    params: [muA, lamL, kappa, tauHatA, a, mA, muBi, mBf,
            beta, tauHatB, mB, muC].
    Returns: (true stress, statev1)"""
muA, lamL, kappa, s = params[0:4]
muC = params[11]
StressC = toVec(EC_3D(F1, [muC, lamL, kappa]))

statev1 = scipy.integrate.odeint(TNM_timeDer_3D, statev0, \
    [time0, time1], args=(params, time0, time1, F0, F1))[1]
FAv1 = statev1[0:3]
FBv1 = statev1[3:6]
muB = statev1[6]

FAe1 = F1 / FAv1
StressA = toVec(EC_3D(FAe1, [muA, lamL, kappa]))

FBe1 = F1 / FBv1
StressB = toVec(EC_3D(FBe1, [muB, lamL, kappa]))

Stress = StressA + StressB + StressC
return (Stress, statev1)

```

## 8.10 Exercises

1. Explain the physical reason why elastomers are viscoelastic.
2. Why cannot a linear viscoelastic material model predict the strain amplitude dependence of the storage and loss moduli, but the BB model can?
3. Calibration of the BB model
  - Download the experimental data file `neoprene.csv` from the PolymerFEM.com website: [http://polymerfem.com/polymer\\_files/Nitrile\\_rubber.zip](http://polymerfem.com/polymer_files/Nitrile_rubber.zip)
  - Plot the experimental data
  - Find the material parameters for the BB-model
  - Plot the experimental data and the model predictions in one figure
  - How well does the model work?
  - In what range is the model accurate?
4. Write a Matlab function file for the BB model with Ogden-Roxburgh Mullins effect.
5. What are the main features of the AB viscoplastic material model?

6. The DNF model has been specifically designed for fluoropolymers. What aspects of the model make it more suitable than, say, the AB viscoplastic model?
7. The TNM is an exceptionally versatile material model that can predict the non-linear viscoplastic response of most thermoplastics. What materials are the TN model not suitable for?

## References

- [1] J.S. Bergström, M.C. Boyce. Constitutive modeling of the large strain time-dependent behavior of elastomers. *J. Mech. Phys. Solids*, 46 1998 931-954.
- [2] J.S. Bergström, M.C. Boyce. Mechanical behavior of particle filled elastomers. *Rubber Chem. Technol.*, 72 1999 633-656.
- [3] J.S. Bergström. Large Strain Time-Dependent Behavior of Elastomeric Materials. Ph.D. thesis, MIT 1999.
- [4] J.S. Bergström, M.C. Boyce. Large strain time-dependent behavior of filled elastomers. *Mech. Mater.*, 32 2000 620-644.
- [5] J.S. Bergström, M.C. Boyce. Constitutive modeling of the time-dependent and cyclic loading of elastomers and application to soft biological tissues. *Mech. Mater.*, 33 2001 523-530.
- [6] M.S. Green, A.V. Tobolsky. A new approach to the theory of relaxing polymeric media. *J. Chem. Phys.*, 14 (2) 1946 80-92.
- [7] A.R. Johnson, C.J. Quigley. A viscohyperelastic Maxwell model for rubber viscoelasticity. *Rubber Chem. Technol.*, 65 1992 137-153.
- [8] A.R. Johnson, R.G. Stacer. Rubber viscoelasticity using the physically constrained system's stretches as internal variables. *Rubber Chem. Technol.*, 66 1993 567-577.
- [9] A.R. Johnson, C.J. Quigley, C.E. Freese. A viscohyperelastic finite element model for rubber. *Comput. Methods Appl. Mech. Eng.*, 127 1995 163-180.
- [10] C.M. Roland. Network recovery from uniaxial extension I: elastic equilibrium. *Rubber Chem. Technol.*, 62 1989 863-879.
- [11] C.M. Roland, M.L. Warzel. Orientation effects in rubber double networks. *Rubber Chem. Technol.*, 63 1990 285-297.

- [12] ANSYS, Inc.. ANSYS Mechanical, Release 15.0, 2014.
- [13] MSC Software. Marc 2014, 2014.
- [14] LSTC Inc.. LS-DYNA V971, 2014.
- [15] HKS, Inc.. ABAQUS, Pawtucket, RI, ver. 6.14, 2014.
- [16] M.C. Boyce, G.G. Weber, D.M. Parks. On the kinematics of finite strain plasticity. *J. Mech. Phys. Solids*, 37 (5) 1989 647-665.
- [17] P.G. de Gennes. Reptation of a polymer chain in the presence of fixed obstacles. *J. Chem. Phys.*, 55 (2) 1971 572-579.
- [18] M. Doi, S.F. Edwards. *The Theory of Polymer Dynamics*. Oxford University Press, Oxford, 1986.
- [19] E.M. Arruda, M.C. Boyce. A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials. *J. Mech. Phys. Solids*, 41 (2) 1993 389-412.
- [20] E.M. Arruda, M.C. Boyce. Evolution of plastic anisotropy in amorphous polymers during finite straining. *Int. J. Plasticity*, 9 1993 697-720.
- [21] E.M. Arruda, M.C. Boyce. Effects of strain rate, temperature and thermomechanical coupling on the finite strain deformation of glassy polymers. *Mech. Mater.*, 19 1995 193-212.
- [22] PolyUMod. <http://PolyUMod.com/>.
- [23] O.A. Hasan, M.C. Boyce. Energy storage during inelastic deformation of glassy polymers. *Polymer*, 34 (24) 1993 5085-5092.
- [24] J.S. Bergström, C.M. Rimnac, S.M. Kurtz. Prediction of multiaxial mechanical behavior for conventional and highly crosslinked UHMWPE using a hybrid constitutive model. *Biomaterials*, 24 2003 1365-1380.
- [25] K. Ho, E. Krempl. Extension of the viscoplastic theory based on overstress (VBO) to capture non-standard rate dependence in solids. *Int. J. Plasticity*, 18 2002 851-872.
- [26] A. Lion. On the large deformation behavior of reinforced rubber at different temperatures. *J. Mech. Phys. Solids*, 45 1997 1805-1834.
- [27] A. Khan, H. Zhang. Finite deformation of a polymer: experiments and modeling. *Int. J. Plasticity*, 17 (9) 2001 1167-1188.
- [28] T. Kletschkowski, U. Schomburg, A. Bertram. Endochronic viscoplastic material models for filled PTFE. *Mech. Mater.*, 34 (12) 2002 795-808.
- [29] J.S. Bergström, S.M. Kurtz, C.M. Rimnac, A.A. Edidin. Constitutive modeling of ultra-high molecular weight polyethylene under large-deformation and cyclic loading conditions. *Biomaterials*, 23 2002 2329-2343.
- [30] Veryst Engineering. <http://PolyUMod.com/>.

# 9 Determination of Material Parameters from Experimental Data

---

## Chapter Outline

9.1 Introduction	437
9.2 Mathematics of Material Parameter Determination	438
9.3 Initial Guess of the Material Parameters	440
9.4 Error Measurement Functions	442
9.5 Algorithms for Parameter Extraction	444
9.6 Exercises	445
References	446

## 9.1 Introduction

Every finite element model requires a calibrated material model for each material to be simulated. Each material model, in turn, consists of two parts: a constitutive model specifying the equations that govern the material response, and a set of parameters that go into the equations and are specific for each material. As an example, the linear elastic material model is given by Equation (5.7) and takes two material parameters: a Young's modulus and a Poisson's ratio.

Different techniques can be used to determine the necessary material parameters. All methods, however, require experimental test data and the selection of an appropriate constitutive model, followed by some procedure for determining the parameters for the model from the experimental data. So far in this book we have focused on the first two of these topics, but have not spent much time discussing different procedures that can be used for calibrating the selected material model; and clearly, if the material parameter extraction is not done properly then the finite element results will not be very accurate either.

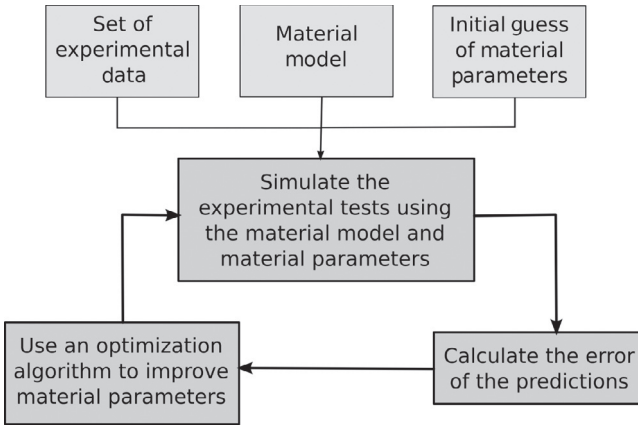
It is interesting to note that material parameter extraction is a topic that is not that well covered in the literature, perhaps because on the surface the theory is not very difficult. The difficulties involved with finding the best set of material parameters for a material model, however, are real and often a major challenge when it comes to using an advanced material model for polymers.

There are various commercial software packages that can be used to calibrate a material model from experimental data, and most FE programs contain some functionality for material model calibration. None of the major FE programs, however, include a general purpose material model calibration tool, or even tools for all the material models that are included in their library of material models. One approach that is sometimes used to overcome this problem is to rely on graphical techniques, and trial-and-error techniques. These approaches are useful for learning how a material model behaves, but it is difficult and time-consuming to use these approaches in general.

This chapter will introduce the theory behind material parameter extraction, and discuss different techniques that can be used to find an appropriate set of parameters for both simple and advanced material models.

## **9.2 Mathematics of Material Parameter Determination**

Determining the most appropriate material parameters for use with a material model require: (1) a set of experimental data; and (2), the selection of a constitutive model. As was discussed in Chapter 2, different constitutive models need different types of experimental data for the purpose of calibration. For example, a hyperelastic material model will require, at a minimum, monotonic loading to a final strain, in one or more loading modes. A hyperelastic model does not need experimental data at different strain rates, as this class of models predicts the same stress for a given strain independent of applied strain rate. To calibrate a viscoplastic model, on the other hand, requires also data at different strain-rates.



**Figure 9.1** Flow diagram for material parameter extraction.

To start the material parameter extraction, many techniques also require an initial guess of the material parameters. This guess often does not have to be very close to the optimal set of parameters, and prior knowledge of the parameters of similar materials is often sufficient.

A flow chart outlining a procedure for determining the material parameters is given in [Figure 9.1](#).

As illustrated, the first step is to simulate the experimental loading conditions using the selected material model with an initial guess of the material parameters. The model predictions are then compared to the experimental data, and the difference is used to calculate a scalar error value of the predictions. Finally, a numerical minimization algorithm is used to determine a new guess of the material parameters and the procedure is repeated until no further improvement is achieved.

This procedure can be written in the form of a mathematical minimization problem:

$$\min_{\xi} \sum_{i=1}^N f(\mathbf{M}^i(\xi) - \mathbf{E}^i), \quad (9.1)$$

where  $\xi$  is the vector of material parameters to optimize,  $N$  is the number of experimental data sets,  $f(\cdot)$  is a function for calculating the error value of a prediction,  $\mathbf{M}^i(\xi)$  is the model prediction of



experimental data set  $i$ ,  $\mathbf{E}_i$  is the experimental data set  $i$ . Both  $\mathbf{M}^i$  and  $\mathbf{E}^i$  are vectors of either stress or strain as discussed in Section 9.4. This formulation, which is also called a non-linear programming problem, needs to be solved once for each material.

To solve this non-linear optimization problem it is necessary to: (1) have a good strategy for determining the initial material parameters; (2) select an appropriate error measurement function  $f(\cdot)$ ; and (3) use a good non-linear optimization finder. These topics are discussed in more detail in the following sections.

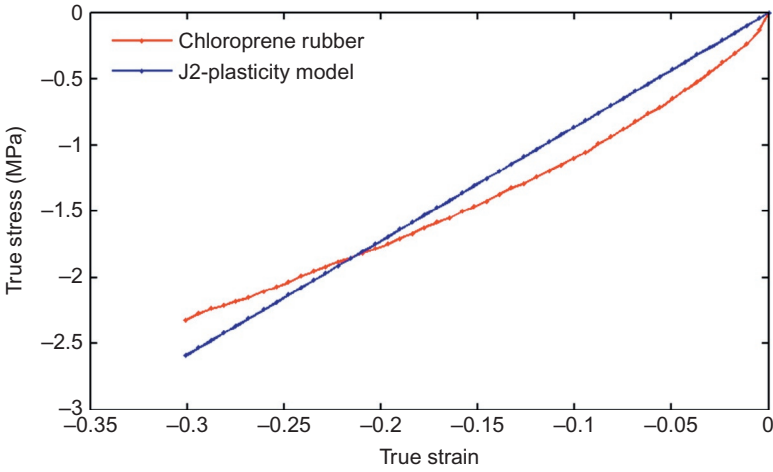
### 9.3 Initial Guess of the Material Parameters

Finding a good initial guess of the material parameters is important in order to ensure that the material parameters that are determined by the minimization algorithm are close to the global optimum. A poor initial guess can cause the minimization algorithm to get stuck at an undesirable local minimum, and it can also significantly slow down the parameter extraction procedure.

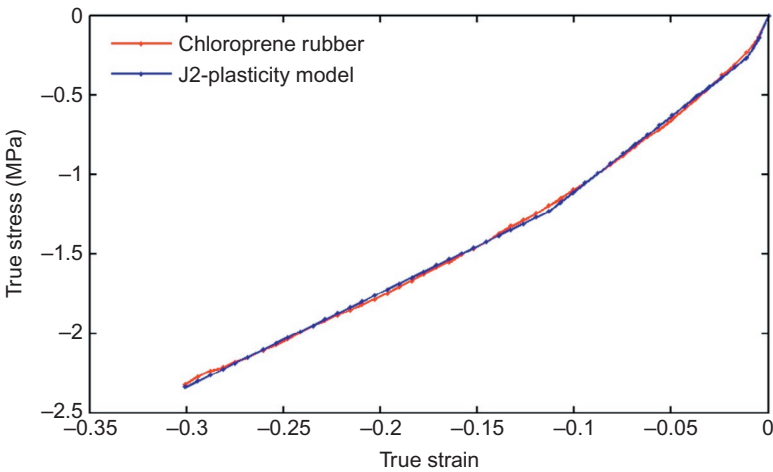
As an example, Figure 9.2 shows experimental data for a chloroprene rubber tested in uniaxial tension. If one tries to fit an elastic-plastic material model with isotropic hardening using three pairs of yield stress and plastic strain values, but use an initial guess of the yield stress values that are too high, then the material parameter optimization can get stuck at the solution shown in Figure 9.2.

By selecting a better initial guess of the yield stress values, the same material model and optimization algorithm can give the predictions shown in Figure 9.3, which is significantly better.

This example is somewhat trivial since it is well-known that the predicted stress-strain curves from the  $J_2$ -plasticity model with isotropic hardening is piecewise linear, and hence the predictions in Figure 9.2 are clearly not optimal. The problem illustrated in the example, however, can be a significant hurdle when working with a highly non-linear material model for which the behavior of the material model may not be clear a priori.



**Figure 9.2** Example of how a bad initial guess of the material parameters can give poor model predictions. In this case the yield stress values were too high.



**Figure 9.3** Example of how a better initial guess of the material parameters can give good model predictions. The example in this figure uses the same material model and optimization algorithm as in [Figure 9.2](#).

The two most common approaches for finding a good initial guess are:

- **The Monte Carlo method**

The Monte Carlo method [1, 2] is an optimization method in which each material parameter is first restricted to be in a certain pre-defined interval, and then for each parameter a random value in the specified range is generated. The set of parameters is then evaluated using an error measurement function, as discussed in [Section 9.4](#). The whole procedure is then repeated a number of times and the best value is used as the initial guess of the material parameters.

- **Prior knowledge of similar materials**

Having prior knowledge of the material parameters of similar materials is typically the best way to construct a good set of initial material parameters. Not only does it allow for the determination of a good set of parameters, but it is also computationally fast.

## 9.4 Error Measurement Functions

Once the material model has been used to simulate an experimental test, the next step is to evaluate the magnitude of the error of the model predictions. If the experiment was run in strain-control mode, then the simulation will follow the exact same strain history, and the difference between the model predicted and experimental stress values can be used to evaluate the accuracy of the model predictions, see [Figure 9.4](#). That is, the error of the prediction is given by:

$$error = f\left(\boldsymbol{\sigma}^{\text{pred}} - \boldsymbol{\sigma}^{\text{exp}}\right), \quad (9.2)$$

where  $\boldsymbol{\sigma}^{\text{pred}}$  is the predicted stress vector, and  $\boldsymbol{\sigma}^{\text{exp}}$  is the experimentally determined stress vector.

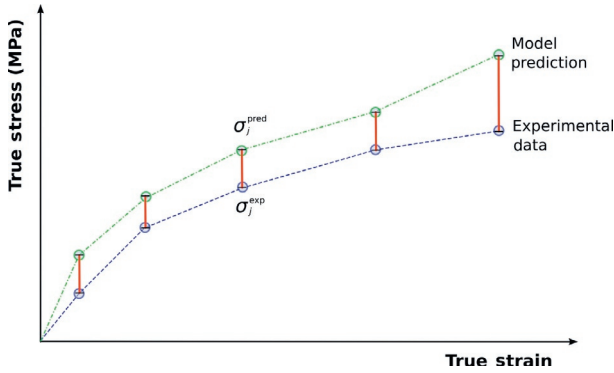


Figure 9.4 Residual error for a strain-controlled experiment.

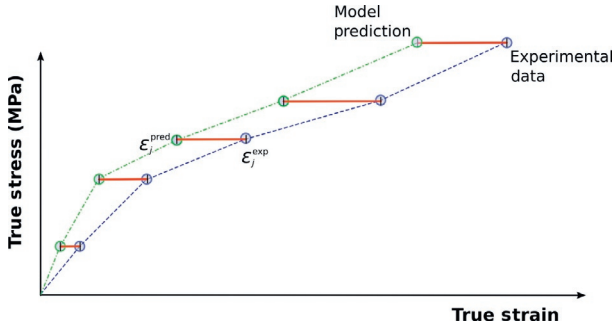


Figure 9.5 Residual error for a stress-controlled experiment.

On the other hand, if the experiment was run in load-control mode, then the simulation will follow the exact same stress history, and the difference between predicted and experimental strain values can be used to evaluate the accuracy of the model predictions, see Figure 9.5. That is, the error of the prediction is given by:

$$error = f(\epsilon^{pred} - \epsilon^{exp}), \tag{9.3}$$

where  $\epsilon^{pred}$  is the predicted strain vector, and  $\epsilon^{exp}$  is the experimentally determined strain vector.

The function  $f(\cdot)$  in Equations (9.2) and (9.3) is typically selected from one of the following:

- **Normalized Root-Mean Square Difference**

This function is defined by

$$f(\mathbf{y}^{\text{pred}}, \mathbf{y}^{\text{exp}}) = \sqrt{\langle (\mathbf{y}^{\text{pred}} - \mathbf{y}^{\text{exp}})^2 \rangle} / \sqrt{\langle (\mathbf{y}^{\text{exp}})^2 \rangle}. \quad (9.4)$$

- **Normalized Mean Absolute Difference**

This function is defined by

$$f(\mathbf{y}^{\text{pred}}, \mathbf{y}^{\text{exp}}) = \langle |\mathbf{y}^{\text{pred}} - \mathbf{y}^{\text{exp}}| \rangle / \langle |\mathbf{y}^{\text{exp}}| \rangle. \quad (9.5)$$

- **Coefficient of Determination,  $R^2$**

This function is defined by

$$f(\mathbf{y}^{\text{pred}}, \mathbf{y}^{\text{exp}}) = 1 - \frac{\sum_{i=1}^n (y_i^{\text{pred}} - y_i^{\text{exp}})^2}{\sum_{i=1}^n (y_i^{\text{exp}} - \langle \mathbf{y}^{\text{exp}} \rangle)^2}. \quad (9.6)$$

In these equations  $\langle \cdot \cdot \cdot \rangle$  is the arithmetic mean,  $\mathbf{y}^{\text{pred}}$  and  $\mathbf{y}^{\text{exp}}$  are either stress or strain vectors depending on if the experiment was run in displacement control or load control. It is important to note that all of these equations are normalized. The normalization is introduced in order to give equal weights to experimental data with different magnitudes of stress and strain. It is also possible to apply a weight function to portions of experimental data curves, or different weights to different experimental tests.

## 9.5 Algorithms for Parameter Extraction

One way to find the optimal material parameters giving the smallest difference between predicted and experimental data is to use a custom computer program written in any mathematical software or general purpose language. As an example, the Matlab function `find_material_params`, shown below, uses the root mean square error to evaluate the fitness of a set of model

predictions. The function uses an optimization algorithm based on the Nelder-Mead simplex method.

Matlab File Name: find\_material\_params.m

```
function [bestParams, error] = find_material_params(model, paramGuess, expData)
%FIND_MATERIAL_PARAMS Find the best material parameters
%Example: [res,err] = find_material_params(&NH, 1.0, expData)
[bestParams,error] = fminsearch(@(x) errorValue(model,x,expData), paramGuess);
end

function [err] = errorValue(model, params, expData)
%ERRORVALUE Calculates a scalar error value
for i = 1 : length(expData)
    predStress = model(expData(i).time, expData(i).strain, params);
    errVec(i) = sqrt(mean((predStress - expData(i).stress).^2)); % RMS error
end
err = mean(errVec);
end
```

The example shown above is using the `fminsearch` function which is a built-in Matlab implementation of the Nelder-Mead Simplex method [3]. The Simplex method is often used for material parameter extraction because of its robustness. The main weakness of this optimization algorithm is that it can be slow and have a tendency of getting stuck at a local minimum. There are a number of alternative optimization methods that have been designed to overcome these limitations. One interesting method is the Powell method [4], another is a class of optimization algorithms called genetic algorithm [5, 6].

Another option is to use a commercial material parameter extraction tool, such as the MCalibration software [7]. The goal of this chapter is to show that it is very important to have a general purpose material parameter extraction tool, and that a material parameter extraction tool can be immensely valuable for a finite element simulation engineer.

## 9.6 Exercises

1. What are the different steps that are required for calibrating a material model?
2. What methods are commonly used to determine an initial site of material parameters? What are the strengths and limitations of these methods?

3. What error measurement functions are typically used for material parameter extraction? Which error measurement function do you think is most useful?
4. Write a material parameter extraction tool in your favorite mathematical software language, and use the tool to determine the optimum Young's modulus for a linear elastic material model.

## References

- [1] R.Y. Rubinstein, D.P. Kroese. *Simulation and the Monte Carlo Method*. Wiley, New York, 2007.
- [2] C.P. Robert, G. Casella. *Monte Carlo Statistical Methods*. Springer, New York, 2005.
- [3] J.A. Nelder, R. Mead. A Simplex Method for Function Minimization. *Computer J.*, 7 1965 308-313.
- [4] R.P. Brent. *Algorithms for Minimization Without Derivatives*. Prentice-Hall, Englewood Cliffs, NJ, 1993.
- [5] M. Michelle. *An Introduction to Genetic Algorithms*. The MIT Press, Cambridge, 1998.
- [6] D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Boston, MA, 1989.
- [7] Veryst Engineering. <http://PolyUMod.com>.

# 10 User Material Subroutines

---

## Chapter Outline

10.1 Introduction	447
10.2 Abaqus/Explicit VUMAT for the Neo-Hookean Model	448
10.3 Abaqus/Implicit UMAT for the Neo-Hookean Model	450
References	454

## 10.1 Introduction

Commercial finite element (FE) codes contain a limited selection of material models that are suitable for certain materials and applications, but as has been highlighted in previous chapters it is many times desirable (or even necessary) to achieve more accurate predictions than what is possible with the built-in options. Under these circumstances the best approach is to use a suitable *user material subroutine* to represent the material response. A user material subroutine is a source code subroutine that is used by the FE software to calculate the stress for a given increment in time and deformation state. User material subroutines are typically written in Fortran, but can be written in any computer language. The purpose of the user subroutine is essentially to update the stress and the state variables during each increment, as in the following:

1. Known at time  $t$ :  $\boldsymbol{\sigma}$ ,  $\mathbf{F}$ ,  $\xi$  (state variables).
2. Known at time  $t + dt$ :  $\mathbf{F}$ .
3. Use the user material subroutine to calculate the following at  $t + dt$ :  $\boldsymbol{\sigma}$ ,  $\xi$ .



There are commercial options that work as plug-ins to the FE code. The PolyUMod library [1] that was discussed in Chapter 8 is one example. It is of course also possible to write your own user material model, and this chapter will briefly introduce how this can be done. The examples presented in this chapter are based on the Neo-Hookean hyperelastic material model with a stress given by:

$$\boldsymbol{\sigma} = \frac{\mu}{J} \operatorname{dev}[\mathbf{b}^*] + \kappa(J - 1)\mathbf{I}, \quad (10.1)$$

where  $\mu$  is the shear modulus,  $\kappa$  is the bulk modulus,  $J = \det(\mathbf{F})$ ,  $\mathbf{b}^* = J^{-2/3}\mathbf{F}\mathbf{F}^\top$  is the distortional left Cauchy-Green tensor, and  $\boldsymbol{\sigma}$  is the Cauchy stress.

The following subsections provide the code for an Abaqus/Explicit VUMAT and an Abaqus/Standard UMAT. Subroutines for other FE solver can also be written but are left as an exercise. The code examples listed below use `real` to define a floating point variable. This code can be compiled into either single or double precision simply by specifying the command line argument `-r8` or not to the Fortran compiler.

## 10.2 Abaqus/Explicit VUMAT for the Neo-Hookean Model

Abaqus/Explicit requires that the stress is returned in a co-rotational coordinate frame:

$$\hat{\boldsymbol{\sigma}} = \mathbf{R}^T \boldsymbol{\sigma} \mathbf{R}. \quad (10.2)$$

Since  $\mathbf{R}^T \mathbf{b}^* \mathbf{R} = \mathbf{R}^T (\mathbf{F}^* \mathbf{F}^{*T}) \mathbf{R} = \mathbf{R}^T (\mathbf{R} \mathbf{U}^* \mathbf{U}^{*T} \mathbf{R}^T) \mathbf{R} = \mathbf{U}^{*2}$ , the required stress becomes:

$$\hat{\boldsymbol{\sigma}} = \frac{\mu}{J} \operatorname{dev}[\mathbf{U}^{*2}] + \kappa(J - 1)\mathbf{I}. \quad (10.3)$$

These equations can be implemented in the following code:

```
c      FILE:
c      vumat_nh.f
c      AUTHOR:
c      Jorgen Bergstrom (jorgen@polymerFEM.com)
```

```

c     CONTENTS:
c     Abaqus VUMAT subroutine for the
c     Neo-Hookean (NH) model. The subroutine is an example
c     of how to write a VUMAT, and not is not designed to
c     be a commercial quality implementation.
c     USAGE:
c     2 material constants:
c     1: mu    (shear modulus)
c     2: kappa (bulk modulus)
c
c     |<- column 1 begins here
c     |
c     *User material, constants=2
c     **    mu,    kappa
c     5.0,    100.0
c     *Density
c     1000e-12
c
c     subroutine vumat(nblock, ndi, nshr, nstatev, nfieldv, nprops,
c     .   lanneal, stepTime, totTime, dt, cmname, coordMp,
c     .   charLen, props, density, Dstrain, rSpinInc, temp0,
c     .   U0, F0, field0, stressVec0, state0,
c     .   intEne0, inelaEn0, temp1, U1,
c     .   F1, field1, stressVec1, state1, intEne1, inelaEn1)
c     implicit none
c     integer nblock, ndi, nshr, nstatev, nfieldv, nprops, lanneal
c     real stepTime, totTime, dt
c     character*80 cmname
c     real coordMp(nblock,*)
c     real charLen, props(nprops), density(nblock),
c     .   Dstrain(nblock,ndi+nshr), rSpinInc(nblock,nshr),
c     .   temp0(nblock), U0(nblock,ndi+nshr),
c     .   F0(nblock,ndi+nshr+nshr), field0(nblock,nfieldv),
c     .   stressVec0(nblock,ndi+nshr), state0(nblock,nstatev),
c     .   intEne0(nblock), inelaEn0(nblock), temp1(nblock),
c     .   U1(nblock,ndi+nshr), F1(nblock,ndi+nshr+nshr),
c     .   field1(nblock,nfieldv), stressVec1(nblock,ndi+nshr),
c     .   state1(nblock,nstatev), intEne1(nblock), inelaEn1(nblock)
c
c     local variables
c     real F(3,3), B(3,3), J, t1, t2, t3, mu, kappa
c     integer i
c
c     mu = props(1)
c     kappa = props(2)
c
c     loop through all blocks
c     do i = 1, nblock
c
c     setup F (upper diagonal part)
c     F(1,1) = U1(i,1)
c     F(2,2) = U1(i,2)
c     F(3,3) = U1(i,3)
c     F(1,2) = U1(i,4)
c     if (nshr .eq. 1) then
c         F(2,3) = 0.0
c         F(1,3) = 0.0
c     else
c         F(2,3) = U1(i,5)
c         F(1,3) = U1(i,6)
c     end if
c
c     calculate J
c     t1 = F(1,1) * (F(2,2)*F(3,3) - F(2,3)**2)
c     t2 = F(1,2) * (F(2,3)*F(1,3) - F(1,2)*F(3,3))
c     t3 = F(1,3) * (F(1,2)*F(2,3) - F(2,2)*F(1,3))
c     J = t1 + t2 + t3
c     t1 = J**(-2.0/3.0)

```

```

c      Bstar = J^(-2/3) * F * Ft      (upper diagonal part)
      B(1,1) = t1*(F(1,1)**2 + F(1,2)**2 + F(1,3)**2)
      B(2,2) = t1*(F(1,2)**2 + F(2,2)**2 + F(2,3)**2)
      B(3,3) = t1*(F(1,3)**2 + F(2,3)**2 + F(3,3)**2)
      B(1,2) = t1*(F(1,1)*F(1,2)+F(1,2)*F(2,2)+F(1,3)*F(2,3))
      B(2,3) = t1*(F(1,2)*F(1,3)+F(2,2)*F(2,3)+F(2,3)*F(3,3))
      B(1,3) = t1*(F(1,1)*F(1,3)+F(1,2)*F(2,3)+F(1,3)*F(3,3))

c      Stress = mu/J * Dev(Bstar) + kappa*log(J)/J * Eye
      t1 = (B(1,1) + B(2,2) + B(3,3)) / 3.0
      t2 = mu/J
      t3 = kappa * log(J) / J
      StressVec1(i,1) = t2 * (B(1,1) - t1) + t3
      StressVec1(i,2) = t2 * (B(2,2) - t1) + t3
      StressVec1(i,3) = t2 * (B(3,3) - t1) + t3
      StressVec1(i,4) = t2 * B(1,2)
      if (nshr .eq. 3) then
         StressVec1(i,5) = t2 * B(2,3)
         StressVec1(i,6) = t2 * B(1,3)
      end if

      end do
      return
      end

```

### 10.3 Abaqus/Implicit UMAT for the Neo-Hookean Model

The Abaqus UMAT subroutine is very similar to the VUMAT presented above but it is based on the total deformation gradient and it also needs the Jacobian matrix, which is the partial derivative of the stress increment with respect to the applied strain increment.

The Jacobian for this model is given by:  $\mathbf{c} = \partial \Delta \boldsymbol{\sigma} / \partial \Delta \boldsymbol{\varepsilon}$ . Here, due to space constraints, each column of  $[\mathbf{c}]_{ij}$  is presented individually:

$$\begin{bmatrix} \mathbf{c}_{11} \\ \mathbf{c}_{21} \\ \mathbf{c}_{31} \\ \mathbf{c}_{41} \\ \mathbf{c}_{51} \\ \mathbf{c}_{61} \end{bmatrix} = \frac{\mu}{9J} \begin{bmatrix} 8b_{11}^* + 2b_{22}^* + 2b_{33}^* \\ -4b_{11}^* - 4b_{22}^* + 2b_{33}^* \\ -4b_{11}^* + 2b_{22}^* - 4b_{33}^* \\ 3b_{12}^* \\ 3b_{13}^* \\ -6b_{23}^* \end{bmatrix} + \kappa \begin{bmatrix} J \\ J \\ J \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (10.4)$$

$$\begin{bmatrix} \mathbf{c}_{12} \\ \mathbf{c}_{22} \\ \mathbf{c}_{32} \\ \mathbf{c}_{42} \\ \mathbf{c}_{52} \\ \mathbf{c}_{62} \end{bmatrix} = \frac{\mu}{9J} \begin{bmatrix} -4b_{11}^* - 4b_{22}^* + 2b_{33}^* \\ 2b_{11}^* + 8b_{22}^* + 2b_{33}^* \\ 2b_{11}^* - 4b_{22}^* - 4b_{33}^* \\ 3b_{12}^* \\ -6b_{13}^* \\ 3b_{23}^* \end{bmatrix} + \kappa \begin{bmatrix} J \\ J \\ J \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (10.5)$$

$$\begin{bmatrix} \mathbf{c}_{13} \\ \mathbf{c}_{23} \\ \mathbf{c}_{33} \\ \mathbf{c}_{43} \\ \mathbf{c}_{53} \\ \mathbf{c}_{63} \end{bmatrix} = \frac{\mu}{9J} \begin{bmatrix} -4b_{11}^* + 2b_{22}^* - 4b_{33}^* \\ 2b_{11}^* - 4b_{22}^* - 4b_{33}^* \\ 2b_{11}^* + 2b_{22}^* + 8b_{33}^* \\ -6b_{12}^* \\ 3b_{13}^* \\ 3b_{23}^* \end{bmatrix} + \kappa \begin{bmatrix} J \\ J \\ J \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (10.6)$$

$$\begin{bmatrix} \mathbf{c}_{14} \\ \mathbf{c}_{24} \\ \mathbf{c}_{34} \\ \mathbf{c}_{44} \\ \mathbf{c}_{54} \\ \mathbf{c}_{64} \end{bmatrix} = \frac{\mu}{6J} \begin{bmatrix} 2b_{12}^* \\ 2b_{12}^* \\ -4b_{12}^* \\ 3b_{11}^* + 3b_{22}^* \\ 3b_{23}^* \\ 3b_{13}^* \end{bmatrix} \quad (10.7)$$

$$\begin{bmatrix} \mathbf{c}_{15} \\ \mathbf{c}_{25} \\ \mathbf{c}_{35} \\ \mathbf{c}_{45} \\ \mathbf{c}_{55} \\ \mathbf{c}_{65} \end{bmatrix} = \frac{\mu}{6J} \begin{bmatrix} 2b_{13}^* \\ -4b_{13}^* \\ 2b_{13}^* \\ 3b_{23}^* \\ 3b_{11}^* + 3b_{33}^* \\ 3b_{12}^* \end{bmatrix}, \quad (10.8)$$

$$\begin{bmatrix} \mathbf{c}_{16} \\ \mathbf{c}_{26} \\ \mathbf{c}_{36} \\ \mathbf{c}_{46} \\ \mathbf{c}_{56} \\ \mathbf{c}_{66} \end{bmatrix} = \frac{\mu}{6J} \begin{bmatrix} -4b_{23}^* \\ 2b_{23}^* \\ 2b_{23}^* \\ 3b_{13}^* \\ 3b_{12}^* \\ 3b_{22}^* + 3b_{33}^* \end{bmatrix}. \quad (10.9)$$

```

!
! FILE:
!   umat_NH.F90
! AUTHOR:
!   Jorgen Bergstrom, Ph.D.
!   Copyright Jorgen Bergstrom.
! CONTENTS:
!   Abaqus UMAT subroutine for the Neo-Hookean (NH) model.
!   The subroutine is an example and not is not designed to
!   be a commercial quality implementation.
! USAGE:
!   2 material constants:
!     1: mu   (shear modulus)
!     2: kappa (bulk modulus)

subroutine umatErr(str)
  implicit none
  character(LEN=*) , intent(in) :: str
  print '(a,a)', "UMAT Error:  ", trim(str)
  stop 3
end subroutine umatErr

subroutine umat(strVec, statev, ddsdde, &
  energyElast, energyPlast, energyVisc, &
  rpl, ddsddt, drplde, drpldt, stran, dstran, time, dtime, &
  temp, dtemp, predef, dpred, cmname, ndi, nshr, ntens, &
  nstatev, inProps, nrInProps, coords, drot, pnewdt, celent, &
  dfgrd0, dfgrd1, noel, npt, layer, kspt, kstep, kinc)
  implicit none
  integer, intent(in) :: ndi, nshr, ntens, nstatev, nrInProps, &
    noel, npt, layer, kspt, kstep, kinc
  character(len=8), intent(in) :: cmname
  real, intent(inout) :: strVec(ntens), statev(nstatev)
  real, intent(inout) :: energyElast, energyPlast, energyVisc
  real, intent(out) :: ddsdde(ntens,ntens), rpl, &
    ddsddt(ntens), drplde(ntens), drpldt
  real, intent(in) :: stran(ntens), dstran(ntens), time(2), dtime, &
    temp, dtemp, predef(1), dpred(1)
  real, intent(in) :: inProps(nrInProps), coords(3), drot(3,3)
  real, intent(out) :: pnewdt
  real, intent(in) :: celent, dfgrd0(3,3)
  real, intent(inout) :: dfgrd1(3,3)

  ! local variables
  real :: J, a1, a2, kk, F(3,3), b(3,3), bs(3,3), Stress(3,3),
    devbs(3,3)
  real :: tmp, Eye(3,3), mu, kappa, IIs

  ! setup variables
  if (nrInProps /= 2) call umatErr("Wrong number of input parameters")
  mu = inProps(1)
  kappa = inProps(2)
  if (mu < 0) call umatErr("Parameter mu has to be positive")
  if (kappa < 0) call umatErr("Parameter kappa has to be positive")
  if (ndi /= 3 .or. nshr /= 3) call umatErr("This umat only works for
  3D elements")
  pnewdt=2.0

  ! take the time-step
  F = dfgrd1
  J = F(1,1) * (F(2,2)*F(3,3) - F(2,3)*F(3,2)) + &
    F(1,2) * (F(2,3)*F(3,1) - F(2,1)*F(3,3)) + &
    F(1,3) * (F(2,1)*F(3,2) - F(2,2)*F(3,1))

  b(1,1) = F(1,1)*F(1,1) + F(1,2)*F(1,2) + F(1,3)*F(1,3)
  b(1,2) = F(1,1)*F(2,1) + F(1,2)*F(2,2) + F(1,3)*F(2,3)

```

```

b(1,3) = F(1,1)*F(3,1) + F(1,2)*F(3,2) + F(1,3)*F(3,3)
b(2,1) = b(1,2)
b(2,2) = F(2,1)*F(2,1) + F(2,2)*F(2,2) + F(2,3)*F(2,3)
b(2,3) = F(2,1)*F(3,1) + F(2,2)*F(3,2) + F(2,3)*F(3,3)
b(3,1) = b(1,3)
b(3,2) = b(2,3)
b(3,3) = F(3,1)*F(3,1) + F(3,2)*F(3,2) + F(3,3)*F(3,3)

bs = J**(-2.0/3.0) * b

tmp = bs(1,1) + bs(2,2) + bs(3,3)
devbs = bs
devbs(1,1) = bs(1,1) - tmp/3.0
devbs(2,2) = bs(2,2) - tmp/3.0
devbs(3,3) = bs(3,3) - tmp/3.0

Eye = 0.0
Eye(1,1) = 1.0
Eye(2,2) = 1.0
Eye(3,3) = 1.0

Stress = mu/J * devbs + kappa*(J-1.0) * Eye

I1s = bs(1,1) + bs(2,2) + bs(3,3)
energyElast = 0.5*mu*(I1s - 3.0) + 0.5*kappa*(J-1.0)**2.0

! calculate the Jacobian
a1 = mu / (9.0 * J)
a2 = mu / (6.0 * J)
kk = kappa * J

ddsdde(1,1) = a1 * ( 8*bs(1,1) + 2*bs(2,2) + 2*bs(3,3) ) + kk
ddsdde(2,1) = a1 * (-4*bs(1,1) - 4*bs(2,2) + 2*bs(3,3) ) + kk
ddsdde(3,1) = a1 * (-4*bs(1,1) + 2*bs(2,2) - 4*bs(3,3) ) + kk
ddsdde(4,1) = a1 * 3*bs(1,2)
ddsdde(5,1) = a1 * 3*bs(1,3)
ddsdde(6,1) = a1 * (-6)*bs(2,3)

ddsdde(1,2) = a1 * (-4*bs(1,1) - 4*bs(2,2) + 2*bs(3,3) ) + kk
ddsdde(2,2) = a1 * ( 2*bs(1,1) + 8*bs(2,2) + 2*bs(3,3) ) + kk
ddsdde(3,2) = a1 * ( 2*bs(1,1) - 4*bs(2,2) - 4*bs(3,3) ) + kk
ddsdde(4,2) = a1 * 3*bs(1,2)
ddsdde(5,2) = a1 * (-6)*bs(1,3)
ddsdde(6,2) = a1 * 3*bs(2,3)

ddsdde(1,3) = a1 * (-4*bs(1,1) + 2*bs(2,2) - 4*bs(3,3) ) + kk
ddsdde(2,3) = a1 * ( 2*bs(1,1) - 4*bs(2,2) - 4*bs(3,3) ) + kk
ddsdde(3,3) = a1 * ( 2*bs(1,1) + 2*bs(2,2) + 8*bs(3,3) ) + kk
ddsdde(4,3) = a1 * (-6)*bs(1,2)
ddsdde(5,3) = a1 * 3*bs(1,3)
ddsdde(6,3) = a1 * 3*bs(2,3)

ddsdde(1,4) = a2 * 2*bs(1,2)
ddsdde(2,4) = a2 * 2*bs(1,2)
ddsdde(3,4) = a2 * (-4)*bs(1,2)
ddsdde(4,4) = a2 * 3*(bs(1,1) + bs(2,2))
ddsdde(5,4) = a2 * 3*bs(2,3)
ddsdde(6,4) = a2 * 3*bs(1,3)

ddsdde(1,5) = a2 * 2*bs(1,3)
ddsdde(2,5) = a2 * (-4)*bs(1,3)
ddsdde(3,5) = a2 * 2*bs(1,3)
ddsdde(4,5) = a2 * 3*bs(2,3)
ddsdde(5,5) = a2 * 3*(bs(1,1) + bs(3,3))
ddsdde(6,5) = a2 * 3*bs(1,2)

```

```
ddsdde(1,6) = a2 * (-4)*bs(2,3)
ddsdde(2,6) = a2 * 2*bs(2,3)
ddsdde(3,6) = a2 * 2*bs(2,3)
ddsdde(4,6) = a2 * 3*bs(1,3)
ddsdde(5,6) = a2 * 3*bs(1,2)
ddsdde(6,6) = a2 * 3*(bs(2,2) + bs(3,3))

! return variables:
strVec(1) = Stress(1,1)
strVec(2) = Stress(2,2)
strVec(3) = Stress(3,3)
strVec(4) = Stress(1,2)
strVec(5) = Stress(1,3)
strVec(6) = Stress(2,3)
energyPlast = 0.0
energyVisc = 0.0
rpl = 0.0
ddsddt(1) = 0.0
drplde = 0.0
drpldt = 0.0
end subroutine umat
```

## Reference

- [1] PolyUMod, <http://PolyUMod.com/>.

# 11 Material Modeling Case Studies

---

## Chapter Outline

11.1 Introduction	455
11.2 Acrylate-Butadiene Rubber	455
11.3 Chloroprene Rubber	460
11.4 Nitrile Rubber	464
11.5 Santoprene	468
11.6 High-Density Polyethylene	474
11.7 Polytetrafluoroethylene	479
11.8 Polyethylene Terephthalate	487
11.9 Polyether Ether Ketone	490
11.10 Exercises	496
References	497

## 11.1 Introduction

The previous chapters covered different experimental testing techniques, a continuum mechanics review, and a variety of different material models. This chapter will combine these topics into concrete case studies illustrating how different material models work by comparing the accuracy of the different approaches. The goal is to present, by example, what works and what does not work. Each case study covers a different material, or material class, and all results presented here were obtained using MCalibration.

## 11.2 Acrylate-Butadiene Rubber

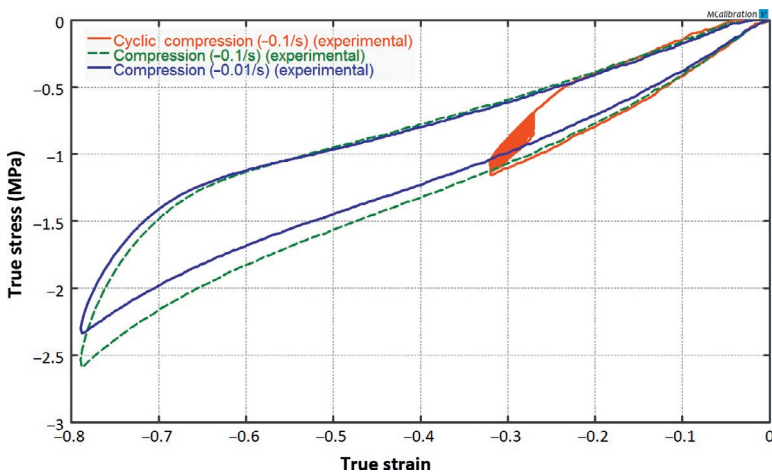
Acrylate-Butadiene rubber (ABR) is a synthetic saturated rubber that is used in sealing and packaging applications. The mechanical response of ABR is characteristic of most elastomers and



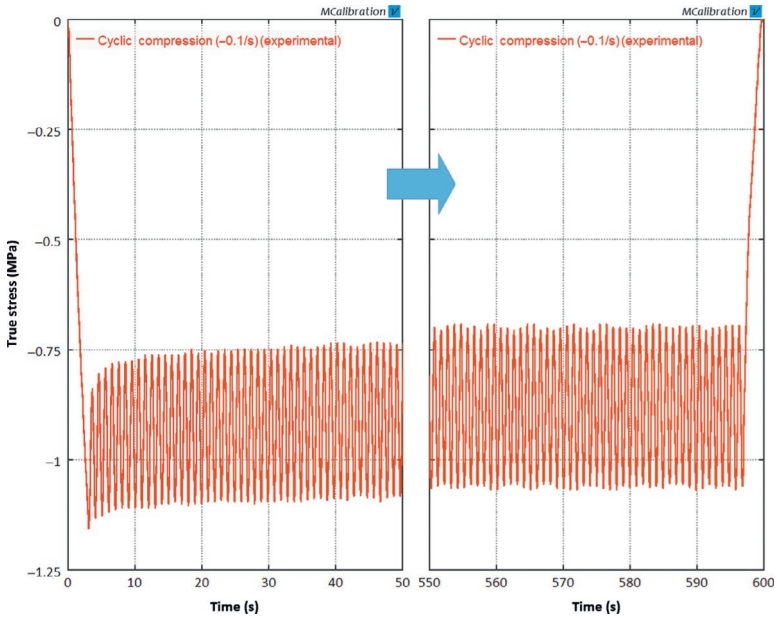
consists of non-linear viscoelasticity (non-LVE) with significant strain-rate dependence.

Figures 11.1 and 11.2 show uniaxial compressive data for a lightly filled ABR. The material was tested at two true strain rates of  $-0.01/s$  and  $-0.1/s$ , using compression followed by unloading segments. The cyclic response of the material was also examined by compressing the material to a true strain of  $-0.3$ , followed by 600 strain cycles with a strain amplitude of  $2.5\%$  at 1 Hz. The average stress in the material is shown to relax during the cyclic loading.

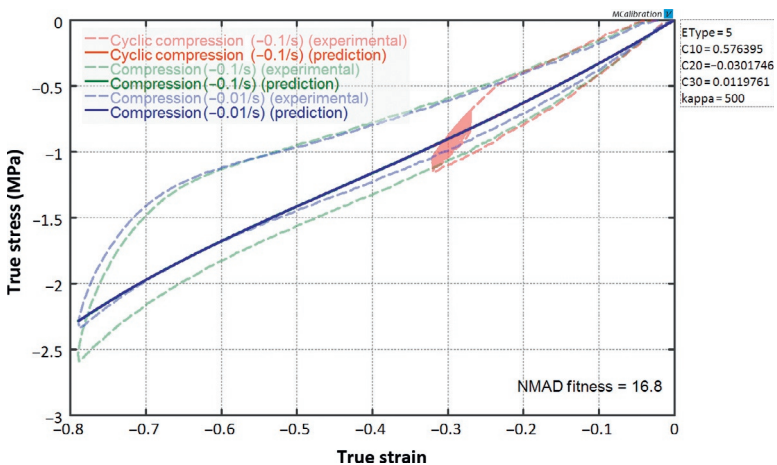
The experimentally determined mechanical response of the ABR can be represented using many different material models, some of which work better than others. Since ABR exhibits hysteresis and energy dissipation during cyclic loading one cannot expect a hyperelastic model to provide an accurate representation of the material response. Figure 11.3 shows the best possible calibration of the Yeoh hyperelastic model (see Section 5.3.8). This model has an average error (calculated using the NMAD error measured defined in Equation (9.5)) of  $17\%$ . As shown in Table 11.1, other hyperelastic models will have similar errors in the predictions.



**Figure 11.1** Uniaxial compression data for ABR at two different strain rates.



**Figure 11.2** Stress-time response of the ABR during the first 50 and the last 50 load cycles.



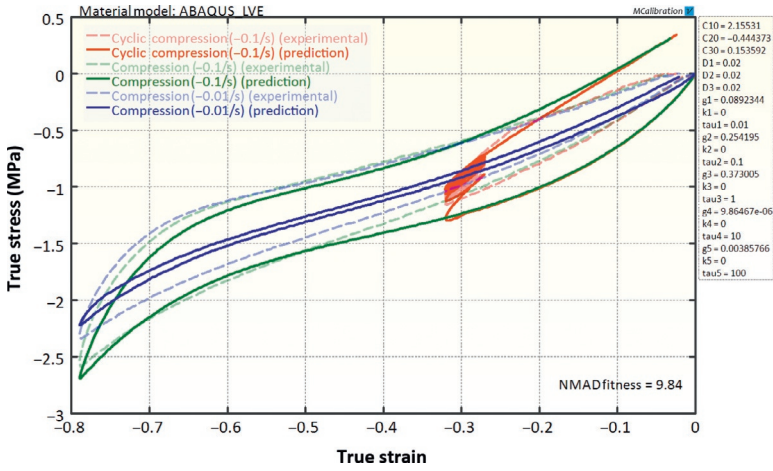
**Figure 11.3** Comparison between experimental data for ABR and the best model predictions from the Yeoh hyperelastic model.

**Table 11.1 Summary of Results from Material Model Calibrations for the ABR**

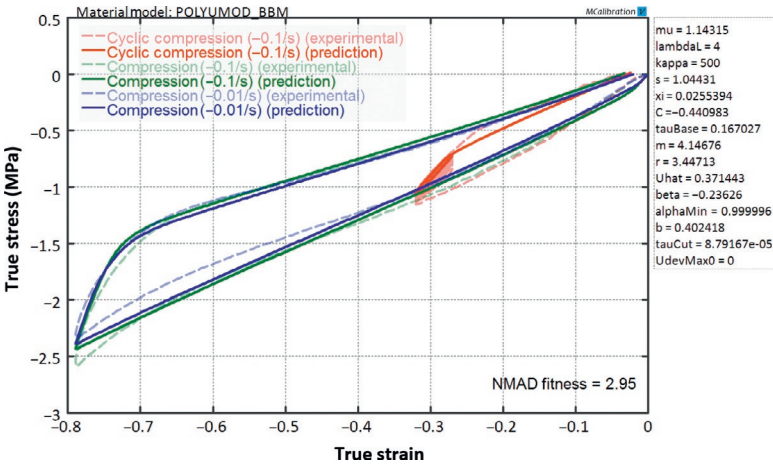
<b>Material Model</b>	<b>Error in Model Calibration (%)</b>
Arruda-Boyce eight-chain	16.9
Yeoh	16.8
Linear viscoelasticity (Yeoh, five Prony series terms)	9.8
Three Network model	6.9
Bergstrom-Boyce (BB)	6.8
BB with Mullins damage	3.0
Higher order Parallel Network models with Mullins damage	3.0

Another candidate material model is LVE combined with Yeoh hyperelasticity (see Chapter 6). Since the strains in this case are relatively large it is no surprise that LVE does not accurately capture the material response. [Figure 11.4](#) compares the experimental data with the predictions from LVE (based on Yeoh hyperelasticity and five Prony series terms). The average error is about 10%, which is better than for hyperelasticity but still not an accurate representation of the true response.

To accurately represent the response of this material requires a nonlinear viscoelastic material model. [Figure 11.5](#) shows the predictions from the Bergstrom-Boyce (BB) model (see Section 8.2) with Ogden-Roxburgh Mullins damage (see Section 5.7.1). This model does an excellent job at capturing the material response. The average error in the predictions is about 3% (as determined by the NMAD fitness value). To achieve this level of accuracy it is necessary to include Mullins damage in the model in order to capture the increased tangent stiffness right after strain reversal. It would, of course, have been nice to have additional experimental



**Figure 11.4** Comparison between experimental data for ABR and the best model predictions from a linear viscoelastic material model based on Yeoh hyperelasticity. The linear viscoelastic model was based on a five-term Prony series.



**Figure 11.5** Comparison between experimental data for ABR and the best model predictions from BB model with Mullins softening.

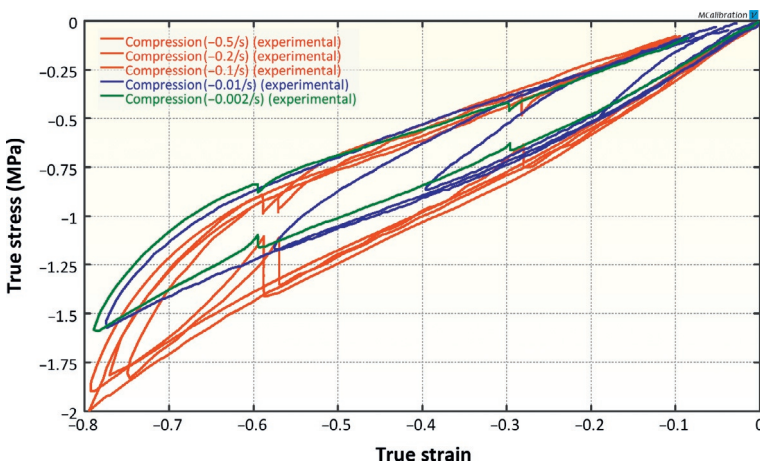
data in order to further validate the accumulation of Mullins damage at different strain levels.

The original version of the BB model (without Mullins damage) in this case has an average error of 7%. The BB model is special case of the Parallel Network (PN) model that was covered

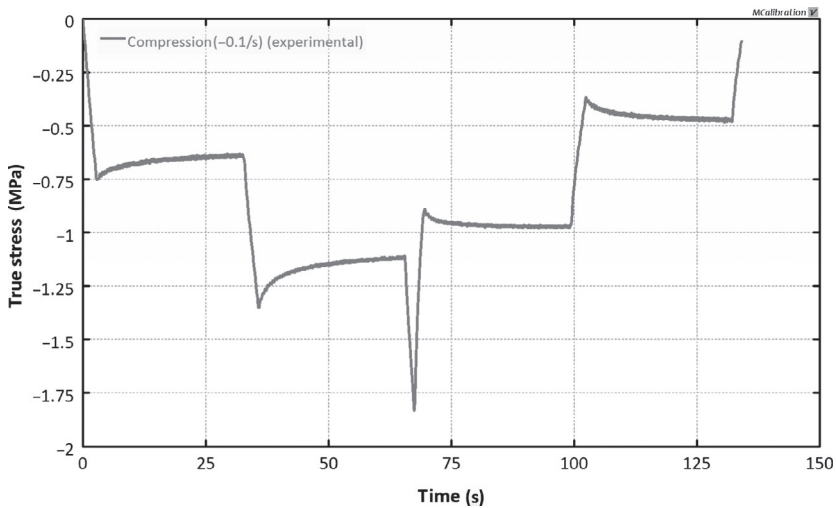
in Section 8.7 of Chapter 8 that is using two parallel networks. Adding additional network in this case does not provide any additional truthfulness to the predictions. As is often the case for rubbers, the Three Network (TN) model (see Section 8.6) is less accurate than the BB model even though it has three nonlinear viscoplastic networks compared to only two for the BB model. The reason for the superior predictions of the BB model is its Equation (8.19) for the viscoelastic flow rate that also considers the strain-dependence of the viscosity.

### 11.3 Chloroprene Rubber

Chloroprene rubber (CR) is a synthetic rubber that is also known by the trade name Neoprene. CR has a good balance of properties, including good chemical stability and usefulness over a wide temperature range. Examples of the uniaxial compressive response are summarized in Figures 11.6 and 11.7. The rubber material was tested at four different strain rates in uniaxial compression to a true strain of  $-0.8$ , followed by unloading back to zero stress. Some of the experiments contained 30 s long stress relaxation segments during both the loading and the unloading.



**Figure 11.6** Uniaxial compression data for a CR with 7 vol% carbon black.

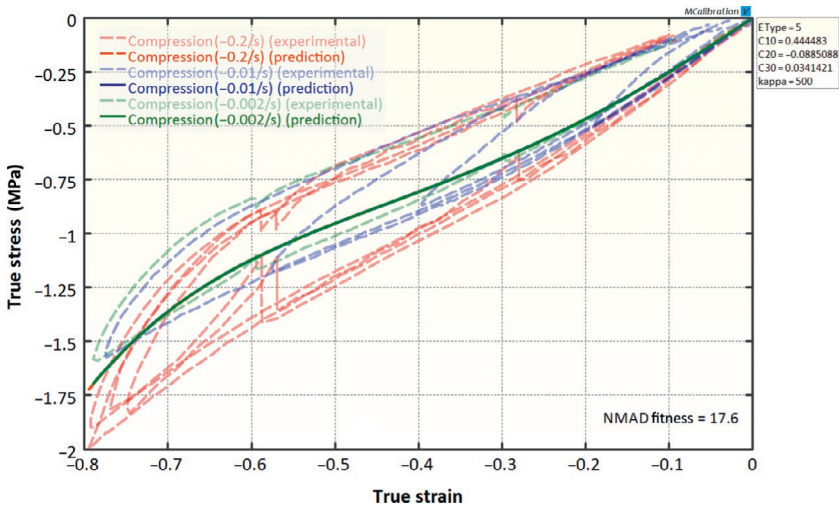


**Figure 11.7** Stress relaxation response of a CR with 7 vol% carbon black.

**Table 11.2** Summary of Results from Material Model Calibrations for the Chloroprene Rubber

Material Model	Error in Model Calibration (%)
Arruda-Boyce eight-chain	18.7
Yeoh	17.6
Linear viscoelasticity (Yeoh, five Prony series terms)	9.9
BB	5.6
BB with Mullins damage	4.4
Three Network model	6.0
Parallel Network model with BB-type flow and Mullins damage	4.2

Similar to the example in the previous section, CR can be represented using different material models. [Table 11.2](#) summarizes the accuracy of various alternative material models. Like virtually all elastomers, CR exhibits hysteresis during cyclic loading, so a

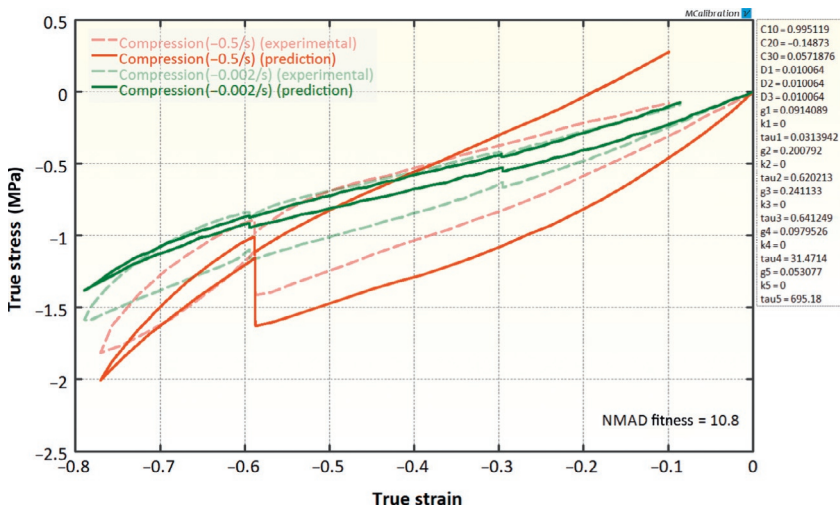


**Figure 11.8** Comparison between experimental data for CR and the best model predictions from the Yeoh hyperelastic model.

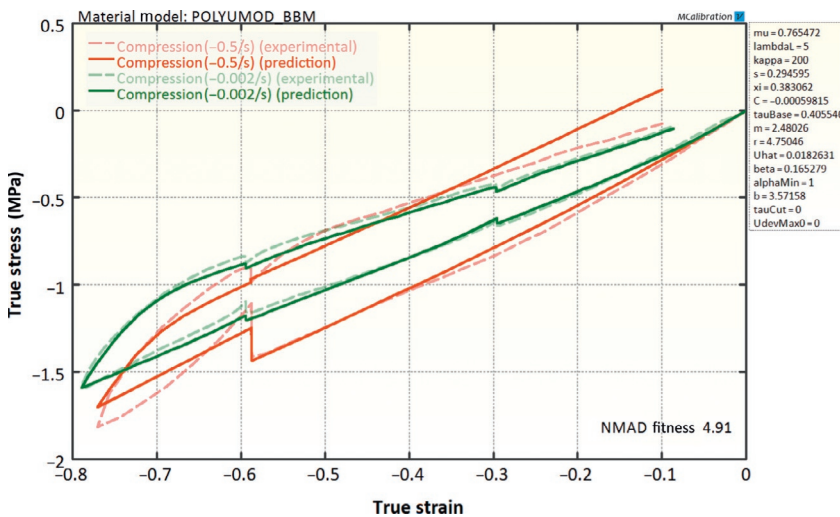
hyperelastic model is unable to accurately represent the material behavior. As is shown in [Figure 11.8](#), the best Yeoh hyperelastic model (see [Section 5.3.8](#)) has an average error of 18% (calculated using the NMAD error defined in [Equation \(9.5\)](#)). Other hyperelastic material models have similar predictive accuracy.

A slightly more advanced material model is LVE. [Figure 11.9](#) shows the predicted stress-strain response of a Yeoh hyperelasticity-based linear viscoelastic model with five Prony series terms. The error of the stress-strain predictions is about 10%. This model does not provide accurate predictions of the material response, but is more accurate than a hyperelastic model.

An accurate material model for the CR is the BB model ([Section 8.2](#)) with Ogden-Roxburgh Mullins damage ([Section 5.7.1](#)). The average error in the model predictions is about 4.4%. Also in this case, this level of accuracy can only be obtained by combining non-LVE from the BB model with Mullins damage in order to capture the increased tangent stiffness right after strain reversal. If the BB model is used without Mullins damage then the error in the model predictions is about 6% ([Figure 11.10](#)).



**Figure 11.9** Comparison between experimental data for CR and the best model predictions from a linear viscoelastic model.



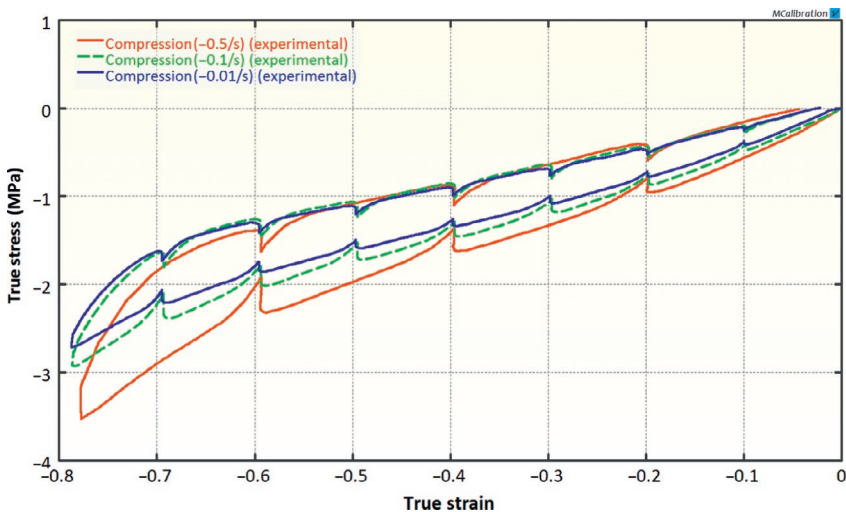
**Figure 11.10** Comparison between experimental data for CR and the best model predictions from the BB model with Mullins damage.



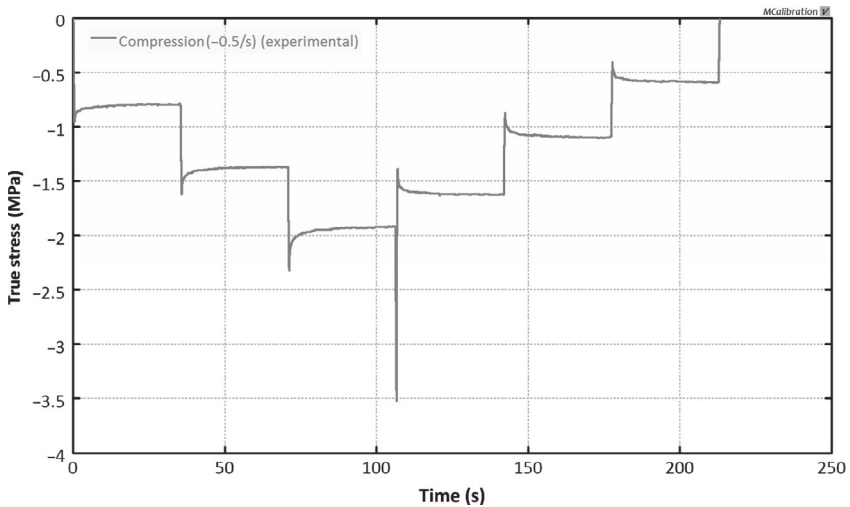
Adding one additional viscoelastic network to the BB model increases the accuracy to 4.2%. Adding further networks does not further improve the accuracy in any significant way. Also for this elastomer the TN model is less accurate than the BB model due to the lack of a flow equation with strain dependence.

## 11.4 Nitrile Rubber

Nitrile rubber, which is also called Buna-N or NBR, is an unsaturated synthetic copolymer. It is commonly used to make hoses, seals, gloves, and many other industrial products. The mechanical response of nitrile rubber can be tested in many different ways. The example shown here used uniaxial compression at different strain rates and strain cycles. [Figures 11.11](#) and [11.12](#) depict the mechanical response at three strain rates ( $-0.5$ ,  $-0.1$ , and  $-0.01$ /s). In each test, multiple stress relaxation segments were inserted. The slow strain rate test had 14 stress relaxation segments inserted, each 300 s long. The two faster tests had 30 s long stress relaxation segments. As is typically seen with rubbers, the stress magnitude goes down during the stress relaxation during the loading phase, and the stress magnitude goes up during the stress relaxation during the unloading phase ([Table 11.3](#)).



**Figure 11.11** Uniaxial compression data for a nitrile rubber.

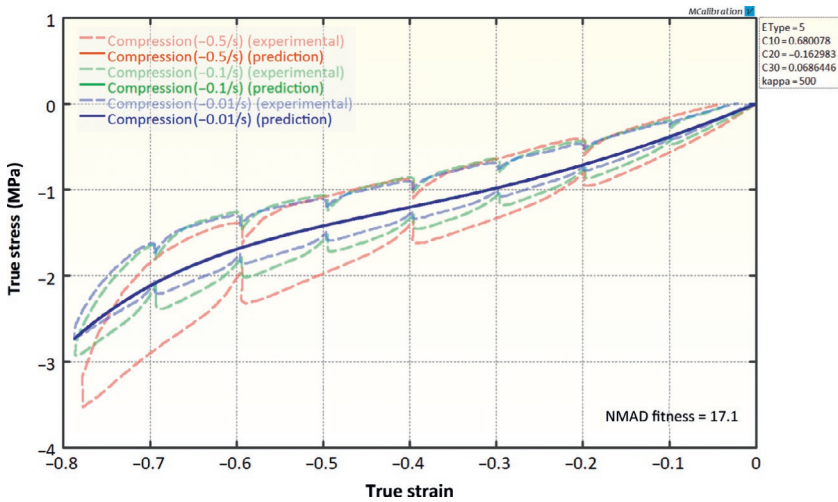


**Figure 11.12** Stress-time response of the nitrile rubber tested at a true strain rate of  $-0.5/s$ .

**Table 11.3** Summary of Results from Material Model Calibrations of the Nitrile Rubber

Material Model	Error in Model Calibration (%)
Arruda-Boyce eight-chain	17.7
Yeoh	17.1
Linear viscoelasticity (Yeoh)	9.8
BB	5.3
BB with Mullins damage	4.6
Higher order Parallel Network models with Mullins damage	4.6

Like the other two elastomers discussed in this chapter, nitrile rubber exhibits significant viscoelasticity and hysteresis making hyperelastic models unable to capture the experimental response. [Figure 11.13](#) shows that hyperelastic models have an error of

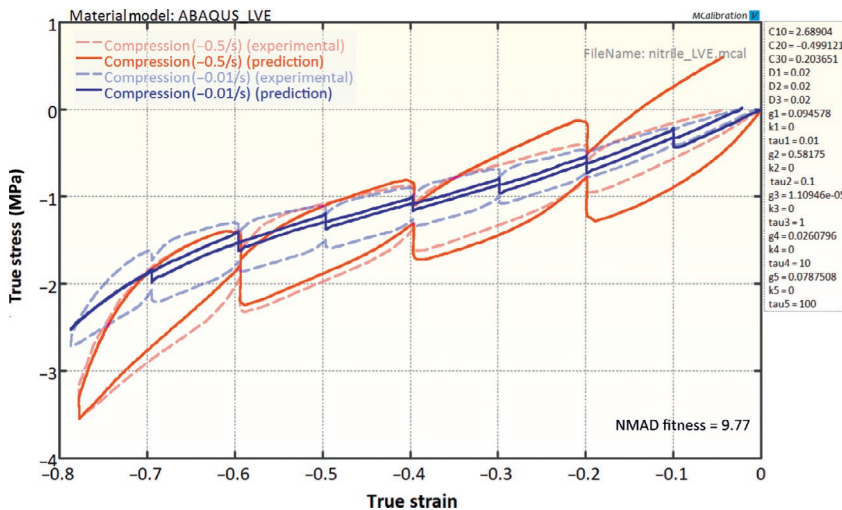


**Figure 11.13** Comparison between experimental data for nitrile rubber and the best model predictions from the Yeoh hyperelastic model.

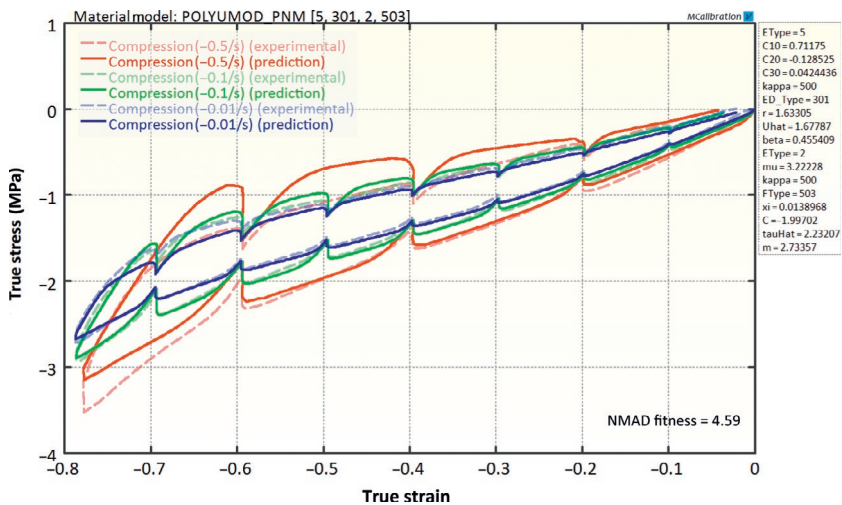
approximately 17% (calculated using the NMAD error defined in Equation (9.5)).

The next step after trying a hyperelastic model is to examine a linear viscoelastic material model. Figure 11.14 shows a comparison between the experimental data for the nitrile rubber and the best predictions from a linear viscoelastic material model based on Yeoh hyperelasticity and a five-term Prony series representation. Since the material response is nonlinear viscoelastic, a linear viscoelastic model cannot accurately represent the material response. The average error is about 9.0%.

The stress-strain response of the nitrile rubber can be accurately represented using the BB model (Section 8.2) with Ogden-Roxburgh Mullins damage (Section 5.7.1). This model accurately represents the material response. The average error of the predictions is about 4.6% (as determined by the NMAD fitness value). The original BB model without Mullins damage in this case has an error of about 5.3%. Adding additional nonlinear viscoelastic networks to the material model framework does not improve the predicted accuracy (Figure 11.15).



**Figure 11.14** Comparison between experimental data for nitrile rubber and the best model predictions from a linear viscoelastic material model based on Yeoh hyperelasticity. The linear viscoelastic model was based on a five-term Prony series.



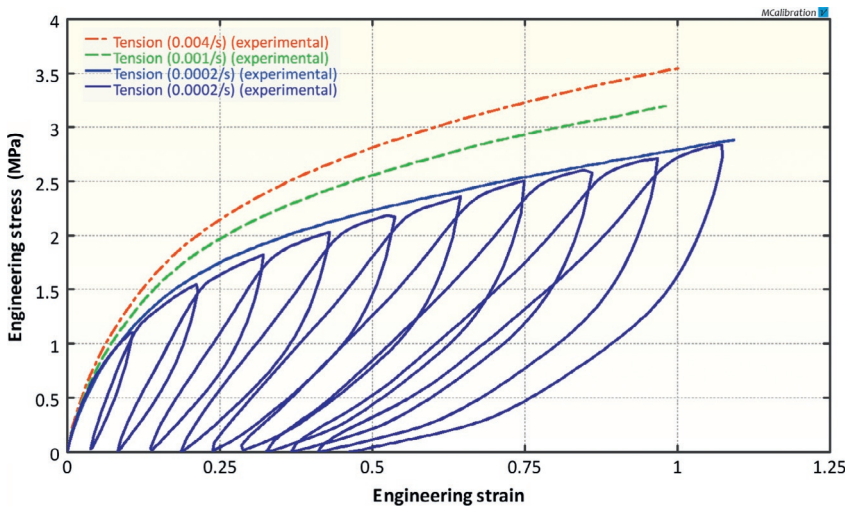
**Figure 11.15** Comparison between experimental data for nitrile rubber and the best model predictions from the BB model with Mullins softening.

## 11.5 Santoprene

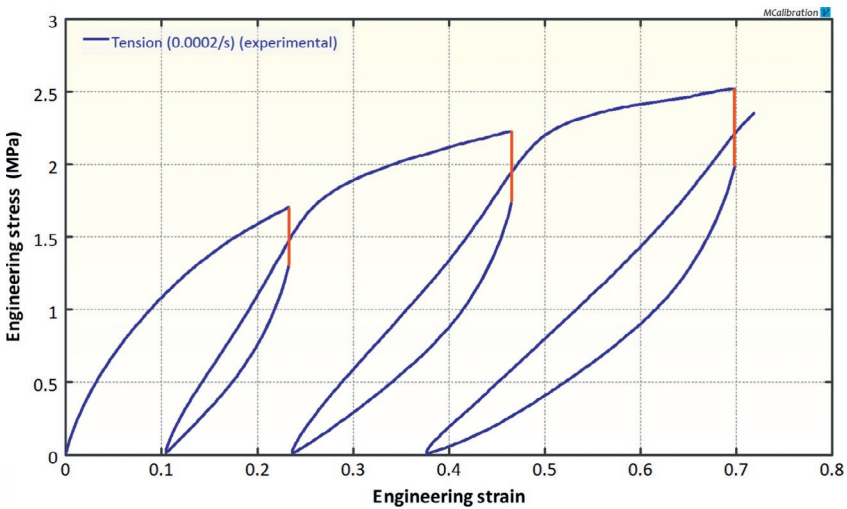
Santoprene is a thermoplastic vulcanizates (TPV) consisting of EPDM rubber particles inside a polypropylene (PP) matrix. Due to this combination, Santoprene has properties that are similar to both elastomers and thermoplastics. Figure 11.16 shows uniaxial tensile stress-strain data at three different strain rates. One of the tests in this figure contained 10 load-unload cycles at intermediate strains. Figure 11.17 shows the results of another cyclic loading experiment on the same material. In this case, the strain was held constant for 10 min at three strain levels. The relaxation segments, indicated by red lines in the figure, show that the material undergoes significant relaxation when the strain is held constant.

The accuracy of many different material models to represent the experimental data of the Santoprene are examined in this section. Table 11.4 summarizes the findings.

Figure 11.18 shows that a hyperelastic material model cannot represent the viscoplastic response of a Santoprene. This figure shows that a Yeoh hyperelastic model has an average error in



**Figure 11.16** Uniaxial tension data for a Santoprene at three different strain rates.



**Figure 11.17** Uniaxial tension data for a Santoprene. The red-line segments show the stress relaxation behavior at three strain levels.

the stress-strain predictions of about 41%. Hence, all hyperelastic models are inadequate for anything but the most basic FE study.

LVE is a more promising candidate model framework, see [Figure 11.19](#)). The LVE model is here based on Yeoh hyperelasticity and a five-term Prony series. The average error of the model predictions is 16.3%.

Since the matrix material of Santoprene is PP it is worthwhile to examine the utility of metal plasticity models. [Figure 11.20](#) shows the predictions from an isotropic hardening plasticity model with rate-dependence. Specifically, the following Abaqus keywords were used:

```
*Elastic
*Plastic, hardening=isotropic
*Rate Dependent, type=power law
```

The average error of the model calibration is 18%.

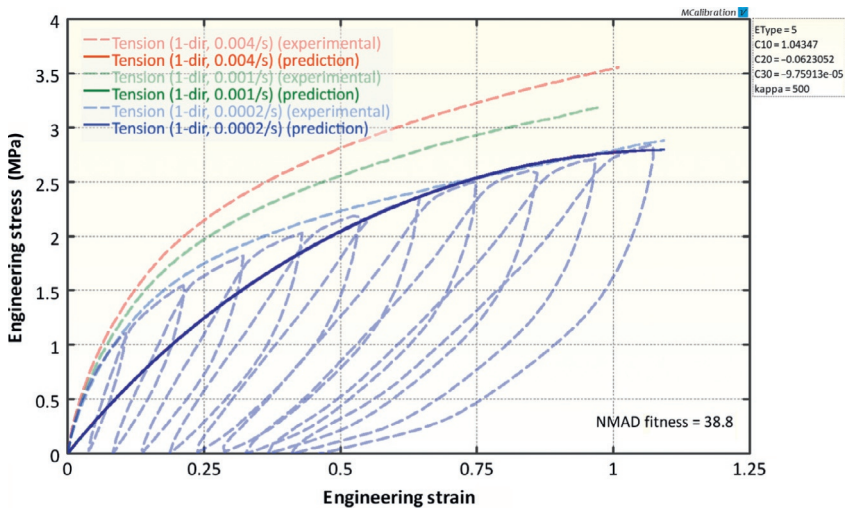
In this example, the material is exposed to cyclic loading, it is therefore more appropriate to use a kinematic hardening plasticity model. [Figure 11.21](#) shows the predicted response of an elastic-plastic material model with combined kinematic hardening and one backstress network. The average error of this model is 19%.

**Table 11.4 Summary of Results from Material Model Calibrations for Santoprene**

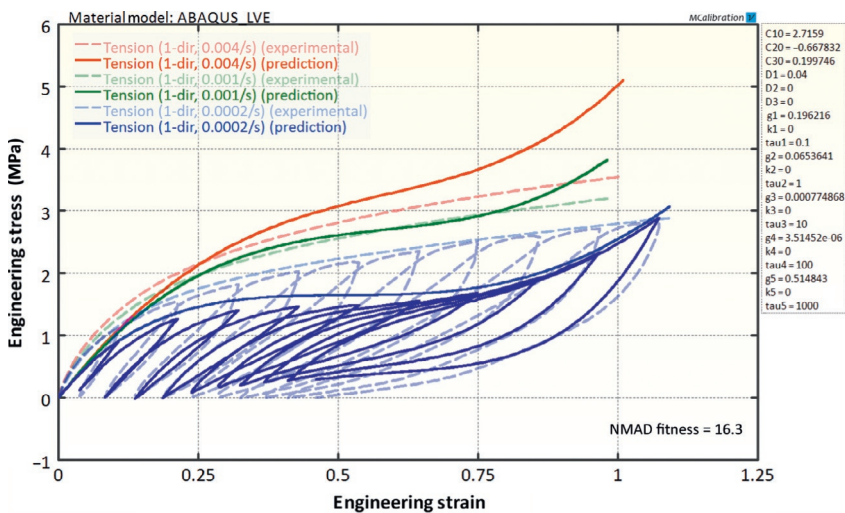
<b>Material Model</b>	<b>Error in Model Calibration (%)</b>
Arruda-Boyce eight-chain	42.5
Yeoh	38.8
Elastic-plastic with kinematic hardening (one backstress network)	18.6
Elastic-plastic with isotropic hardening and rate-dependence	17.6
Linear viscoelasticity (Yeoh)	16.3
Three Network model	9.8
Dual Network Fluoropolymer (DNF) model	9.5
Elastic-plastic with kinematic hardening and rate-dependence (three backstress networks)	9.0
BB	7.5
BB with Mullins damage	5.6
Parallel Network model with three networks of BB type and Mullins damage	2.7
Parallel Network model with four or more networks of BB type and Mullins damage	2.7

The accuracy of the material model can be enhanced by using three backstress networks and including plastic strain creep. As shown in [Figure 11.22](#), the average of this model is 9.0%.

A completely different modeling approach is provided by the BB model, and multi-network versions of this model. [Figure 11.23](#) shows that the basic BB model captures the experimental data reasonably well with an average error of the predictions of 7.5%. This quality of the predictions of this model can be strengthened by including Mullins damage through the

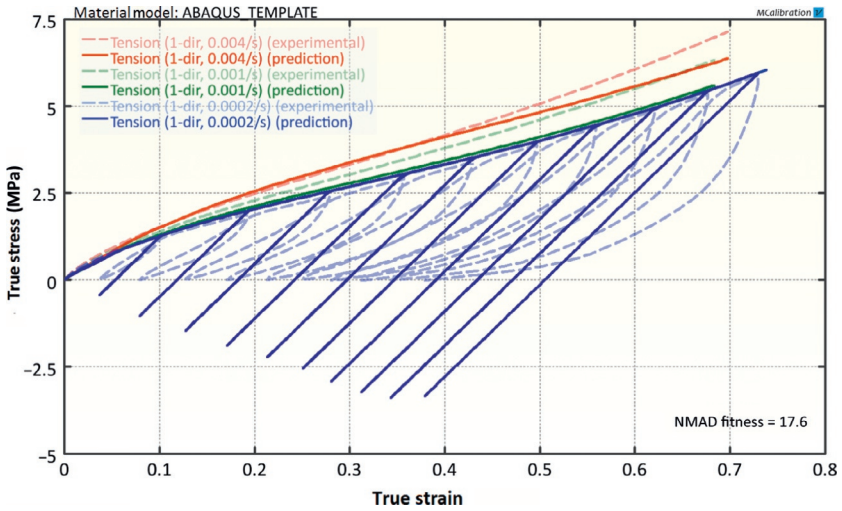


**Figure 11.18** Comparison between experimental data for Santoprene and the best model predictions from the Yeoh hyperelastic model.

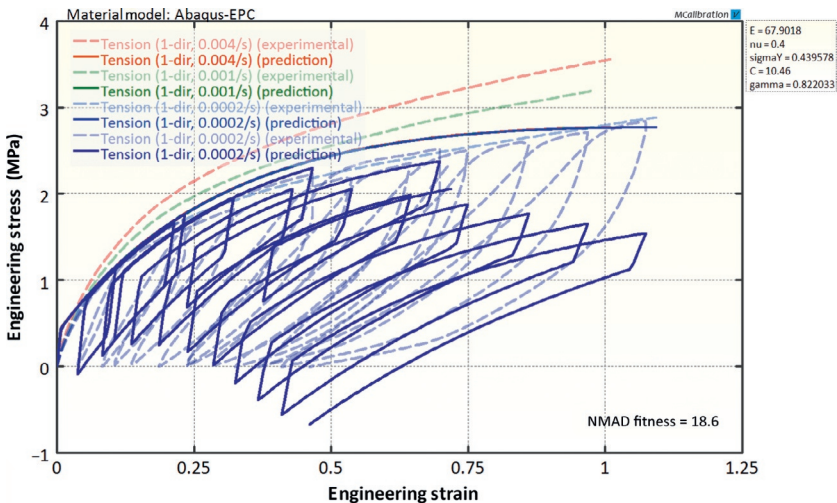


**Figure 11.19** Comparison between experimental data for Santoprene and the best model predictions from a linear viscoplastic model.

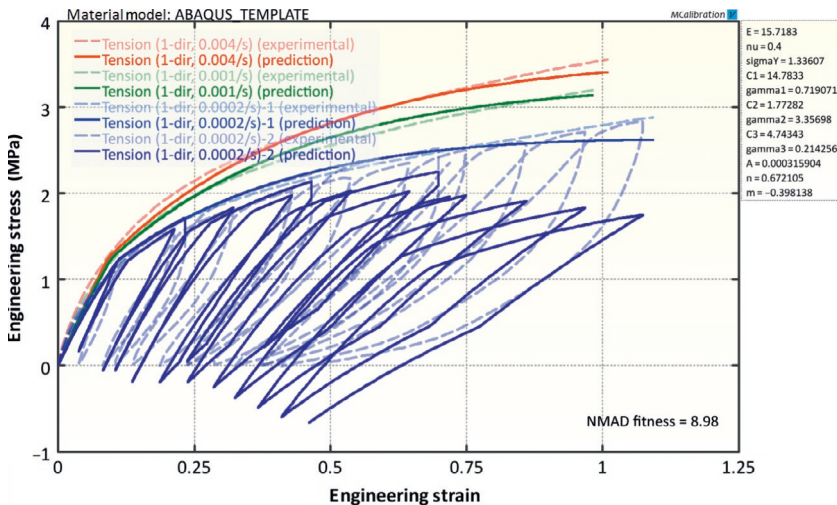




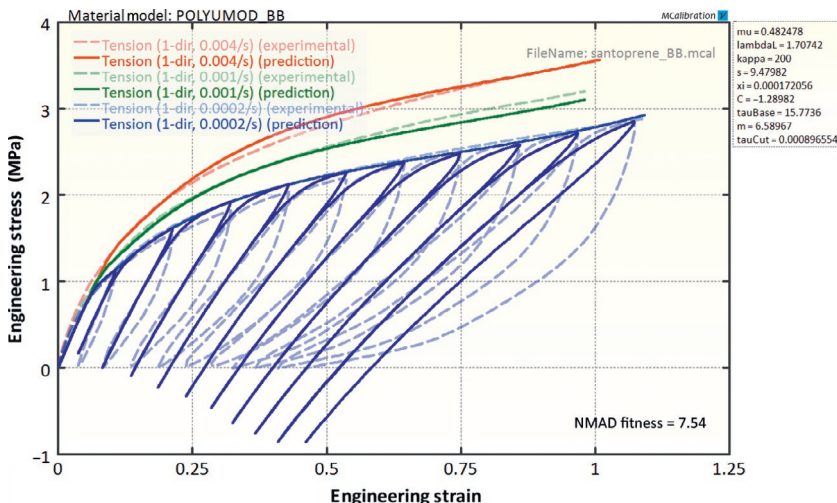
**Figure 11.20** Comparison between experimental data for Santoprene and the best model predictions from an elastic-plastic material model with isotropic hardening and rate-dependence.



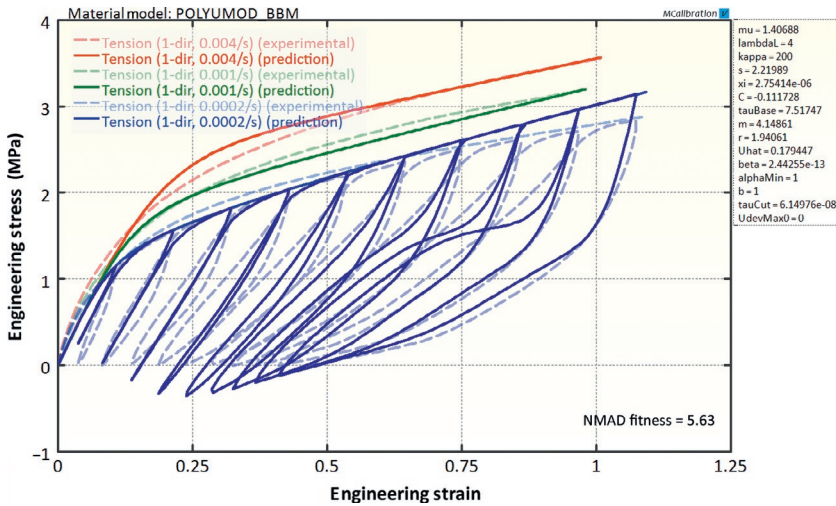
**Figure 11.21** Comparison between experimental data for Santoprene and the best model predictions from an elastic-plastic material model with combined kinematic hardening and one backstress network.



**Figure 11.22** Comparison between experimental data for Santoprene and the best model predictions from an elastic-plastic material model with combined kinematic hardening, plastic creep, and three backstress networks.



**Figure 11.23** Comparison between experimental data for Santoprene and the best model predictions from the BB model.



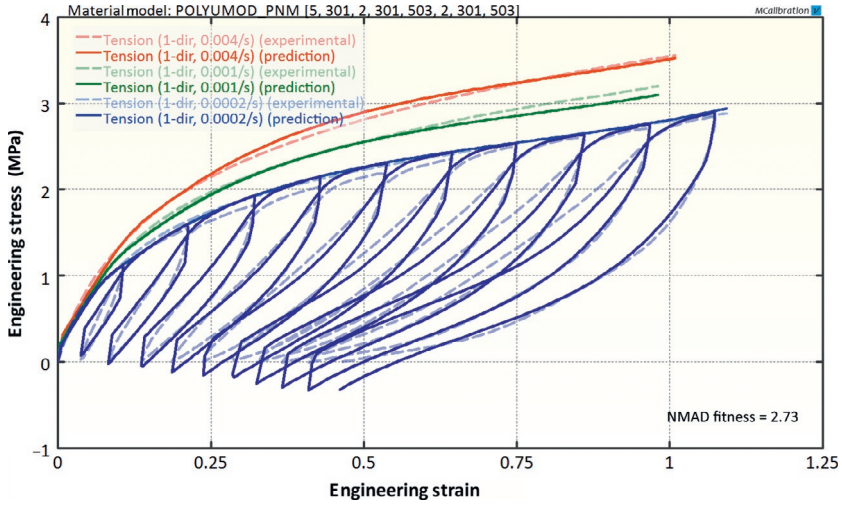
**Figure 11.24** Comparison between experimental data for Santoprene and the best model predictions from the BB model with Mullins damage.

Ogden-Roxburgh approach. [Figure 11.24](#) illustrates a significantly improved visual appearance of the predicted response. The average error for this model is 5.6%.

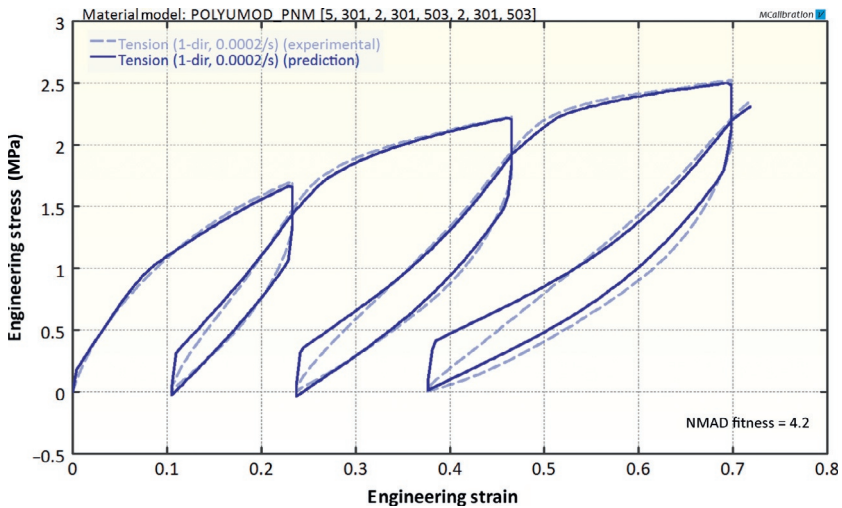
The most accurate material model in this case is the three network extension of the BB model with Mullins damage. That is, the model framework consists of three parallel networks where the first network is a simple Yeoh hyperelastic network, the second and third networks consist of a Neo-Hookean spring in series with a BB viscoplastic flow element. The predicted response of this model is shown in [Figures 11.25](#) and [11.26](#). The average error of this model calibration is 2.7%.

## 11.6 High-Density Polyethylene

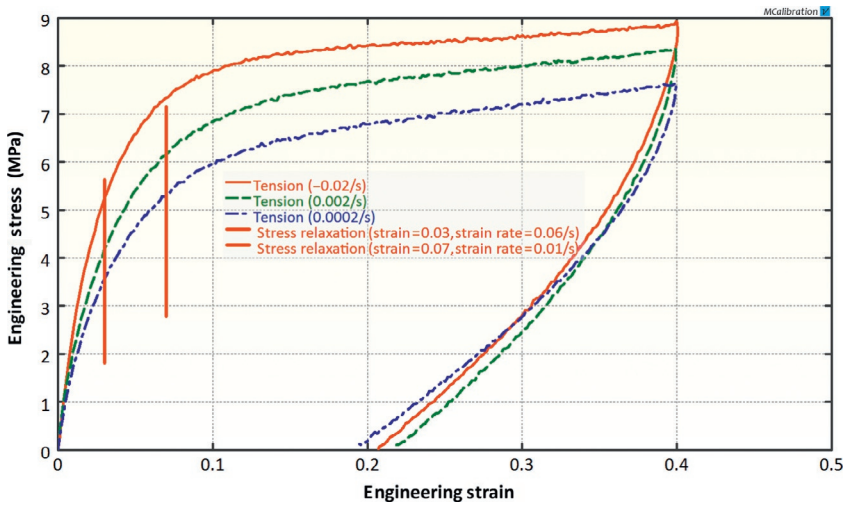
High-density polyethylene (HDPE) is a semicrystalline polymer that is commonly used in both industrial and consumer products. The mechanical response of HDPE is similar to many soft thermoplastics in that it starts to undergo viscoplastic deformations at very small strains. [Figure 11.27](#) shows the stress-



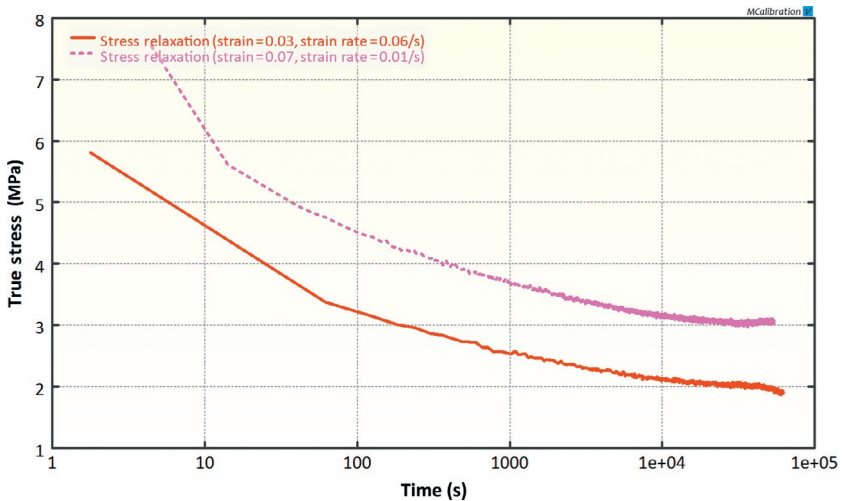
**Figure 11.25** Comparison between experimental data for Santoprene and the best model predictions from the PN model with three networks with BB type flow and Mullins damage.



**Figure 11.26** Comparison between experimental data for Santoprene and the best model predictions from the PN model with three networks with BB type flow and Mullins damage.



**Figure 11.27** Uniaxial tension data for a HDPE.



**Figure 11.28** Stress relaxation data for a HDPE.

strain response of a HDPE tested in uniaxial tension at three different strain rates. The figure also shows the stress relaxation response when loaded to 3% and 7% engineering strain for 15 h. Under these conditions the stress relaxes by more than 50% (Figure 11.28).

**Table 11.5 Summary of Results from Material Model Calibrations for HDPE**

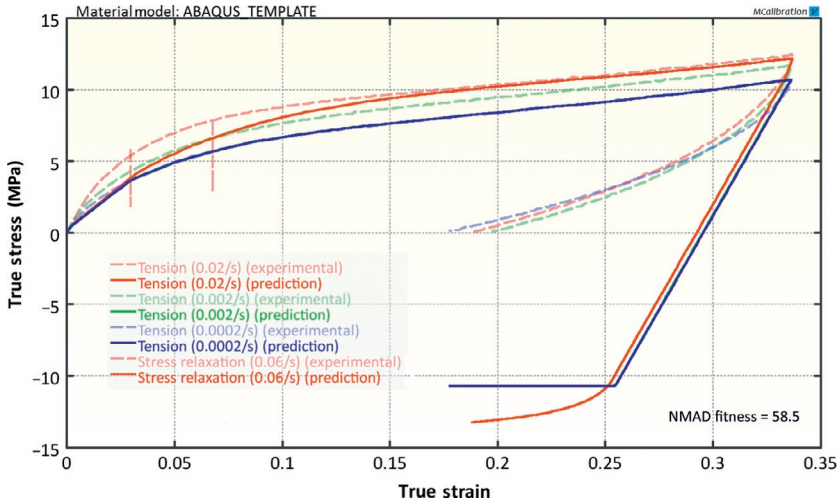
<b>Material Model</b>	<b>Error in Model Calibration (%)</b>
Elastic-plastic with isotropic hardening and rate-dependence	58.5
Arruda-Boyce eight-chain	38.1
Yeoh	27.2
Linear viscoelasticity (Yeoh)	37.2
BB	8.0
Parallel Network model with two networks (Power flow)	7.9
Elastic-plastic with kinematic hardening and rate-dependence (three backstress)	7.8
DNF model	7.6
Three Network model	5.2
Parallel Network model with three networks (Power flow)	5.0
Parallel Network model with four networks (Power flow)	4.8
Parallel Network model with five networks (Power flow)	4.7

As is summarized in [Table 11.5](#), there are many different candidate material models than can be used to model the response of HDPE with varying degree of success.

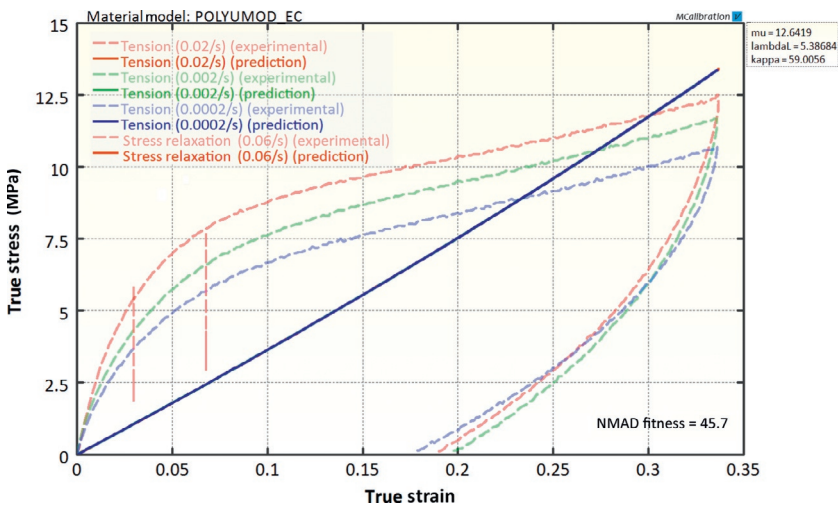
As a first example, [Figure 11.30](#) presents the predictions from an elastic-plastic material model with isotropic hardening and a yield stress that depends on the plastic strain rate (see [Section 7.2](#)). This material model can represent the monotonic tension response at different strain rates, but since it is based

on isotropic hardening the unloading response is quite inaccurate (Figure 11.29).

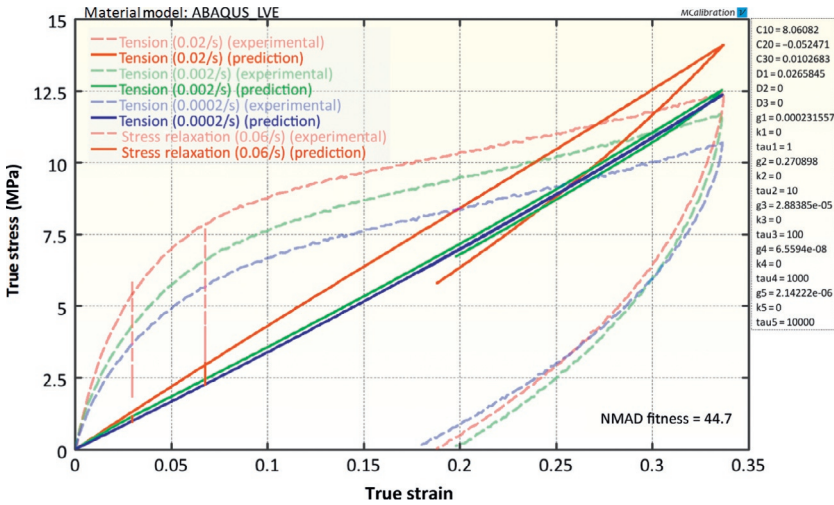
Similarly, a hyperelastic material model is not adequate for HDPE. Figure 11.30 illustrates that the Arruda-Boyce eight-chain model is not suitable for HDPE.



**Figure 11.29** Comparison between experimental data for HDPE and the best model predictions from an elastic-plastic material model with combined kinematic hardening and one backstress network.



**Figure 11.30** Comparison between experimental data for HDPE and the best model predictions from the Arruda-Boyce eight-chain model.



**Figure 11.31** Comparison between experimental data for HDPE and the best model predictions from a linear viscoplastic model.

Due to the large amount of plastic deformation that occurs in the material at finite deformations it is not possible to use a linear viscoelastic material model for the HDPE data (see Figure 11.31).

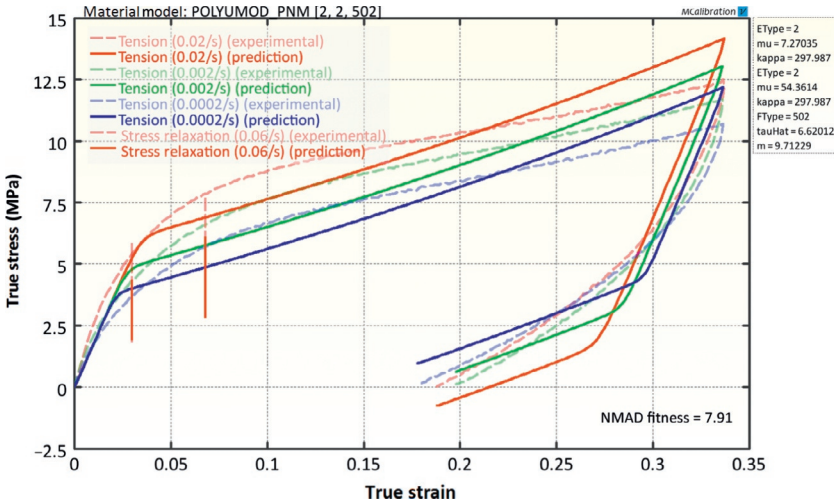
The experimental data for the HDPE can be accurately captured using the PN model (see Section 8.7). Figures 11.32–11.35 demonstrate that the accuracy of the model calibration increases with increasing number of networks in the model representation. In this example, the hyperelastic components of the PN model were taken as Yeoh elements, and the viscoplastic flow elements were taken as a simple Power-flow model (see Equation (8.31)).

## 11.7 Polytetrafluoroethylene

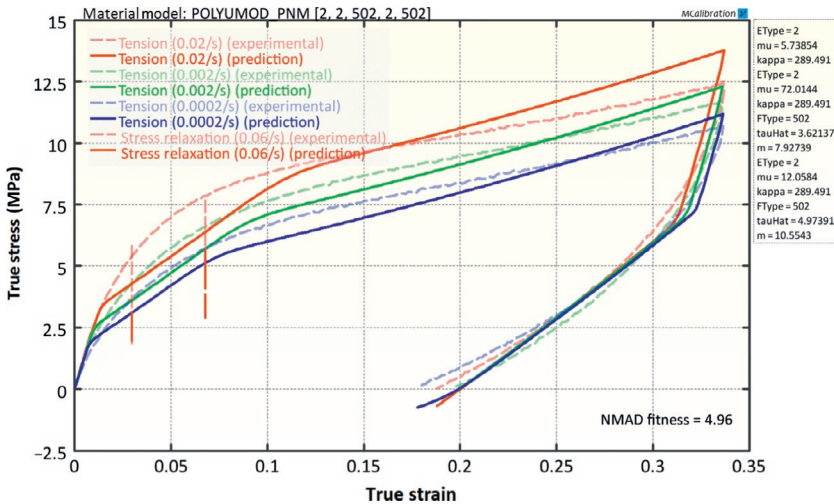
Polytetrafluoroethylene (PTFE) is a solid fluorocarbon polymer containing only carbon and fluorine. PTFE has a lower friction coefficient than most other materials, and it has excellent dielectric properties. It is commonly used in many industrial and medical device applications.

The mechanical behavior of PTFE is characterized by significant nonlinearities in terms of applied loading rates and loading



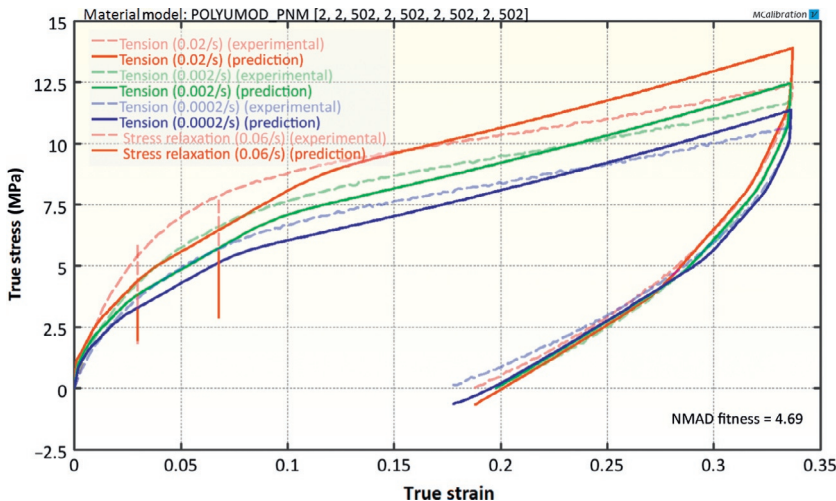


**Figure 11.32** Comparison between experimental data for HDPE and the best model predictions from the PN model with two networks with Power-law type flow.

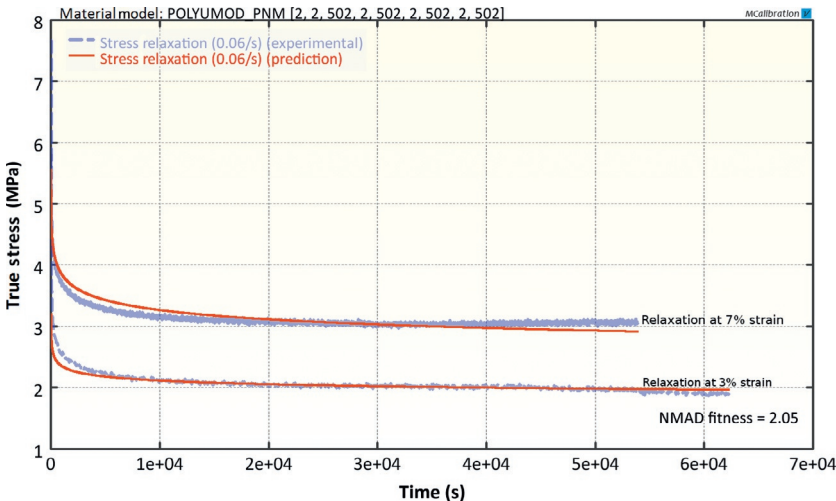


**Figure 11.33** Comparison between experimental data for HDPE and the best model predictions from the PN model with three networks with Power-law type flow.

history. Figures 11.36–11.38 show stress-strain data for a PTFE material filled with 10 vol% glass fibers [1]. As is typical for PTFE the yield stress is significantly higher in compression than in tension. This difference in behavior between tension and

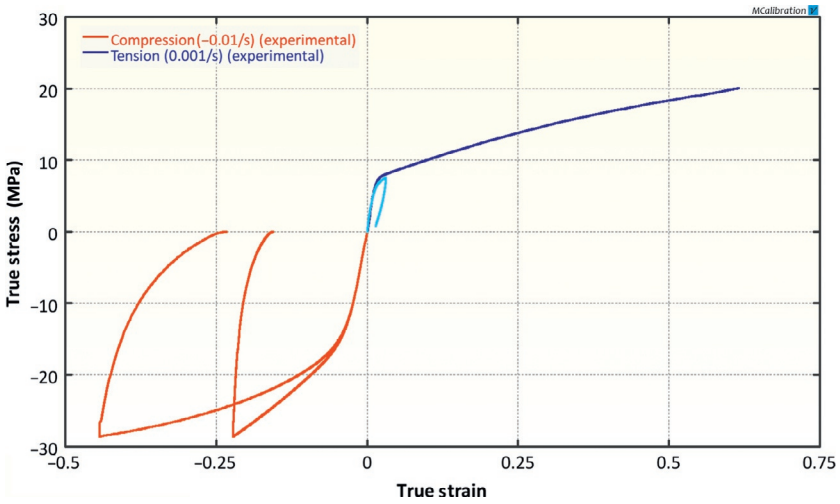


**Figure 11.34** Comparison between experimental data for HDPE and the best model predictions from the PN model with five networks with Power-law type flow.

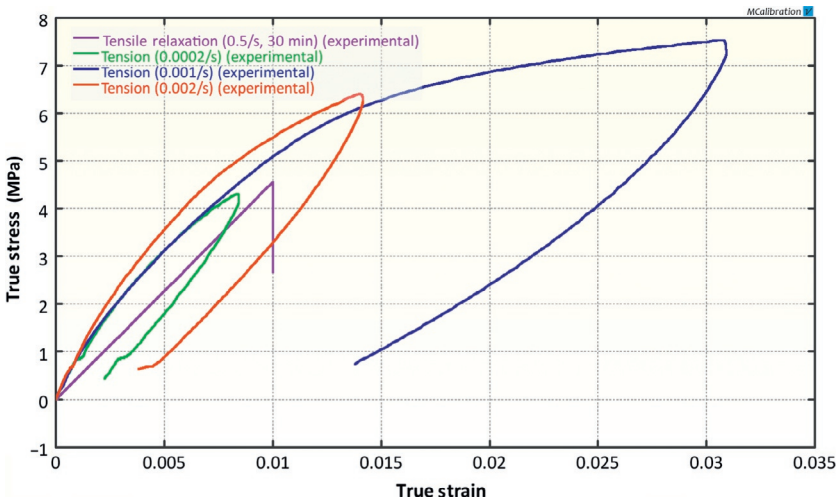


**Figure 11.35** Comparison between experimental data for HDPE and the best model predictions from the PN model with five networks with Power-law type flow.

compression is partly caused by a small amount of microporosity that is characteristic of PTFE. These figures also show that both the deviatoric and volumetric response undergoes viscoplastic relaxation under the tested conditions.



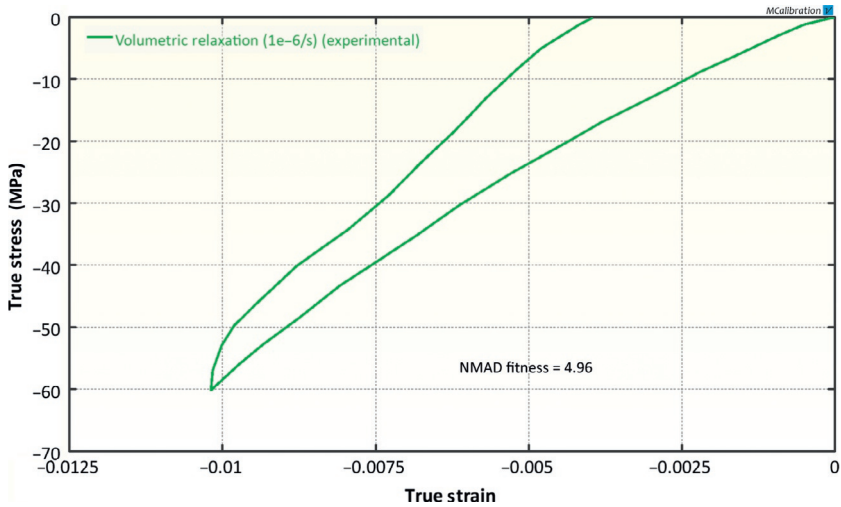
**Figure 11.36** Uniaxial tension and compression data for a PTFE material. The yield stress is higher in compression than in tension.



**Figure 11.37** Uniaxial tension data for a PTFE material tested at multiple strain rates. The tests include loading, unloading, and stress relaxation segments.

The predictive capabilities of a number of different candidate material models are summarized in [Table 11.6](#).

This comprehensive experimental data set containing both tension and compression data at different strain rates is challenging for many material models to capture. An elastic-plastic



**Figure 11.38** Volumetric compression data for a PTFE material.

material model, in this case without strain-rate dependence, is clearly unable to represent the material response as shown in [Figure 11.39](#). One of the main issues with this material model is that it is unable to distinguish between flow resistance in tension and compression. Note that all experimental data and model predictions are plotted in the same graph in order to get one comprehensive view of the data.

A significantly more accurate representation of the data is achieved by a two-network PN model where each network consists of a Neo-Hookean hyperelastic component in series with a Power-law flow element with pressure dependence. The predictions of this model are shown in [Figure 11.40](#). The average error in the model predictions is 16%.

An elastic-plastic material model with combined kinematic hardening and creep-based rate-dependence captures the experimental data surprisingly well in this case. As shown in [Figure 11.41](#), this model has an average error of about 16.3%.

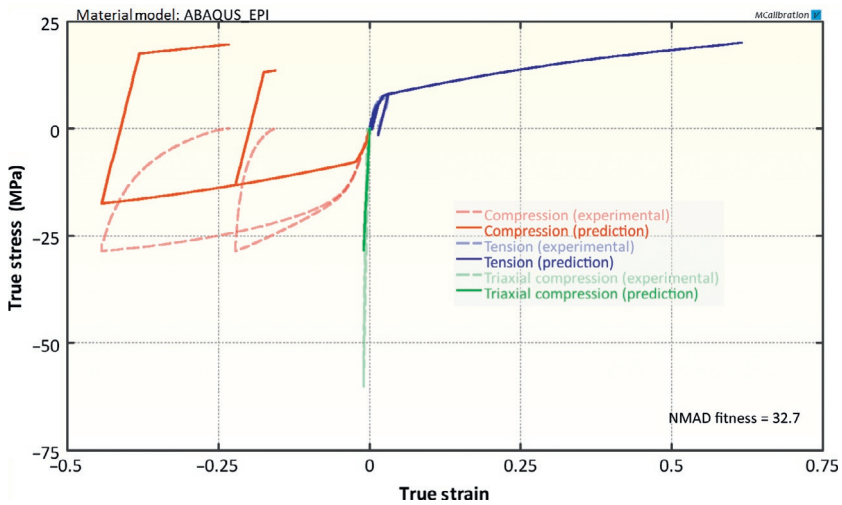
The DNF model is specifically designed for fluoropolymers and as shown in [Figure 11.42](#) reasonably accurately captures the complete experimental data setup for the fiber filled PTFE. The main limitation of the DNF model is that it under predicts

**Table 11.6 Summary of Results from Material Model Calibrations for HDPE**

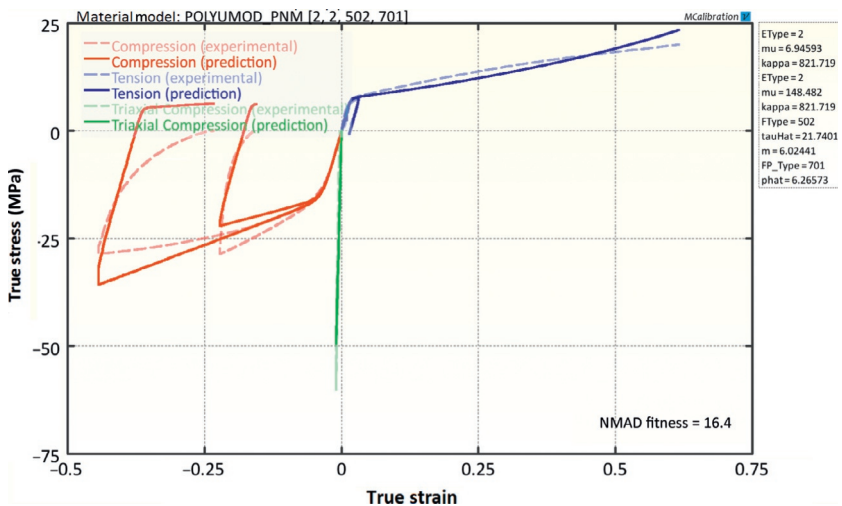
<b>Material Model</b>	<b>Error in Model Calibration (%)</b>
Neo-Hookean	79.0
Arruda-Boyce eight-chain	72.0
Linear viscoelasticity (Yeoh)	52.5
Yeoh	49.6
Elastic-plastic with isotropic hardening	32.7
ANSYS Chaboche with Perzyna rate-dependence	29.0
BB	19.7
Parallel Network model with two networks (Power flow)	16.4
Elastic-plastic with kinematic hardening and rate-dependent creep (three backstress)	16.3
DNF model	14.8
Parallel Network model with three networks (Power flow)	14.0
Parallel Network model with four networks (Power flow)	13.5
Three Network model	10.8

the recovery during unloading. The average error in the model predictions is 15%.

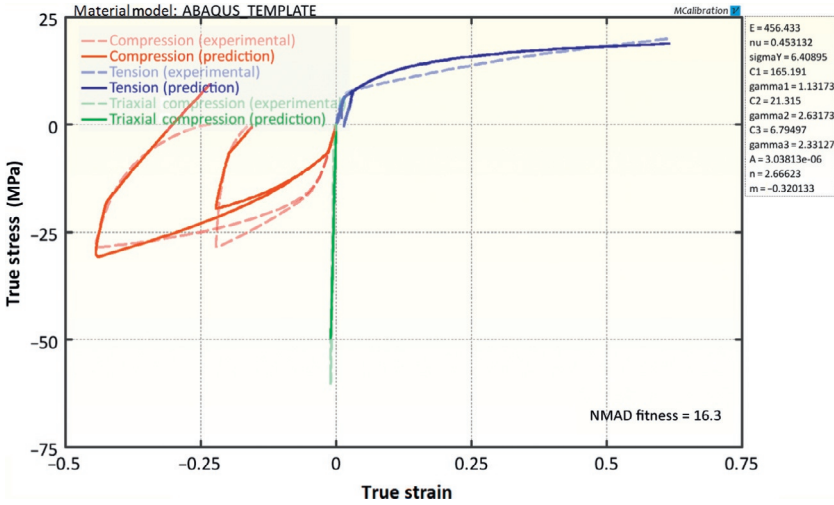
The most accurate material model for the PTFE data is the TN model. [Figure 11.43](#) shows that the TN model provides an overall accurate representation of all aspects of the stress-strain response of the PTFE in this study. The average error of the TN model predictions is 10.8%.



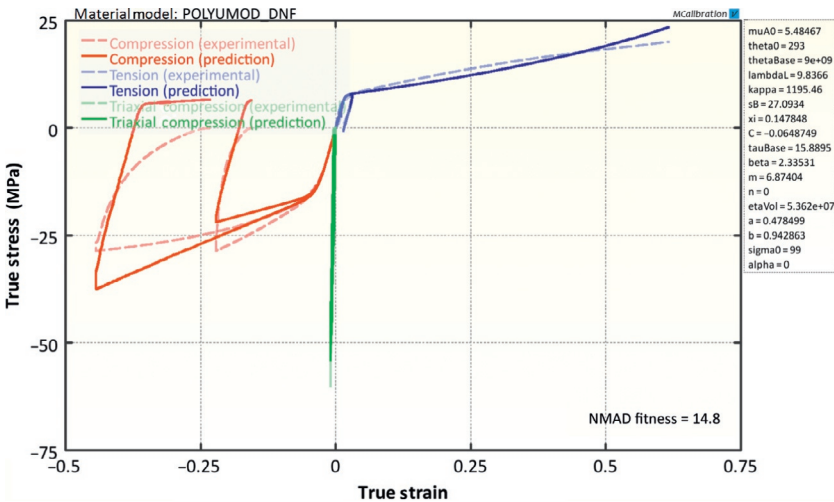
**Figure 11.39** Comparison between experimental data for PTFE and the best model predictions from an elastic-plastic material model with isotropic hardening.



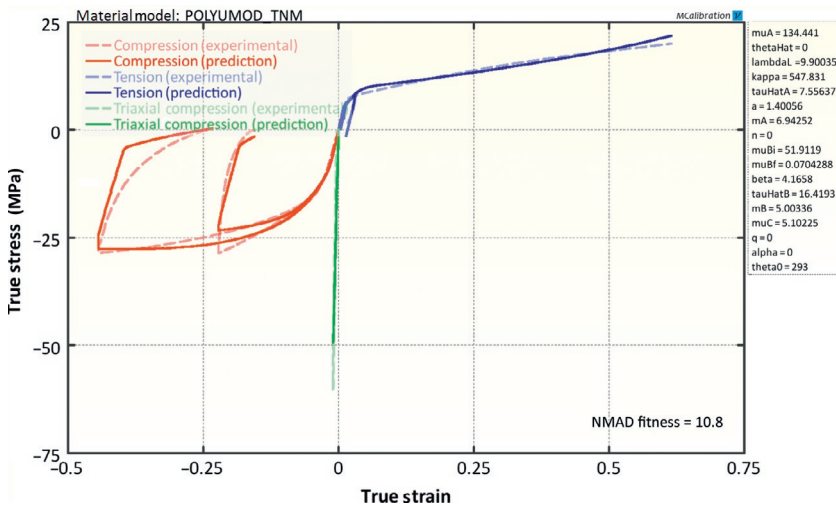
**Figure 11.40** Comparison between experimental data for PTFE and the best model predictions from a two-network PN model.



**Figure 11.41** Comparison between experimental data for PTFE and the best model predictions from the Abaqus elastic-plastic material model with combined kinematic hardening and creep-based rate-dependence. The model has three backstress networks.



**Figure 11.42** Comparison between experimental data for PTFE and the best model predictions from the DNF model.



**Figure 11.43** Comparison between experimental data for PTFE and the best model predictions from the TN model.

## 11.8 Polyethylene Terephthalate

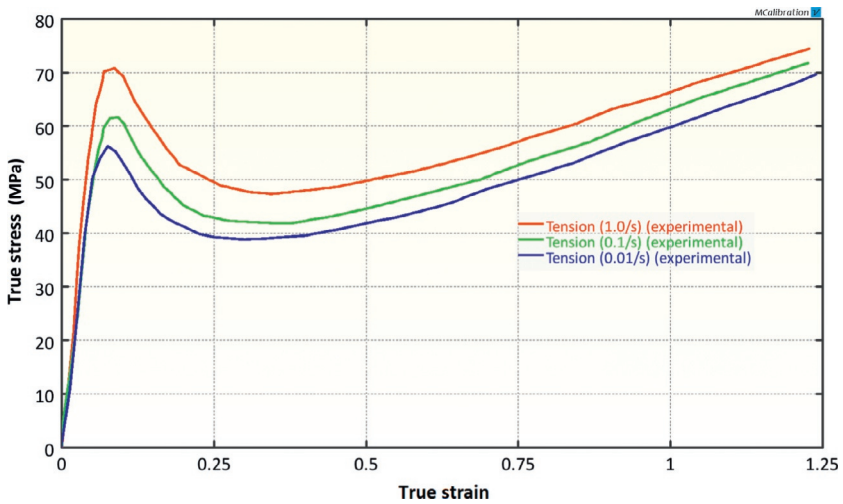
Polyethylene terephthalate (PET) is a thermoplastic polyester that is often used in synthetic fibers, bottles, and containers. PET can be either amorphous or semi-crystalline depending on the thermal history. Experimental tension data for a PET material is shown in [Figure 11.44](#) [2]. This figure shows that the stress-strain response is strain-rate dependent and that the true stress softens significant after the initial peak value. At large strains the stress starts to increase again due to molecular alignment.

The predictive capabilities of a number of different candidate material models are summarized in [Table 11.7](#).

The best calibration of the BB model to the experimental data is shown in [Figure 11.45](#). As expected, the BB model is unable to represent the stress reduction after the initial peak stress. The overall average error in the calibration results is 5.9%.

A significantly more accurate prediction is provided by an elastic-plastic material model with isotropic hardening and rate-dependent yield stress (see [Figure 11.46](#)). This model does a good job at reproducing the overall shape of the stress-strain response, but as was discussed in [Section 7.2](#), it should not

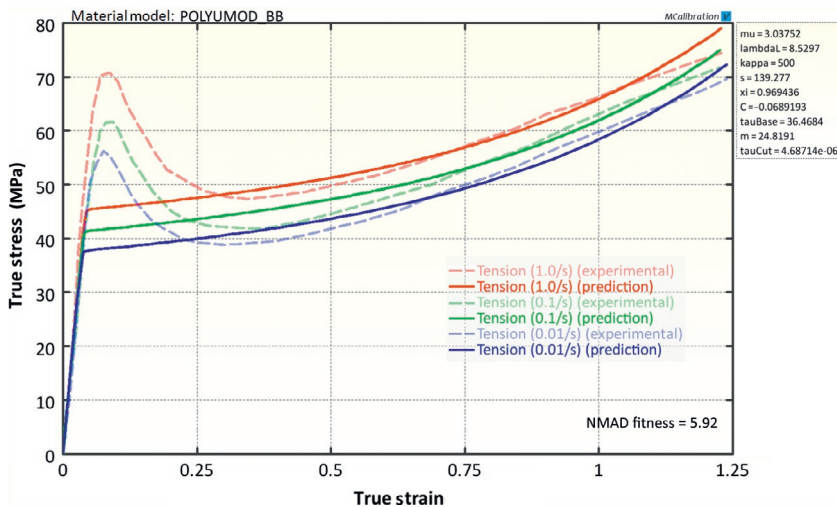




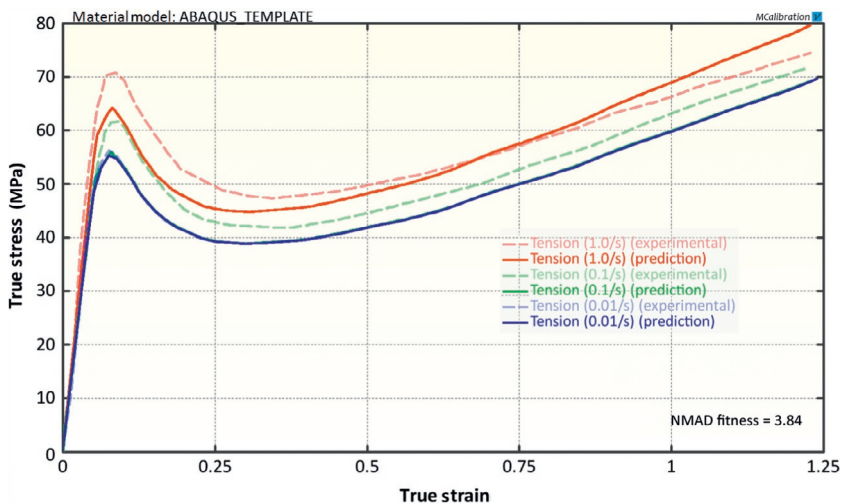
**Figure 11.44** Uniaxial tension data at three different strain rates for a PET material.

**Table 11.7** Summary of Results from Material Model Calibrations for HDPE

Material Model	Error in Model Calibration (%)
Arruda-Boyce eight-chain	41.2
Yeoh	25.9
Linear viscoelasticity (Yeoh)	26.3
BB	19.7
Abaqus elastic-plastic with isotropic hardening and creep	7.1
Abaqus elastic-plastic with isotropic hardening and rate dependence	3.8
Parallel Network model with two networks (yield evolution)	2.4
Three Network model	2.4



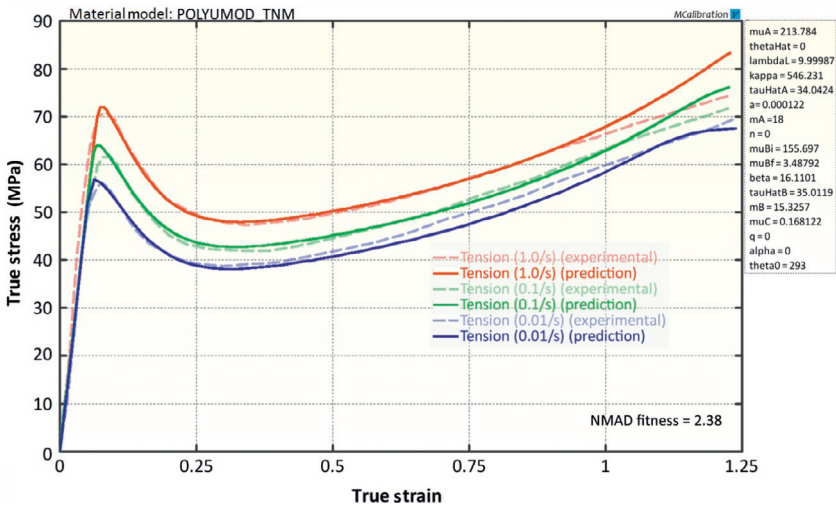
**Figure 11.45** Comparison between experimental data for PET and the best model predictions from the BB material model.



**Figure 11.46** Comparison between experimental data for PET and the best model predictions from Abaqus elastic-plastic model with isotropic hardening and rate-dependence.

be used in applications where the applied load also includes unloading.

The most accurate material model for this material was found to be the TN model. Figure 11.47 shows that the TN model represents the experimental data with great accuracy.



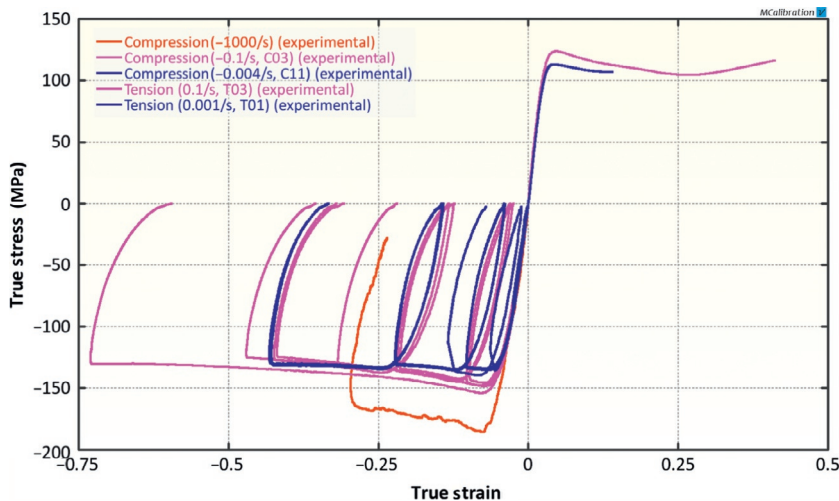
**Figure 11.47** Comparison between experimental data for PET and the best model predictions from the TN model.

## 11.9 Polyether Ether Ketone

Polyether ether ketone (PEEK) is a high strength and stiffness semicrystalline thermoplastic. Due to its excellent mechanical properties it is commonly used in seals and bearings, and more recently also in medical implants (e.g., spinal implants, screws, woven textiles).

Experimental data for a PEEK material is shown in [Figure 11.48](#). This figure summarizes uniaxial tension data at two strain rates (0.1 and 0.001/s), and uniaxial compression data at strain rates of  $-1000$ ,  $-0.1$ , and  $-0.004$ /s. The response at  $-1000$ /s was obtained using a split Hopkinson pressure bar testing (see Section 2.2.8). The data illustrate that the material has slightly higher yield stress in compression than in tension, and that the stress-strain response after unloading is highly nonlinear.

The predictive capabilities of a number of different candidate material models are summarized in [Table 11.8](#). The large error in the elastic-plastic model with isotropic hardening and rate-dependent plastic flow is a results of the calibration procedure. First, the plasticity parameters were determined from the large strain tension results, then the rate-dependence parameters were found in order to best match the complete data set.



**Figure 11.48** Uniaxial tension and compression data for an unfilled PEEK material.

The best calibration of the BB model to the experimental data for PEEK is shown in [Figure 11.49](#). As expected, the BB model captures the overall response of the material reasonably well, but is unable to accurately represent the nonlinear stress response after unloading. The average error in the calibration results is 24.3%.

The Johnson-Cook (JC) plasticity model (see [Section 7.4](#)) is similar to the BB model in its predictive capabilities of the PEEK data. Also in this case the main limitation of the predictions is the linear stress response during unloading. The average error in the calibration results is 24.9% ([Figure 11.50](#)).

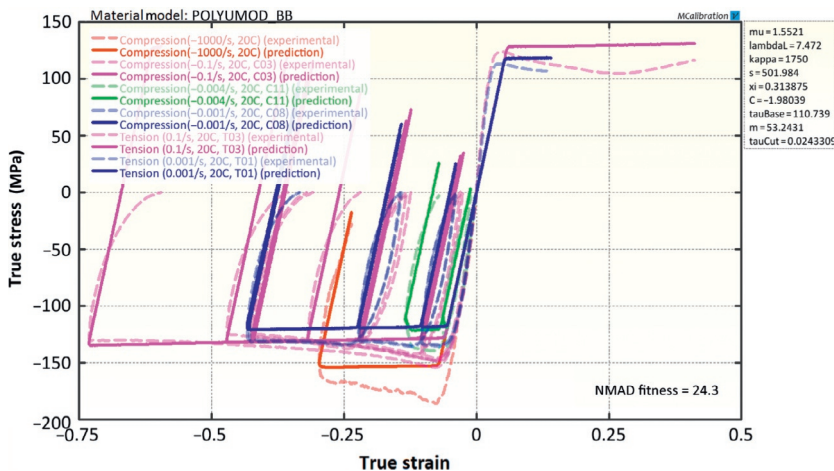
The PN model has the advantage that it allows for suitable components to be selected. In this case, an appropriate starting point is a two-network representation where network 1 has a Neo-Hookean element, and where network 2 has a Neo-Hookean element to represent the elastic response and a Power-law flow element (Equation (8.68)) with yield evolution (Equation (8.69)) and pressure dependence of the yield stress (similar to the  $a$  parameter in Equation (8.64)). The calibration results from this model is shown in [Figure 11.51](#). The predictions are in reasonable agreement with the experimental data, and the average error in the

**Table 11.8 Summary of Results from Material Model Calibrations for PEEK**

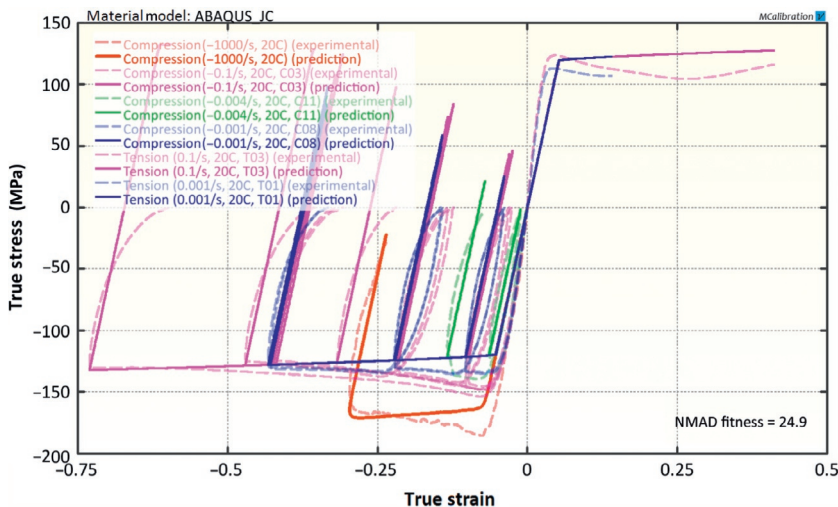
<b>Material Model</b>	<b>Error in Model Calibration (%)</b>
Elastic-plastic with isotropic hardening and rate-dependence	127
Arruda-Boyce eight-chain	68.4
Yeoh	59.6
DNF	30.9
Elastic-plastic with kinematic hardening and rate-dependent creep (three backstress)	30.6
Johnson-Cook	24.8
BB	24.4
Parallel Network model with two networks (Power flow, yield evolution, pressure dependence)	16.5
Parallel Network model with three networks (Power flow, yield evolution, pressure dependence)	9.1
Three Network model	9.1

calibration results is 16.5%. The accuracy of this model framework can be improved further by adding one more viscoelastic network of similar type.

The TN model accurately predicts the response of the PEEK at all tested conditions, see [Figure 11.52](#). The stress-strain response may look somewhat multilinear due to the small number of mechanisms used to represent the material response. The average error of the model predictions is 9.1%.

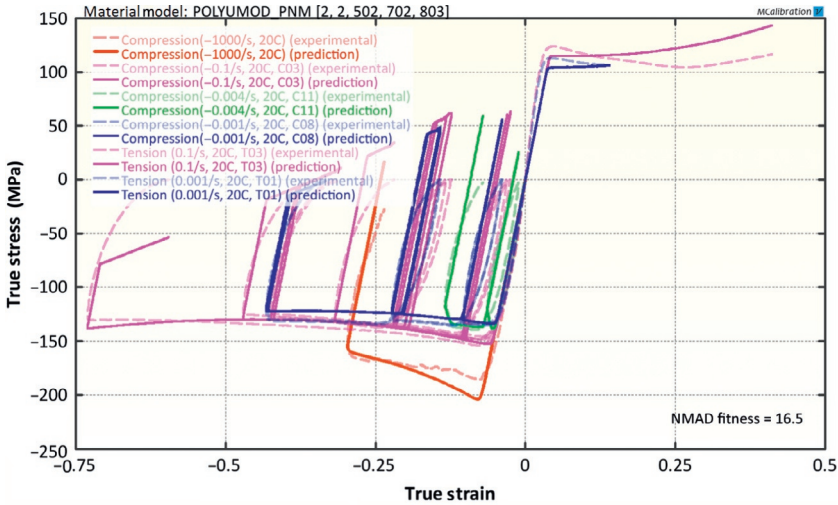


**Figure 11.49** Comparison between experimental data for PEEK and the best model predictions from the BB material model.

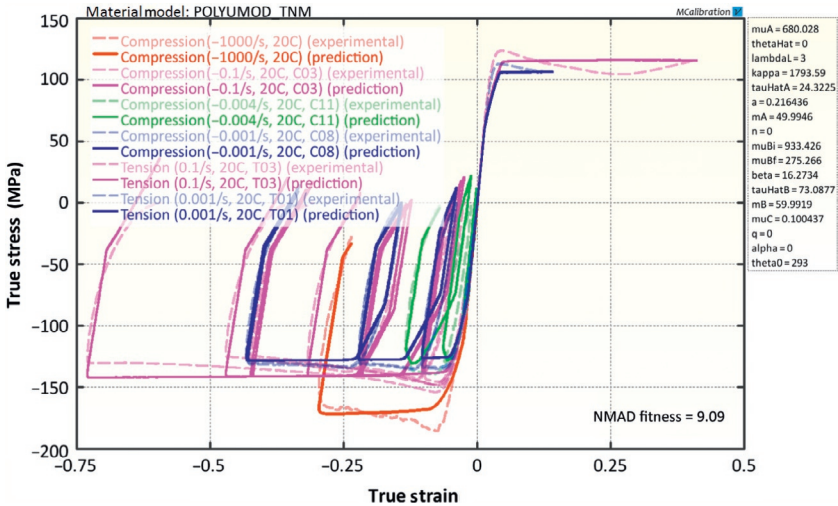


**Figure 11.50** Comparison between experimental data for PEEK and the best model predictions from the JC plasticity model.

The calibrated material models for PEEK were validated by comparing small punch data (see Section 2.3.2) and indentation testing with a spherical indenter (see photo in Figure 11.54) to finite element predictions of the experimental tests.

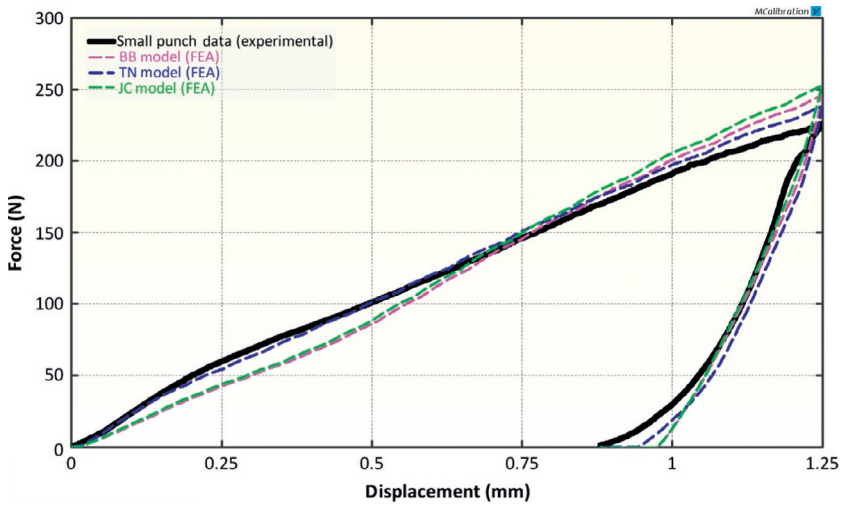


**Figure 11.51** Comparison between experimental data for PEEK and the best model predictions from the PN model with two networks.

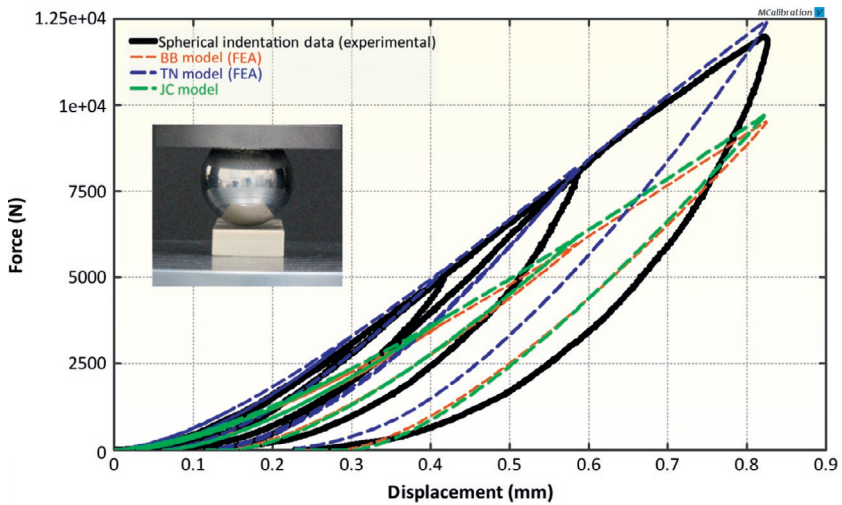


**Figure 11.52** Comparison between experimental data for PEEK and the best model predictions from the TN model.

Figure 11.53 shows the force-displacement results from a small punch experiment and the corresponding results from the BB model, the TN model, and the JC model. The figure shows that the TN model is more accurate at predicting the force response of this multiaxial test than the BB and JC models. This results also



**Figure 11.53** Comparison between experimental small punch data for PEEK and FE predictions from a few different material models.



**Figure 11.54** Comparison between experimental small punch data for PEEK and FE predictions from a few different material models.

demonstrates that these models can be used to accurately predict the response of a material under multiaxial loading even if they have only been calibrated using uniaxial data.

Results from the spherical indentation test and the model predictions from the BB model the TN model, and the JC model are shown in [Figure 11.54](#). The results shown in this figure are



similar to the results from the previous figure in that the TN model is more accurate than the BB and the JC models. Again, these models can accurately predict the response of a material under multiaxial loading even if they have only been calibrated using uniaxial data.

## 11.10 Exercises

1. The case studies presented in this section cover a wide variety of different materials. Are there any polymeric material classes that are not included?
2. Explain why a hyperelastic material model cannot accurately represent the behavior of the ABR.
3. Explain why adding Mullins damage to the material models of the ABR increases the accuracy of the predictions?
4. What other aspects than accuracy can be important to consider when selecting a material model?
5. Why does not LVE provide an accurate representation of the experimental data for the rubber materials studied in this chapter?
6. Which hyperelastic material model would you select for the CR material? Why?
7. Explain how the stress can increase in magnitude during a stress relaxation experiment.
8. What material model would you select for the Santoprene material? Why?
9. Why is the compressive stress so much higher than the tensile stress (at the same strain level) for the PTFE material? What material model can capture the stress difference?
10. Which material models can capture a large drop in stress after yielding?
11. Why does not a plasticity model work well for PEEK?

## References

- [1] J.S. Bergström, L.B. Hilbert Jr., A constitutive model for the large deformation thermomechanical behavior of fluoropolymers, *Mech. Mater.*, 37 (2005) 899-913.
- [2] R.B. Dupaix, Temperature and rate dependent finite strain behavior of poly(ethylene terephthalate) and poly(ethylene terephthalate)-glycol above the glass transition temperature, Ph.D. thesis, MIT, 2003.

*Note:* Page numbers followed by *f* and *t* refers to figures and tables respectively.

## A

Abaqus/Explicit VUMAT

subroutine, for Neo-Hookean model, 448–450

Abaqus/Implicit UMAT

subroutine, for Neo-Hookean model, 450–454

AB model. *See* Arruda–Boyce (AB) model

Acrylate–Butadiene rubber (ABR)

Bergstrom–Boyce model, 458, 459–460, 459*f*

calibrations, 456, 458*t*

linear viscoelasticity model, 458, 459*f*

mechanical response, 455–456

stress–time response, 456, 457*f*

uniaxial compression data, 456, 456*f*

Yeoh hyperelastic model, 456, 457*f*

Addition polymerization, 11

Adiabatic thermoelastic material, 194

Almansi strain, 163

Amorphous polymers, 5–7, 6*f*

Anisotropic elasticity, 215–217

Anisotropic hyperelasticity

Bergstrom anisotropic eight-chain model, 285

Bischoff anisotropic eight-chain model, 283–285

generalized Fung model, 282

Holzappel–Gasser–Ogden model, 285–287

invariant based anisotropy, 282–283

Anisotropic material, 199

Arrhenius model, 344–345

Arruda–Boyce (AB) model, 283

athermal shear resistance, 396

deviatoric back stress,

394–395

glassy polymers, 393–394

linear elastic response, 394

plastic flow rate, 395–396

rheological representation, 394, 394*f*

stress–strain predictions, 396,

397*f*

Atomic force microscopy (AFM), 87–88

## B

Balance law, 171–184

Balance of angular momentum, 178–180

Balance of linear momentum, 175–178

BAM model, 275–277

Barcol hardness testing, 50

BB model. *See* Bergstrom–Boyce (BB) model

Bergstrom anisotropic eight-chain model, 285

- Bergstrom–Boyce (BB) model,  
 27–28  
 acrylate-butadiene rubber, 458,  
 459–460, 459*f*  
 applied strain history, 372–374,  
 373*f*  
 Brownian motion, 379, 380*f*  
 Cauchy stress, 376–377  
 chain stretch, 379–381  
 chloroprene rubber, 372–374,  
 373*f*, 462, 463*f*  
 creep experiment, 379–381  
 crosslinked polymer,  
 378, 379*f*  
 dynamic loading predictions,  
 387–392  
 eight-chain model, 376–377  
 elastic and viscoelastic  
 components, 376, 376*f*  
 elastomers, 375  
 equilibrium stress, 374–375,  
 374*f*  
 generic numerical  
 implementation, 386–  
 387  
 hyperelastic response, 377  
 hypothetical stress-strain curve,  
 374–375  
 Matlab implementation,  
 382–384  
 nitrile rubber, 466, 467*f*  
 non-linear viscoelastic flow  
 element, 375  
 polyether ether ketone, 491,  
 493*f*  
 polymer modeling, 392–393  
 Python implementation,  
 384–386  
 Rouse relaxation time, 379–381  
 santoprene, 470–474, 473*f*,  
 474*f*  
 strain amplitude dependence,  
 372  
 time derivative, 378  
 viscous components, 377–378  
 viscous flow, 381–382
- Biot strain, 163
- Birefringence spectroscopy,  
 95–97
- Bischoff anisotropic eight-chain  
 model, 283–285
- Blatz–Ko foam model, 289
- Boltzmann’s superposition  
 principle, 310
- Bulk modulus, 64–71, 241
- Buna-N. *See* Nitrile rubber
- C**
- Cauchy stress theorem,  
 176–178
- Cauchy surface tractions, 176
- Chemical characterization  
 techniques  
 energy dispersive X-ray  
 spectroscopy, 101–103  
 Fourier transform infrared  
 spectroscopy, 100–101  
 Raman spectroscopy, 109–110  
 size-exclusion chromatography,  
 103–107  
 thermogravimetric analysis,  
 107–109
- Chloroprene rubber (CR)  
 BB model with Mullins  
 damage, 462, 463*f*  
 calibrations, 461–462, 461*t*  
 linear viscoelastic model, 462,  
 463*f*  
 stress relaxation response, 460,  
 461*f*  
 uniaxial compression data, 460,  
 460*f*

- uniaxial tension, 440, 441*f*
  - Yeoh hyperelastic model, 461–462, 462*f*
  - Coefficient of determination, 444
  - Condensation polymerization, 11
  - Conductive polymers, 9
  - Confocal microscopy, 83
  - Conservation of mass, 173–175
  - Continuum mechanics
    - foundations, 219–224
    - balance laws and field equations, 171–184
    - constitutive equations, 187–194
    - coordinate transformations, 149
    - deformation gradient, 150–157
    - derivatives of scalar, vector, and tensor fields, 147–149
    - Dyadic product, 143–144
    - energy balance and stress power, 184–186
    - invariants, 150
    - large strain kinematics, 137–141
    - material symmetry, 198–199
    - multiaxial loading, 135–137
    - observer transformation, 194–198
    - rates of deformation, 164–165
    - strain, stretch, and rotation, 157–164
    - stress tensors, 165–170
    - symbols, 199
    - tensor operations, 144–147
    - uniaxial loading, 133–135
    - vector operations, 141–143
  - Coordinate transformations, 149
  - Corrugated hose failure, 127
  - CR. *See* Chloroprene rubber (CR)
  - Creep compliance
    - definition, 335
    - vs.* relaxation modulus, 336–337
- D**
- Dark field microscopy, 83
  - Deformation
    - modeling, 120–125
    - simple shear, 152
    - undeformed state, 151
    - uniaxial tension, 151
    - volumetric deformation, 153
  - Dependence of stored energy, 229–232
  - Differential interference contrast (DIC) microscopy, 83
  - Differential scanning calorimetry (DSC), 89–90
  - Digital image correlation (DIC)
    - strain measurement system, 66
  - DNF model. *See* Dual network fluoropolymer (DNF) model
  - Drucker Prager plasticity, 366–367, 367*f*
  - Drucker stability, 297
  - Dual network fluoropolymer (DNF) model, 121–122
    - Cauchy stress, 398–400
    - constant viscosity, 402
    - deviatoric viscoelastic flow, 401–402
    - kinematics of deformation, 398, 399*f*
    - material parameters, 403
    - Matlab implementation, 404
    - plastic flow, 402–403
    - polymer modeling, 404
    - strain rates, 397–398
    - structure, 398, 399*f*

- Dual network fluoropolymer  
 (DNF) model (*Continued*)  
 thermal expansion, 398–400  
 thermoplastics, 398  
 velocity gradient, 401  
 viscoelastic deformation  
 gradient, 400–401  
 volumetric viscoelastic flow,  
 401–402
- Dyadic product, 143–144
- Dynamic mechanical analysis  
 (DMA), 43–47, 347
- E**
- Eigenvalue and spectral  
 decompositions, 154–  
 157
- Eight-chain (EC) model,  
 250–259
- Elastomers, 24, 25*f*
- Energy balance and stress power,  
 184–186
- Energy dispersive X-ray  
 spectroscopy (EDS),  
 101–103
- Entropy, 183
- Environmental SEM (ESEM), 86
- Environmental stress cracking  
 (ESC), 126
- Euler–Almansi strain, 164
- Extended tube (ET) model,  
 273–275
- F**
- Failure model calibration, 73
- Failure modeling, 125–130
- FEA. *See* Finite element analysis  
 (FEA)
- Fiber-reinforced composite, 217,  
 218*f*
- Finite element analysis (FEA)  
 deformation modeling, 120–  
 125  
 failure modeling, 125–130  
 polymer mechanics, 115  
 properties of polymers and  
 metals, 116–117  
 required inputs, 117–118  
 types, 119
- First law of thermodynamics,  
 180–182
- First Piola–Kirchhoff stress  
 tensor, 167
- Flex circuit pressure sensor, 124
- Fluorescence microscopy, 84
- Fourier transform approach, 326
- Fourier transform infrared  
 spectroscopy (FTIR),  
 100–101
- Freely jointed chain (FJC) model,  
 232–236
- G**
- Gaussian chains, 258
- Gel permeation chromatography  
 (GPC), 103–107
- Generalized Fung model, 282
- Genetic algorithm, 445
- Gent model, 263–265
- Glass transition temperature, PET,  
 24
- Green–Lagrange strain, 163
- H**
- Hardness and indentation testing,  
 47–51
- HDPE. *See* High-density  
 polyethylene (HDPE)
- Heaviside step function, 310
- Helmholtz free energy,  
 191–192
- Hencky strain, 163, 164

- High-density polyethylene (HDPE)  
 Arruda–Boyce eight-chain model, 477–478, 478*f*  
 calibrations, 477, 477*t*  
 elastic-plastic material model, 477–478, 478*f*  
 linear viscoelastic model, 479, 479*f*  
 PN model, 479, 480*f*  
 power-flow model, 479, 481*f*  
 stress relaxation data, 474–476, 476*f*  
 stress-strain response, 474–476  
 uniaxial tension data, 476*f*
- Holzapfel–Gasser–Ogden (HGO) model, 285–287
- Hooke’s law, 67–68, 211–212
- Horgan and Saccomandi model, 265–266
- Hybrid model (HM)  
 backstress network, 411  
 deformation map, 409, 410*f*  
 energy activation approach, 412  
 isotropic linear elasticity expression, 410–411  
 Matlab implementation, 413–414  
 polymer modeling, 414–416  
 relative stiffness, 411  
 rheological representation, 409, 410*f*  
 strain elastic constants, 412  
 ultra-high molecular weight polyethylene, 409  
 viscoelastic deformation gradient, 412  
 viscoplastic flow, 411–412
- Hyperelastic foam models  
 Blatz–Ko foam model, 289  
 hyperfoam model, 290–291
- Hyperelasticity  
 code examples, 299–303  
 Drucker stability, 297  
 experimental testing, 296–297  
 limitations, 298–299  
 material parameters, 298
- Hyperfoam model, 290–291
- I**
- $I_1$  and  $I_2$  model, 250
- Impact testing, 40–43
- Incompressible biaxial deformation, 237–238
- Incompressible planar deformation, 237–238
- Incompressible uniaxial deformation, 237–238
- Interface friction, 27–28
- Invariant based anisotropy, 282–283
- Inverse Langevin function, 256–257
- Isothermal thermoelastic material, 194
- Isotropic elasticity, 211–215
- Isotropic hardening plasticity model. *See*  $J_2$ -plasticity, isotropic hardening
- Isotropic hyperelasticity  
 BAM model, 275–277  
 continuum mechanics foundations, 219–224  
 dependence of stored energy, 229–232  
 eight-chain model, 250–259  
 extended tube model, 273–275  
 freely jointed chain model, 232–236  
 Gent model, 263–265

## Isotropic hyperelasticity

*(Continued)*

- Horgan and Saccomandi model, 265–266
  - $I_1$  and  $I_2$  model, 250
  - Knowles hyperelastic model, 268–270
  - Mooney–Rivlin model, 243–245
  - Neo–Hookean model, 236–242
  - Ogden model, 259–261
  - predictive capabilities, 277–281
  - pure shear *vs.* planar tension, 226–228
  - response function
    - hyperelasticity, 270–272
  - uniaxial compression *vs.* biaxial tension, 225–226
  - Yeoh model, 245–248
- Isotropic material, 199

**J**

- Johnson–Cook plasticity model, 365–366, 366*f*
- $J_2$ -plasticity, isotropic hardening
  - abacus, 354
  - ANSYS, 354
  - cyclic loading, 355–357, 356*f*
  - Matlab implementation, 357–359
  - Python implementation, 359–360, 359*f*, 360*f*
  - stress-strain representation, 355, 355*f*
- UHMWPE thermoplastic
  - material, 361–362, 361*f*, 362*f*

**K**

- Kinematic hardening plasticity
  - model
    - Abaqus material definition, 363, 364, 365
    - backstress network, 363, 363*f*, 364*f*, 365
    - Chaboche type, 362–363
    - limitations, 365
    - MCalibration software, 363
- Knowles hyperelastic model, 268–270

**L**

- Lagrangian and Eulerian
  - Formulations, 139
- Large strain kinematics, 137–141
- Large strain linear viscoelasticity
  - generalization, 331–332
  - hyperelastic stress function, 332
  - numerical implementation, 332–334
- Linear elasticity
  - anisotropic elasticity, 215–217
  - isotropic elasticity, 211–215
  - transversely isotropic elasticity, 217–218
- Linear viscoelasticity
  - creep compliance, 335–337
  - differential form, 337–340
  - large strain, 331–334
  - polymer modeling, 345–349
  - shift functions, 340–345
  - small strain, 310–331
- Loss modulus, 322–323

**M**

- Material parameters, 437
- determination, 438–440



- error measurement functions, 442–444
  - extraction, 439, 439*f*
  - find\_material\_params, 444–445
  - initial guess, 440–442, 441*f*
  - mathematical minimization problem, 439–440
  - Monte Carlo method, 442
  - optimization algorithm, 444–445
  - prior knowledge, 442
  - Matlab implementation
    - Bergstrom–Boyce model, 382–384
    - dual network fluoropolymer model, 404
    - hybrid model, 413–414
    - J<sub>2</sub>-plasticity, isotropic hardening, 357–359
    - small strain linear viscoelasticity, 329, 330*f*
    - three network model, 422
  - Maxwell rheological model, 338–339, 339*f*
  - MCalibration software, 445
  - Mechanical stress, 134
  - Metal plasticity model, 353
  - Mises stress, 123, 123*f*, 170
  - Monte Carlo method, 442
  - Mooney–Rivlin (MR) model, 243–245
  - Mullins effect models
    - Ogden–Roxburgh, 293–295
    - Qi–Boyce, 295
  - Multiaxial loading, 135–137
  - Multi-network Maxwell model, 340*f*
- N**
- Nanoindentation, 51
  - Nanson’s formula, 156
  - Natural polymers, 4–5, 5*f*
  - NBR. *See* Nitrile rubber
  - Near-field scanning optical microscopy (NSOM), 83
  - Nelder–Mead simplex method, 444–445
  - Neo–Hookean hyperelastic material model
    - Abaqus/Explicit VUMAT, 448–450
    - Abaqus/Implicit UMAT, 450–454
    - stress, 447–448
  - Neo–Hookean (NH) model, 236–242
  - Neoprene. *See* Chloroprene rubber (CR)
  - Nitrile rubber
    - BB model, 466, 467*f*
    - calibrations, 464, 465*t*
    - linear viscoelastic model, 466, 467*f*
    - stress-time response, 464, 465*f*
    - uniaxial compression data, 464, 464*f*
    - Yeoh hyperelastic model, 465–466, 466*f*
  - Nominal strain, 164
  - Nominal traction vector, 167
  - Normalized mean absolute difference, 444
  - Normalized root-mean square difference, 444
- O**
- Ogden model, 259–261
  - Ogden–Roxburgh Mullins effect model, 293–295

Optical microscopy, 81–84  
 Orthotropic elasticity,  
 216–217

## P

Parallel network (PN) model,  
 427–431, 459–460  
 high-density polyethylene, 479,  
 480*f*  
 polyether ether ketone,  
 491–492, 494*f*  
 Payne effect, 348–349  
 PEEK. *See* Polyether ether ketone  
 (PEEK)  
 Plane strain tension, 33–37  
 Plasticity theory. *See*  $J_2$ -plasticity,  
 isotropic hardening  
 Polarized light microscopy, 82  
 Polyether ether ketone (PEEK)  
 BB model, 491, 493*f*  
 calibrations, 490, 492*t*  
 force-displacement results,  
 494–495, 495*f*  
 Johnson–Cook plasticity  
 model, 491, 493*f*  
 PN model, 491–492, 494*f*  
 spherical indentation test,  
 495–496  
 TN model, 492, 494*f*  
 uniaxial tension and  
 compression data,  
 490, 491*f*  
 Polyethylene terephthalate (PET),  
 487–489  
 Polylactic acid (PLA), 7  
 Polymers  
 description, 1, 2–3  
 history, 7–10  
 manufacturing and processing,  
 11  
 mechanics, 11–15

plasticity models, 367–368  
 types, 4–7  
 Polypropylene (PP), 10  
 Polytetrafluoroethylene (PTFE)  
 calibrations, 484*t*  
 dual network fluoropolymer  
 model, 483–484, 486*f*  
 elastic-plastic material model,  
 482–483, 485*f*  
 mechanical behavior, 479–481  
 microporosity, 479–481  
 TN model, 484, 487*f*  
 volumetric compression data,  
 483*f*  
 yield stress, 479–481  
 PolyUMod library, 447–448  
 Powell method, 445  
 Pressure-volume-temperature  
 (PVT) testing, 66  
 Prony series, 315–316, 317*f*,  
 336–337, 345–346  
 PTFE. *See* Polytetrafluoroethy-  
 lene (PTFE)  
 Pure shear vs. planar tension,  
 226–228  
 Python implementation  
 Bergstrom–Boyce model,  
 384–386  
 $J_2$ -plasticity, isotropic  
 hardening, 359–360,  
 359*f*, 360*f*  
 small strain linear  
 viscoelasticity, 330–  
 331, 331*f*  
 three network model, 422  
 viscoplasticity models, 432–  
 434

## Q

Qi–Boyce Mullins effect model,  
 295

**R**

Raman spectroscopy, 109–110

Rates of deformation, 164–165

Relaxation time spectrum, 328

Residual error

strain-controlled experiment,  
442, 443*f*

stress-controlled experiment,  
443, 443*f*

Response function hyperelasticity,  
270–272

Retardation time spectrum, 328

Rheologically simple material,  
342

Rheological models, 338–340,  
339*f*, 340*f*

Rockwell hardness testing,  
47–48

**S**

Santoprene

BB model, 470–474, 473*f*,  
474*f*, 475*f*

calibrations, 468, 470*t*

elastic-plastic material model,  
469–470, 472*f*, 473*f*

isotropic hardening plasticity  
model with rate-  
dependence, 469,  
472*f*

linear viscoelastic model, 469,  
471*f*

uniaxial tensile stress-strain  
data, 468, 468*f*, 469*f*

Yeoh hyperelastic model,  
468–469, 471*f*

Scanning electron microscopy  
(SEM), 84–86

Second law of thermodynamics,  
183–184

Semicrystalline polymers, 5–7, 6*f*

Shear and bulk relaxation moduli,  
312–313

Shear modulus, 239

Shore (durometer) testing, 48–49

Simple anisotropic hyperelastic  
model, 283

Simple shear, 37–39, 152

Size-exclusion chromatography  
(SEC), 103–107

Small-angle X-ray diffraction, 95

Small punch testing, 77–79

Small-strain classical theory, 135

Small strain linear viscoelasticity  
applied strain history,  
311, 312*f*

Boltzmann's superposition  
principle, 310

characteristic relaxation time,  
313

cyclic loading response,  
320–322

Heaviside step function, 310

Matlab implementation, 329,  
330*f*

mat\_LVE() function, 329

monotonic loading response,  
314–320, 317*f*

Prony series, 315–316, 317*f*

Python implementation,  
330–331, 331*f*

relaxation time spectrum, 328

retardation time spectrum, 328

shear and bulk relaxation  
moduli, 312–313

storage and loss modulus,  
322–327

stress relaxation, 310, 311*f*,  
313, 314*f*

stretched exponential stress  
relaxation modulus,  
316–318, 318*f*, 319*f*

- Small strain linear viscoelasticity  
(*Continued*)  
test\_mat\_LVE function, 329,  
330*f*
- Spatial velocity gradient,  
164–165
- Spin tensor, 164–165
- Split-Hopkinson pressure bar  
(SHPB) testing, 53–63
- Stereo microscopy, 84
- Storage modulus, 322–323
- Strain matrix, 136
- Stress invariants, 169–170
- Stress-strain response, 24
- Stress tensors, 165–170
- Surface characterization  
techniques  
atomic force microscopy,  
87–88  
optical microscopy, 81–84  
scanning electron microscopy,  
84–86
- Swell testing, 97–99
- Synthetic polymers, 4–5, 5*f*
- T**
- Tensor operations, 144–147
- Thermoelastic material,  
189–194
- Thermogravimetric analysis  
(TGA), 107–109
- Thermomechanical deformations,  
121
- Thermoplastics, 5, 6*f*, 24, 26*f*
- Thermoplastic vulcanizates  
(TPV). *See* Santoprene
- Thermosets, 5, 6*f*, 24, 27*f*
- Threaded connection gasket, 121
- Three network model (TNM),  
459–460
- arbitrary rigid body rotation,  
420–421
- Cauchy–Green deformation  
tensor, 417–418
- deformation gradient, 417–418
- elastic and viscous components,  
419–420
- flow rate, 419–420
- material parameters, 421, 421*t*
- Matlab implementation, 422
- plastic strain, 418–419
- polyether ether ketone, 492,  
494*f*
- polymer modeling, 426
- polytetrafluoroethylene, 484,  
487*f*
- Python implementation, 422
- rheological representation, 417,  
417*f*
- shear modulus, 419
- viscoelastic deformation  
gradient, 418–419
- Time shifts, 342
- Time-temperature equivalence,  
341–345, 341*f*, 343*f*, 344*f*
- TNM. *See* Three network model  
(TNM)
- Transmission electron microscopy  
(TEM), 90–91
- Transversely isotropic elasticity,  
217–218
- Tresca stress, 170
- U**
- Ultra-high molecular  
weight polyethylene  
(UHMWPE), 213–214,  
214*f*
- isotropic hardening plasticity  
model, 361–362

- Johnson–Cook model, 365–366, 366*f*
  - kinematic hardening plasticity model, 362–365
  - linear viscoelasticity
    - application, 346, 346*f*
  - Uniaxial compression
    - vs. biaxial tension, 225–226
    - testing, 24–29
  - Uniaxial loading, 133–135
  - Uniaxial tension, 29–33, 151
  - User material subroutines
    - Abaqus/Explicit VUMAT, 448–450
    - Abaqus/Implicit UMAT, 450–454
    - description, 447–448
    - purpose, 447–448
- V**
- Vector and tensor algebra, 141–150
  - Vertical shifts, 345
  - Viscoplastic deformations, 130
  - Viscoplasticity models
    - Arruda–Boyce model, 393–396
    - Bergstrom–Boyce model, 372–393
    - dual network fluoropolymer model, 397–404
    - hybrid model, 409–416
    - parallel network model, 427–431
    - polymer modeling, 431–432
    - Python code examples, 432–434
    - three network model, 417–426
  - V-notch shear testing, 80
  - Volume characterization techniques
    - birefringence, 95–97
    - differential scanning calorimetry, 89–90
    - swell testing, 97–99
    - transmission electron microscopy, 90–91
    - X-ray diffraction, 92–95
  - Volumetric deformation, 153
  - Vulcanized natural rubber, 8–9
- W**
- Water filter failure, 126
  - Wide-angle X-ray diffraction, 93–94
  - William–Landel–Ferry (WLF) equation, 343, 344, 344*f*
  - Work conjugate stress, 185
- X**
- X-ray diffraction (XRD), 92–95
- Y**
- Yeoh hyperelastic model, 456, 457*f*
    - acrylate-butadiene rubber, 456
    - chloroprene rubber, 461–462, 462*f*
    - nitrile rubber, 465–466, 466*f*
    - santoprene, 468–469, 471*f*
  - Yeoh model, 245–248
  - Young’s modulus, 23, 23*f*