ORANI-G

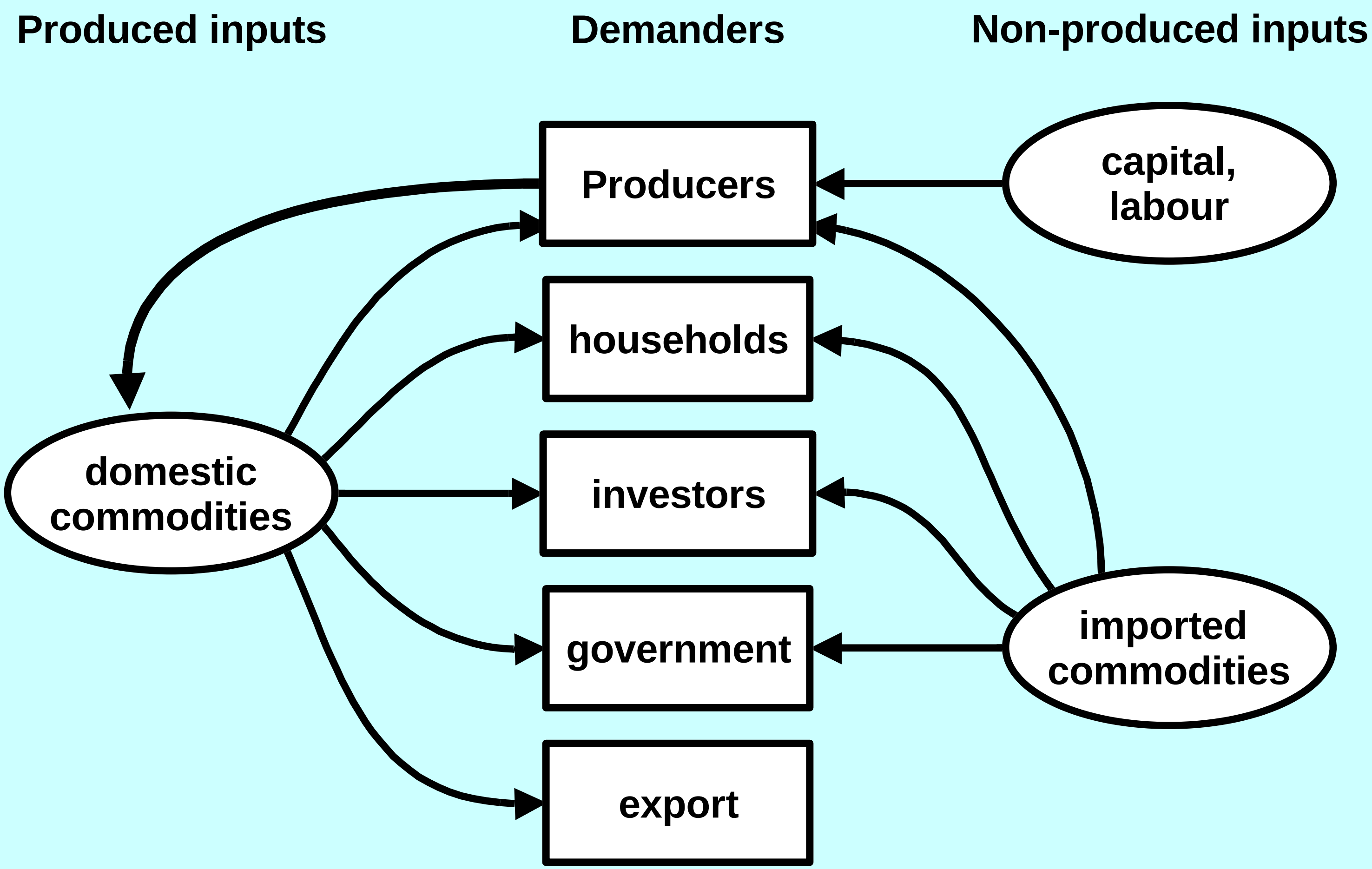
A Generic CGE Model

Document: ***ORANI-G: a Generic Single-Country Computable General Equilibrium Model***

Please tell me if you find any mistakes in the document !

Contents

Stylized GE model: material flows



Stylized GE model: demand equations

	Producer i	Absorption C I G	Export	Total Demand
Domestic good c	$Q_i F(P/PD_c)$	$E_u F(P,PD_c)$	$F(1/PD_c)$	Q_c = sum(left)
Imported good c	$Q_i F(P/PM_c)$	$E_u F(P,PM_c)$		
Primary factor f	$Q_i F(P/PF_f)$			QF_f = sum(left)
Production cost	total cost of above = $PD_i Q_i$			

Supply = demand

Quantity of good c used by sector i

Costs = Sales

Notation:

- PD_c = price dom good c
- PM_c = price imp good c
- PF_f = price factor f
- P = full price vector [PD,PM,PF]
- Q_i = output good i
- F = various functions
- QF_f = supply factor f
- E_u = expenditure final user u

Stylized CGE model: Number of equations = number endogenous variables

Variable	Determined by:
PD_c = price dom good c	ZERO PURE PROFITS values of sales = PD_cQ_c = sum(input costs) = F(all variables)
Q_c = output good c	MARKET CLEARING Q_c = sum(individual demands) = F(all variables)
PF_f = price factor f	For each f, one of PF or QF fixed, the other determined by:
QF_f = price factor f	QF_f = sum(individual demands) = F(all variables)
E_u = spending final user u	either fixed, or linked to factor incomes (with more equations)
PM_c = price imp good c	fixed

Red: exogenous (set by modeler)
Green: endogenous (explained by system)

What is an applied CGE model ?

- Computable, based on data
- It has many sectors
- And perhaps many regions, primary factors and households
- A big database of matrices
- Many, simultaneous, equations (hard to solve)
- Prices guide demands by agents
- Prices determined by supply and demand
- Trade focus: elastic foreign demand and supply

CGE simplifications

- Not much dynamics (leads and lags)
- An imposed structure of behaviour, based on theory
  - Neoclassical assumptions (optimizing, competition)
  - Nesting (separability assumptions)

Why: time series data for huge matrices cannot be found.

Theory and assumptions (partially) replace econometrics

What is a CGE model good for ?

Analysing policies that affect different sectors in different ways

The effect of a policy on different:

- Sectors
- Regions
- Factors (Labour, Land, Capital)
- Household types

Policies (tariff or subsidies) that help one sector a lot, and harm all the rest a little.

What-if questions

What if productivity in agriculture increased 1%?

What if foreign demand for exports increased 5%?

What if consumer tastes shifted towards imported food?

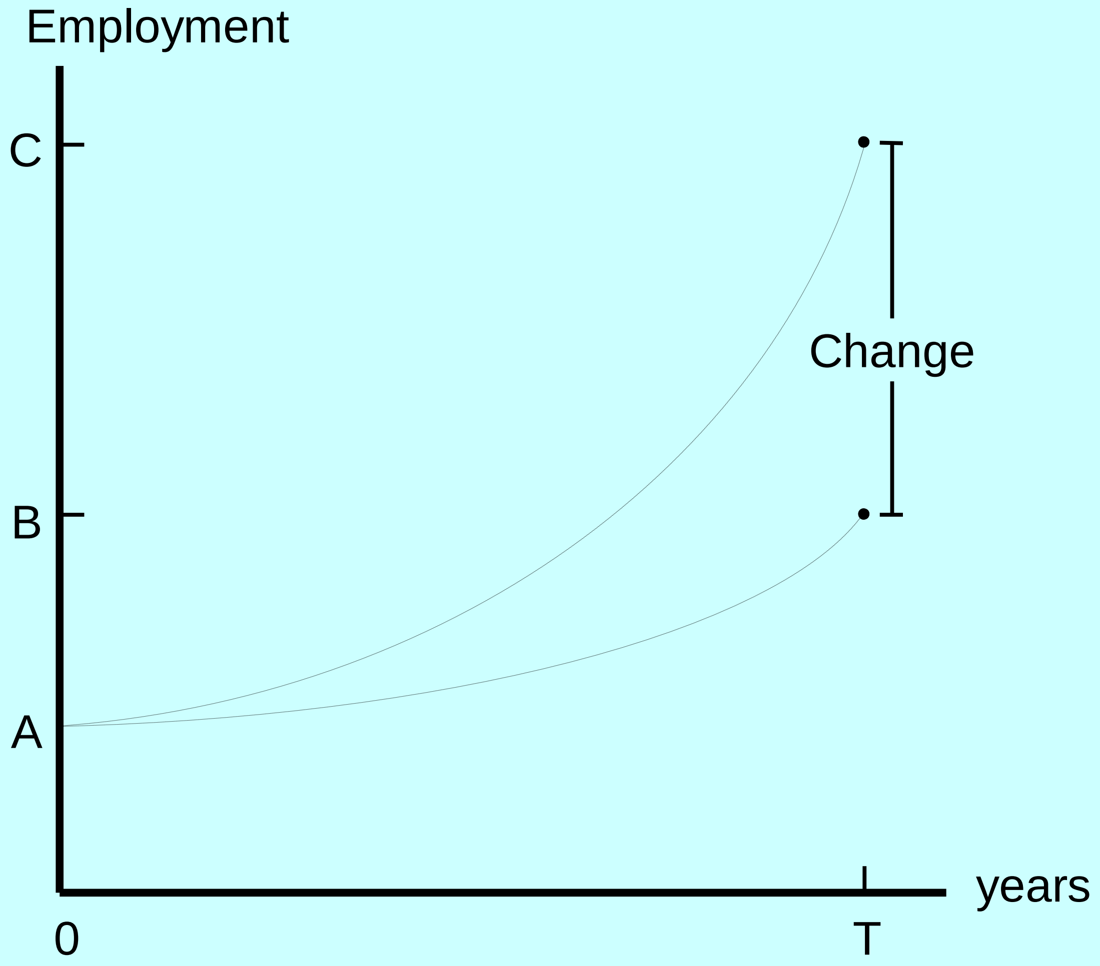
What if CO_2 emissions were taxed?

What if water became scarce?

A great number of exogenous variables (tax rates, endowments, technical coefficients).

Comparative static models: Results show effect of policy shocks only, in terms of changes from initial equilibrium

Comparative-static interpretation of results



Results refer to changes at some future point in time.

ORANI-G

- A model of the Australian economy, still used, but superseded at Monash (by MMRF and MONASH models).
- A teaching model.
- A template model, adapted for use in many other countries (INDORANI, TAIGEM, PRCGEM).
- Most versions do not use all features and add their own features.
- Still evolving: latest is ORANIG06.
- Various Australian databases:

  23 sector 1987 data is public and free (document),

  34 sector 1994 data used in this course (simulations).

  144 sector 1997 data used by CoPS.

ORANI-G like other GE models

Equations typical of an AGE model, including:

- market-clearing conditions for commodities and primary factors;
- producers' demands for produced inputs and primary factors;
- final demands (investment, household, export and government);
- the relationship of prices to supply costs and taxes;
- a few macroeconomic variables and price indices.

Neo-classical flavour

- Demand equations consistent with optimizing behaviour (cost minimisation, utility maximisation).
- competitive markets: producers price at marginal cost.

What makes ORANI special ?

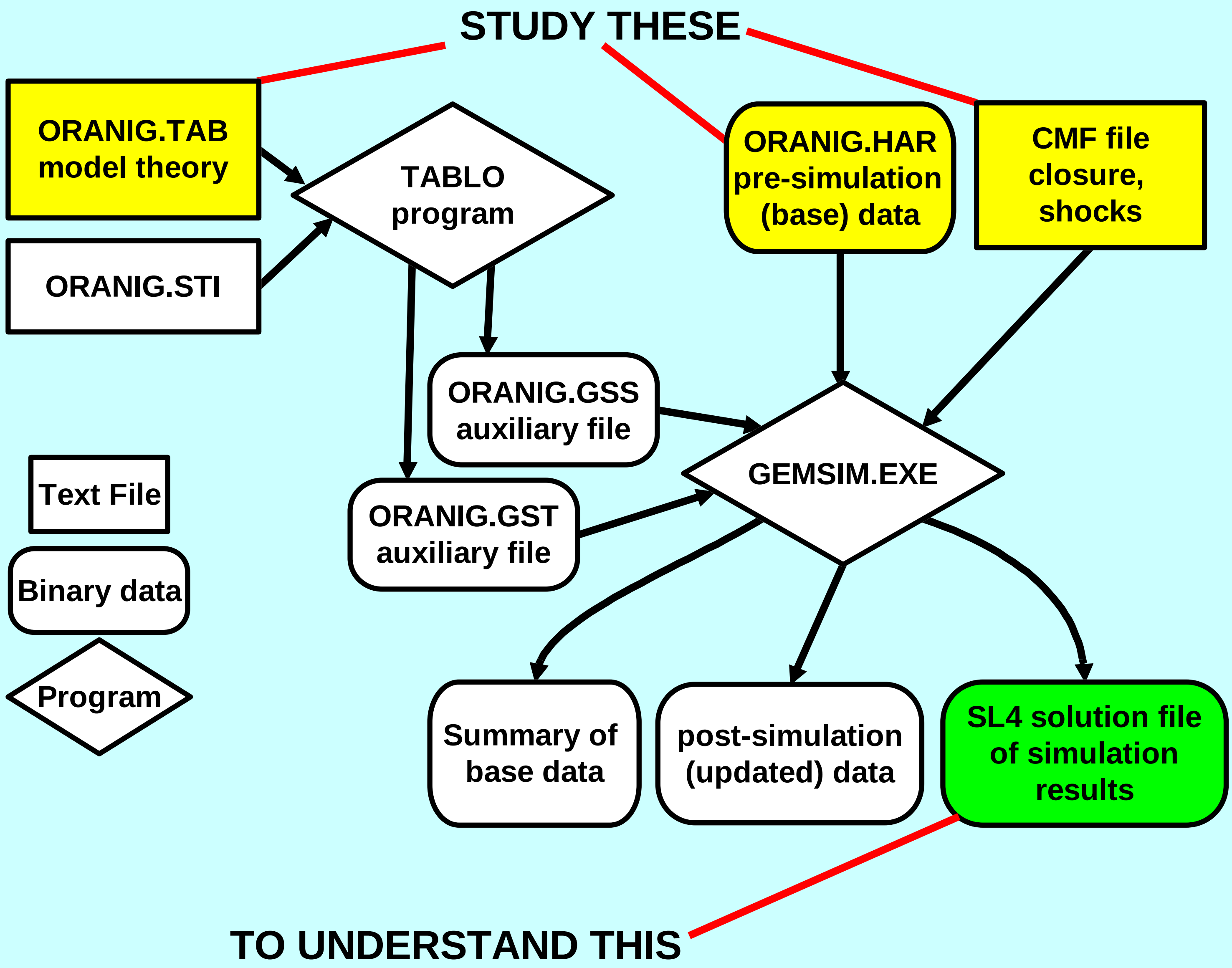
Australian Style	USA style
Percentage change equations	Levels equations
Big, detailed data base	Less detailed data
Industry-specific fixed factors	Mobile capital, labour
Shortrun focus (2 years)	Long, medium run (7-20 yr)
Many prices	Few prices
Used for policy analysis	Prove theoretical point
Winners and Losers	National welfare
Missing macro relations (more exogenous variables)	Closed model:labour supply income-expenditure links
Variety of different closures	One main closure
Input-output database	SAM database
"Dumb" solution procedure	Special algorithm

page no. in document

You will learn

- how microeconomic theory -- cost-minimizing, utility-maximizing -- underlies the equations;
- the use of nested production and utility functions:
- how input-output data is used in equations;
- how model equations are represented in percent change form;
- how choice of exogenous variables makes model more flexible;
- how GEMPACK is used to solve a CGE model.

CGE models mostly similar, so skills will transfer.


STUDY THESE
ORANIG.TAB model theory
ORANIG.STI
TABLO program
ORANIG.HAR pre-simulation (base) data
CMF file closure, shocks
ORANIG.GSS auxiliary file
ORANIG.GST auxiliary file
GEMSIM.EXE
Text File
Binary data
Program
Summary of base data
post-simulation (updated) data
SL4 solution file of simulation results
TO UNDERSTAND THIS

Progress so far . . .

Introduction

→ **Database structure**

Solution method

TABLO language

Production: input decisions

Production: output decisions

Investment: input decisions

Household demands

Export demands

Government demands

Inventory demands

Margin demands

Market clearing

Price equations

Aggregates and indices

Investment allocation

Labour market

Decompositions

Closure

Regional extension

Model Database

memorize numbers

	Size	1 Producers	2 Investors	3 Household	4 Export	5 Government	6 Inventories
		← I →	← I →	← 1 →	← 1 →	← 1 →	← 1 →
Basic Flows	C×S	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	C×S×M	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	C×S	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	O	V1LAB					
Capital	1	V1CAP					
Land	1	V1LND					
Production Tax	1	V1PTX					
Other Costs	1	V1OCT					

C = Number of Commodities

I = Number of Industries

S = 2: Domestic,Imported

O = Number of Occupation Types

M = Number of Commodities used as Margins

Size	Joint Production Matrix
	← I →
↑ C ↓	MAKE

Size	Import Duty
	← 1 →
↑ C ↓	V0TAR

Features of Database

- **Commodity flows are valued at "basic prices": do not include user-specific taxes or margins.**
- **For each user of each imported good and each domestic good, there are numbers showing:**
  - **tax levied on that usage.**
  - **usage of several margins (trade, transport).**
- **MAKE multiproduction:**
  - **Each commodity may be produced by several industries.**
  - **Each industry may produce several commodities.**
- **For each industry the total cost of production is equal to the total value of output (column sums of MAKE).**
- **For each commodity the total value of sales is equal to the total value of output (row sums of MAKE).**
- **No data regarding direct taxes or transfers. Not a full SAM.**

Progress so far . . .

Introduction

Database structure

→ **Solution method**

TABLO language

Production: input decisions

Production: output decisions

Investment: input decisions

Household demands

Export demands

Government demands

Inventory demands

Margin demands

Market clearing

Price equations

Aggregates and indices

Investment allocation

Labour market

Decompositions

Closure

Regional extension

Johansen method: overview

1. We start with the model's equations represented in their levels form

2. The equations are linearised: take total differential of each equation

3. Total differential expressions converted to (mostly) % change form

4. Linear equations evaluated at initial solution to the levels model

5. Exog. variables chosen. Model then solved for movements in endog. variables, given user-specified values for exog. variables.

But, a problem: Linearisation error

Multi-step, extrapolation

Percent-change equations - examples

Levels form: $A = B + C$

Ordinary change form: $\Delta A = \Delta B + \Delta C$

Convert to % change form: $A(\underline{100.\Delta A/A}) = B(\underline{100.\Delta B/B}) + C(\underline{100.\Delta C/C})$

$A\ \underline{a} = B\ \underline{b} + C\ \underline{c}$

Typically two ways of expressing % change form

Intermediate form: $A\ a = B\ b + C\ c$

Percentage change (share) form: $a = S_b\ b + S_c\ c$

where $S_b = B/A$; $S_c = C/A$

Percent-change equations - examples

Levels form: $A = B\,C$

Ordinary change form: $\Delta A = \Delta B\, C + \Delta C\, B$

Convert to % change form:

$$A(100.\Delta A/A) = BC(100.\Delta B/B) + BC(100.\Delta C/C)$$

$$A\quad a \qquad = BC\quad b \qquad + BC\quad c$$

$$a \qquad = \qquad b \qquad + \qquad c$$

PRACTICE: $X = F\,P^{\varepsilon}$

Ordinary Change and Percent Change are both linearized

Linearized equations easier for computers to solve

% change equations easier for economists to understand: elasticities

Percent-change Numerical Example

Levels form $Z = X*Y$

2nd-order

Ordinary Change form $\Delta Z = Y*\Delta X + X*\Delta Y$ $[+ \Delta X\ \Delta Y]$

multiply by 100: $100*\Delta Z = 100*Y*\Delta X + 100*X*\Delta Y$

define x = % change in X, so $X*x=100\Delta X$

so: $Z*z = X*Y*x + X*Y*y$

divide by Z=X*Y to get:

Percent Change form $z = x + y$

25%*20%
=5%
=50%- 45%

Initially X=4, Y=5, so Z = X*Y = 20

Suppose x=25%, y=20% [ie, X:4⇒5, Y:5⇒6]

linear approximation z = x + y gives z = 45%

true answer: 30 = 5*6…… = 50% more than original 20

Error 5% is 2nd order term: $z = x+y + x*y/100$

Note: reduce shocks by a factor of 10, error by factor of 100

Johansen method: example

F(Y,X) = 0 the model (thousands of equations)

Y = vector of endogenous variables (explained by model)

X = vector of exogenous variables (set outside model).

For example, a simple 2 equation model (but with no economic content) (see DPPW p. 73 - 79)

(1) $Y_1 = X^{-1/2}$

(2) $Y_2 = 2 - Y_1$

Model in original levels form

or

(1) $Y_1 X^{1/2} - 1 = 0$

(2) $Y_2 - 2 + Y_1 = 0$

Vector function notation

Johansen method (cont.)

We have initial values Y^0, X^0 which are a solution of F :

$$F(Y^0, X^0) = 0$$

We require an initial solution to the levels model

EG: In our simple 2 equation example:

$V^0 = (Y_1^0, Y_2^0, X^0) = (1, 1, 1)$ might be the initial solution

(1) $Y_1 X^{1/2} - 1 = 0$ → $1\ 1^{1/2} - 1 = 0$

(2) $Y_2 - 2 + Y_1 = 0$ → $1 - 2 + 1 = 0$

Johansen method (cont.)

$F_Y(Y,X).dY + F_X(Y,X).dX = 0$

dY, dX are ordinary changes

Linearised model

We prefer percentage changes y = 100dY/Y, x = 100dX/X

$G_Y(Y,X).y + G_X(Y,X).x = 0$

$A.y + B.x = 0$

A = matrix of derivatives of endogenous variables

B = matrix of derivatives of exogenous variables

A and B depend on current values of levels variables: we exploit this in multi-step simulation to increase accuracy (see below)

Johansen method (cont.)

Back to 2 equation example:

(1) $Y_1 X^{1/2} - 1 = 0$

(2) $Y_2 - 2 + Y_1 = 0$

Convert to % change form:

(1a) $2 y_1 + x = 0$

(2a) $Y_2 y_2 + Y_1 y_1 = 0$

Which in matrix form is:

$$\begin{bmatrix} 2 & 0 & 1 \\ Y_1 & Y_2 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ x \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

We can re-write this, distinguishing endogenous and exogenous variables

p4

Johansen method (cont.)

Each column corresponds to a variable

Each row corresponds to an equation

$$\begin{bmatrix} 2 & 0 \\ Y_1 & Y_2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} x = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$G_Y(Y,X) \quad y \quad + \quad G_X(Y,X) \quad x \quad = \quad 0$$

$$A.y + B.x = 0$$

$$y = [- A^{-1} B] x$$

NB: Elasticities depend on initial solution

Johansen method (cont.)

Continuing with our two equation example:

$$y = [- A^{-1} B]\, x$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = - \begin{bmatrix} 2 & 0 \\ Y_1 & Y_2 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} x$$

NB: Elasticities depend on initial solution

<u>Johansen</u>: [- A^{-1} B] evaluated once, using initial solution

<u>Euler</u>: change in x broken into small steps. [- A^{-1} B] is repeatedly re-evaluated at the end of each step. By breaking the movement in x into a sufficiently small number of steps, we can get arbitrarily close to the true solution. <u>Extrapolation:</u> further improves accuracy.

System of linear equations in matrix notation:

$A.y + B.x = 0$

y = vector of endogenous variables (explained by model)

x = vector of exogenous variables (set outside model).

A and B are matrices of coefficients:

each row corresponds to a model equation;

each column corresponds to a single variable.

Express y in terms of x by:

$y = - A^{-1}B.x$ where A^{-1} = inverse of A

A is: square: number of endogenous variable = number of equations

big: thousands or even millions of variables

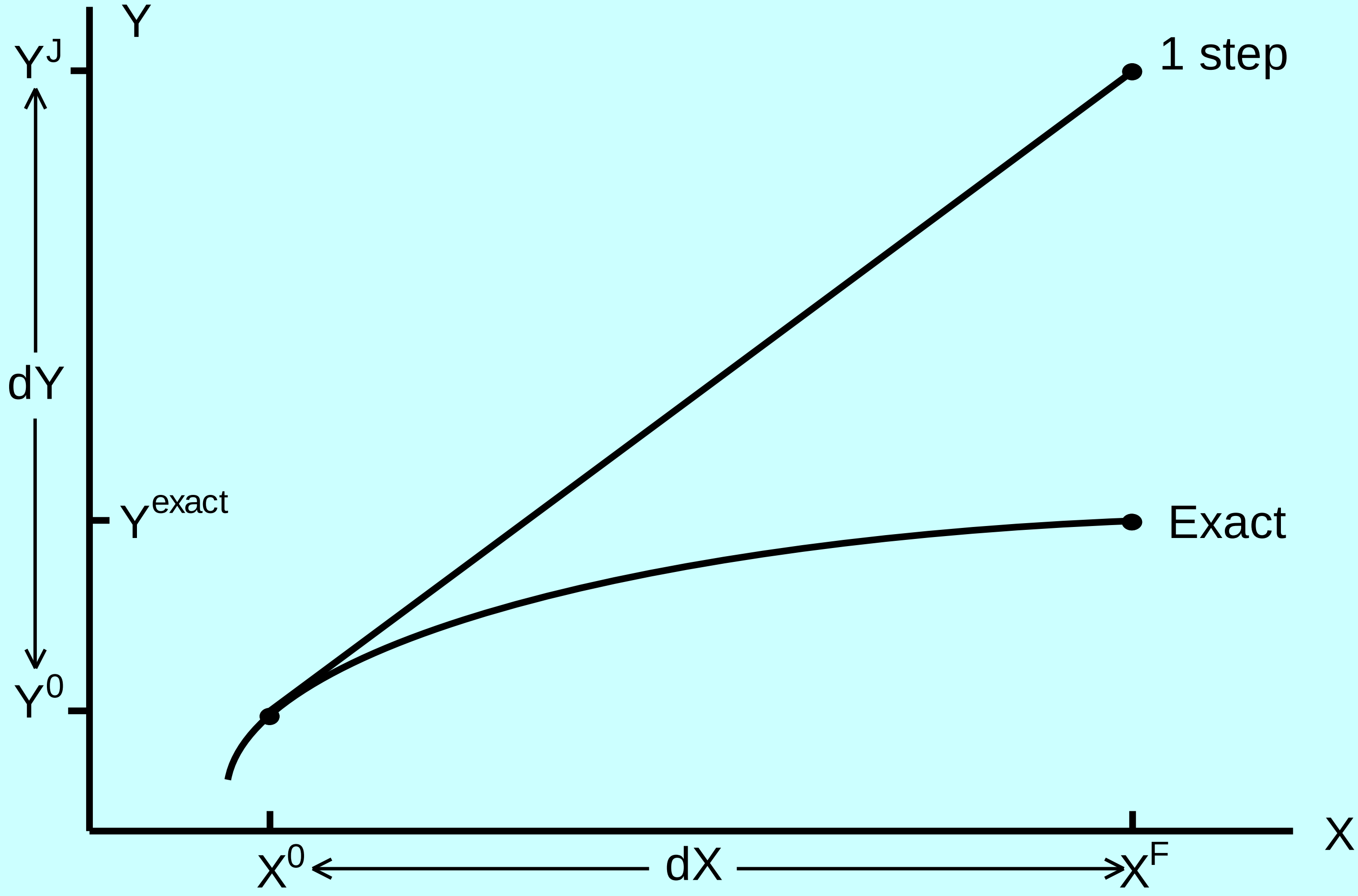
mostly zero: each single equation involves only a few variables.

Linearized equation is

- **just an approximation to levels equation**
- **accurate only for small changes.**

GEMPACK repeatedly solves linear system to get exact solution

Linearization Error

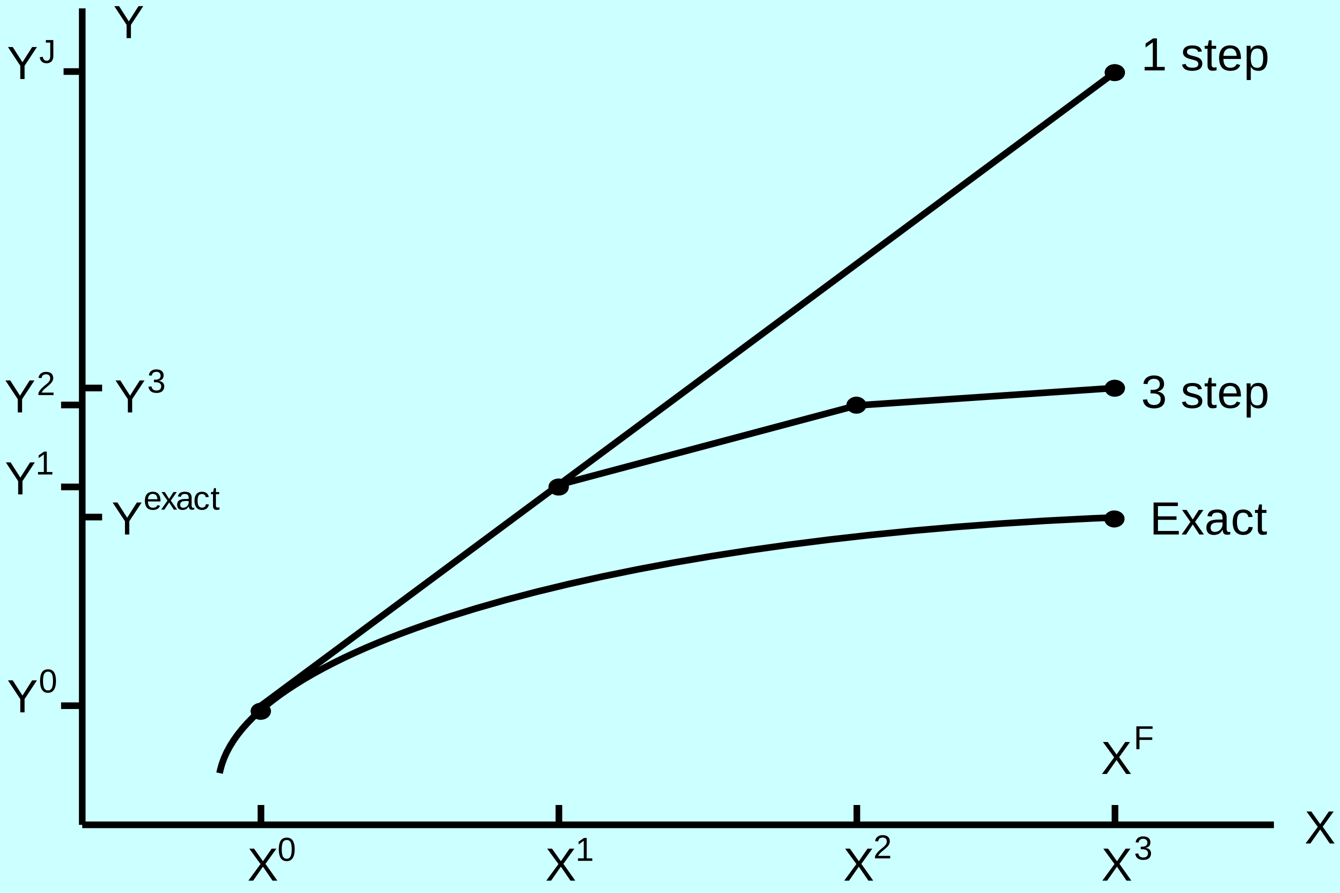


Y^J is *Johansen* estimate.

Error is proportionately less for smaller changes

p5

Breaking large changes in X into a number of steps



Multistep process to reduce linearisation error

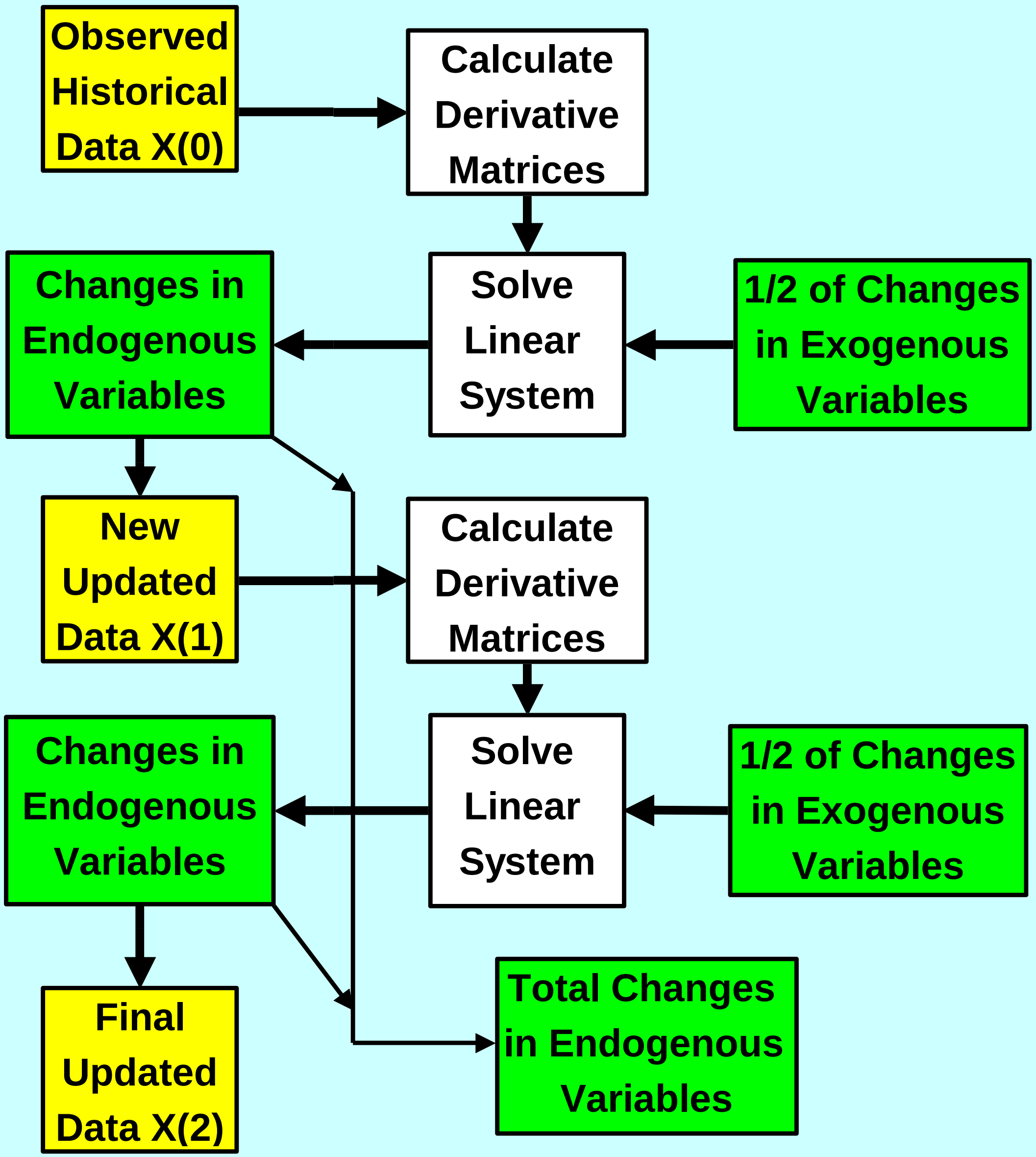
Extrapolating from Johansen and Euler approximations

Method	y	Error
Johansen (1-step)	150%	50%
Euler 2-step	125%	25%
Euler 4-step	112.3%	12.3%
Euler ∞-step (exact)	100%	0

The error follows a rule.

Use results from 3 approximate solutions to estimate exact solution + error bound.

2-step Euler computation in GEMPACK



At each step:

- **compute coefficients from data;**
- **solve linear equation system;**
- **use changes in variables to update data.**

p9

Entire Database is updated at each step

	Size	1 Producers	2 Investors	3 Household	4 Export	5 Government	6 Inventories
		← I →	← I →	← 1 →	← 1 →	← 1 →	← 1 →
Basic Flows	C×S ↕	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	C×S×M ↕	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	C×S ↕	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	O ↕	V1LAB					
Capital	1 ↕	V1CAP					
Land	1 ↕	V1LND					
Production Tax	1 ↕	V1PTX					
Other Costs	1 ↕	V1OCT					

C = Number of Commodities

I = Number of Industries

S = 2: Domestic,Imported

O = Number of Occupation Types

M = Number of Commodities used as Margins

Size	Joint Production Matrix
	← I →
↑ C ↓	MAKE

Size	Import Duty
	← 1 →
↑ C ↓	V0TAR

Progress so far . . .

- Introduction
- Database structure
- Solution method
- → **TABLO language**
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- Government demands
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- Aggregates and indices
- Investment allocation
- Labour market
- Decompositions
- Closure
- Regional extension

The TABLO language

```
Set
 IND # Industries #  (AgricMining, Manufacture, Utilities, Construction,
                     TradeTranspt, FinanProprty, Services);   ! subscript i !
 FAC # Primary factors # (Labour, Capital);                   ! subscript f !
Coefficient
 (all,f,FAC)(all,i,IND) FACTOR(f,i) # Wages and profits #;
 (all,i,IND)                V1PRIM(i)   # Wages plus profits #;
Variable
 (all,i,IND) p1prim(i)  # Price of primary factor composite #;
             p1lab      # Wage rate #;
 (all,i,IND) p1cap(i)   # Rental price of capital #;

Read FACTOR     from file BASEDATA header "1FAC";

Formula (all,i,IND)  V1PRIM(i)  = sum{f,FAC,FACTOR(f,i)};

Equation E_p1prim
 (all,i,IND) V1PRIM(i)*p1prim(i)
         = FACTOR("Labour",i)*p1lab + FACTOR("Capital",i)*p1cap(i);
```

header = location in file

$\sum_{f \in FAC} Factor_{fi}$

Above equation defines average price to each industry of primary factors.

The ORANI-G Naming System

COEFFICIENT

variable

or *GLOSS*

1 intermediate
2 investment
3 households
4 exports
5 government
6 inventories
0 all users

V2TAX(c,s,i)

p1lab_o(i)

x3mar(c,s,m)

V levels value
p % price
x % quantity
del ord.change

c COMmodities
s SouRCe (dom/imp)
i INDustries
m MARgin
o OCCupation
_o add over OCC

bas	**basic (often omitted)**	**cap**	**capital**
mar	**margins**	**lab**	**labour**
tax	**indirect taxes**	**lnd**	**land**
pur	**at purchasers' prices**	**prim**	**all primary factors**
imp	**imports (duty paid)**	**tot**	**total inputs for a user**

Excerpt 1: Files and Sets

```
File     BASEDATA  # Input data file #;
 (new) SUMMARY  # Output for summary and checking data #;
Set
 COM # Commodities #
    read elements from file BASEDATA header "COM";          ! c !
 SRC # Source of commodities # (dom,imp);                   ! s !
 IND # Industries #
    read elements from file BASEDATA header "IND";          ! i !
 OCC # Occupations #
    read elements from file BASEDATA header "OCC";          ! o !
 MAR # Margin commodities #
    read elements from file BASEDATA header "MAR";          ! m !
Subset MAR is subset of COM;
Set NONMAR # Non-margins # = COM - MAR;                     ! n !
```

Core Data and Variables

We begin by declaring variables and data coefficients which appear in many different equations.

Other variables and coefficients will be declared as needed.

Basic Flows

	Size	1	2	3	4	5	6
		Producers	Investors	Household	Export	Government	Inventories
	Size	← I →	← I →	← 1 →	← 1 →	← 1 →	← 1 →
Basic Flows	C×S ↑↓	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	C×S×M ↑↓	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	C×S ↑↓	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	O ↑↓	V1LAB					
Capital	1 ↑↓	V1CAP					
Land	1 ↑↓	V1LND					
Production Tax	1 ↑↓	V1PTX					
Other Costs	1 ↑↓	V1OCT					

C = Number of Commodities

I = Number of Industries

S = 2: Domestic,Imported

O = Number of Occupation Types

M = Number of Commodities used as Margins

	Joint Production Matrix
Size	← I →
↑ C ↓	MAKE

	Import Duty
Size	← 1 →
↑ C ↓	V0TAR

Excerpt 2a: Basic Commodity Flows

```
Coefficient  ! Basic flows of commodities (excluding margin demands)!
 (all,c,COM)(all,s,SRC)(all,i,IND)  V1BAS(c,s,i)  # Intrmediate basic flows #;
 (all,c,COM)(all,s,SRC)(all,i,IND)  V2BAS(c,s,i)  # Investment basic flows #;
 (all,c,COM)(all,s,SRC)             V3BAS(c,s)    # Household basic flows #;
 (all,c,COM)                        V4BAS(c)      # Export basic flows #;
 (all,c,COM)(all,s,SRC)             V5BAS(c,s)    # Govment basic flows #;
 (all,c,COM)(all,s,SRC)             V6BAS(c,s)    # Inventories basic flows #;

Read
 V1BAS from file BASEDATA header "1BAS";
 V2BAS from file BASEDATA header "2BAS";
 V3BAS from file BASEDATA header "3BAS";
 V4BAS from file BASEDATA header "4BAS";
 V5BAS from file BASEDATA header "5BAS";
 V6BAS from file BASEDATA header "6BAS";
```

Coefficients and Variables

Coefficients

example: V1BAS(c,s,i) UPPER CASE

Mostly values

Either read from file

or computed with formulae

Constant during each step

Variables

example: x1bas (c,s,i) lower case

Often prices or quantities

Percent or ordinary change

Related via equations

Exogenous or endogenous

Vary during each step

Excerpt 2b: Basic Commodity Flows

```
Variable  ! used to update flows !
 (all,c,COM)(all,s,SRC)(all,i,IND)     x1(c,s,i)  # Intermediate demands #;
. . . . . . . . . . . . . . . . . . . . . . . . . . .
 (all,c,COM)                      x4(c)     # Export basic demands #;
 (all,c,COM)(all,s,SRC)     x5(c,s) # Government basic demands #;
 (change) (all,c,COM)(all,s,SRC) delx6(c,s) # Inventories #;
 (all,c,COM)(all,s,SRC)      p0(c,s) # Basic prices for local users #;
 (all,c,COM)                 pe(c)     # Basic price of exportables #;
 (change)(all,c,COM)(all,s,SRC)  delV6(c,s)    # inventories #;
Update
 (all,c,COM)(all,s,SRC)(all,i,IND)  V1BAS(c,s,i)  = p0(c,s)*x1(c,s,i);
. . . . . . . . . . . . . . . . . . . . . . . . . . .
 (all,c,COM)                        V4BAS(c)      = pe(c)*x4(c);
 (all,c,COM)(all,s,SRC)             V5BAS(c,s)    = p0(c,s)*x5(c,s);
 (change)(all,c,COM)(all,s,SRC)     V6BAS(c,s)    = delV6(c,s);
```

Ordinary Change Variables

Variable ! used to update flows !

(all,c,COM)(all,s,SRC)(all,i,IND) x1(c,s,i) # Intermediate #;

. . . . .

(change) (all,c,COM)(all,s,SRC) delx6(c,s) # Inventories #;

By default variables are percent change.

Exact, multi-step solutions made from a sequence of small percent changes.

Small percent changes do not allow sign change (eg, from 2 to -1).

Variables which change sign must be ordinary change.

Update Statements

Default (product) update
V ➜ V(1+p/100+x/100)

Update

(all,c,COM)(all,s,SRC)(all,i,IND) V1BAS(c,s,i) = p0(c,s)*x1(c,s,i);

. . . . .

(all,c,COM) V4BAS(c) = pe(c)*x4(c);

(all,c,COM)(all,s,SRC) V5BAS(c,s) = p0(c,s)*x5(c,s);

(change)(all,c,COM)(all,s,SRC) V6BAS(c,s) = delV6(c,s);

Ordinary change update
V ➜ V + ΔV

Updates: the vital link between variables and data
show how data relates to variables

Margins

	Size	1 Producers ← I →	2 Investors ← I →	3 Household ← 1 →	4 Export ← 1 →	5 Government ← 1 →	6 Inventories ← 1 →
Basic Flows	C×S ↑↓	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	C×S×M ↑↓	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	C×S ↑↓	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	O ↑↓	V1LAB					
Capital	1 ↑↓	V1CAP					
Land	1 ↑↓	V1LND					
Production Tax	1 ↑↓	V1PTX					
Other Costs	1 ↑↓	V1OCT					

C = Number of Commodities

I = Number of Industries

S = 2: Domestic,Imported

O = Number of Occupation Types

M = Number of Commodities used as Margins

Size	Joint Production Matrix ← I →
↑ C ↓	MAKE

Size	Import Duty ← 1 →
↑ C ↓	V0TAR

Excerpt 3a: Margin Flows

Coefficient

(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)

V1MAR(c,s,i,m) # Intermediate margins #;

(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)

V2MAR(c,s,i,m) # Investment margins #;

(all,c,COM)(all,s,SRC)(all,m,MAR)

V3MAR(c,s,m) # Households margins #;

(all,c,COM)(all,m,MAR) V4MAR(c,m) # Export margins #;

(all,c,COM)(all,s,SRC)(all,m,MAR) V5MAR(c,s,m) # Government #;

Read

V1MAR from file BASEDATA header "1MAR";

V2MAR from file BASEDATA header "2MAR";

V3MAR from file BASEDATA header "3MAR";

V4MAR from file BASEDATA header "4MAR";

V5MAR from file BASEDATA header "5MAR";

m: transport bringing
s: imported
c: leather to
i: shoe industry

- ***Note: no margins on inventories***

Excerpt 3b: Margin Flows

```
Variable ! Variables used to update above flows !
 (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
    x1mar(c,s,i,m)# Intermediate margin demand #;
 (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
    x2mar(c,s,i,m)# Investment margin demands #;
 (all,c,COM)(all,s,SRC)(all,m,MAR)
    x3mar(c,s,m)  # Household margin demands #;
 (all,c,COM)
    p0dom(c) # Basic price of domestic goods = p0(c,"dom") #;
Update
 (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
    V1MAR(c,s,i,m)  = p0dom(m)*x1mar(c,s,i,m);
 (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
    V2MAR(c,s,i,m)  = p0dom(m)*x2mar(c,s,i,m);
 (all,c,COM)(all,s,SRC)(all,m,MAR)
    V3MAR(c,s,m)    = p0dom(m)*x3mar(c,s,m);
```

not shown:
4: export
5: government

m: transport bringing
s: imported
c: leather to
i: shoe industry

Commodity Taxes

	Size	1 Producers	2 Investors	3 Household	4 Export	5 Government	6 Inventories
		← I →	← I →	← 1 →	← 1 →	← 1 →	← 1 →
Basic Flows	↑ C×S ↓	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	↑ C×S×M ↓	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	↑ C×S ↓	**V1TAX**	**V2TAX**	**V3TAX**	**V4TAX**	**V5TAX**	n/a
Labour	↑ O ↓	V1LAB					
Capital	↑ 1 ↓	V1CAP					
Land	↑ 1 ↓	V1LND					
Production Tax	↑ 1 ↓	V1PTX					
Other Costs	↑ 1 ↓	V1OCT					

C = Number of Commodities

I = Number of Industries

S = 2: Domestic,Imported

O = Number of Occupation Types

M = Number of Commodities used as Margins

Size	Joint Production Matrix
	← I →
↑ C ↓	MAKE

Size	Import Duty
	← 1 →
↑ C ↓	V0TAR

Excerpt 4a: Commodity Taxes

```
Coefficient  ! Taxes on Basic Flows!
 (all,c,COM)(all,s,SRC)(all,i,IND)  V1TAX(c,s,i) # Taxes on intermediate #;
 (all,c,COM)(all,s,SRC)(all,i,IND)  V2TAX(c,s,i) # Taxes on investment #;
 (all,c,COM)(all,s,SRC)             V3TAX(c,s)   # Taxes on h'holds #;
 (all,c,COM)                        V4TAX(c)     # Taxes on export #;
 (all,c,COM)(all,s,SRC)             V5TAX(c,s)   # Taxes on gov'ment #;
Read
 V1TAX from file BASEDATA header "1TAX";
 V2TAX from file BASEDATA header "2TAX";
 V3TAX from file BASEDATA header "3TAX";
 V4TAX from file BASEDATA header "4TAX";
 V5TAX from file BASEDATA header "5TAX";
```

Simulate: no tax on diesel for farmers
subsidy on cement and bricks used to build schools

Excerpt 4b: Commodity Taxes

```
Variable
 (change)(all,c,COM)(all,s,SRC)(all,i,IND) delV1TAX(c,s,i)  # Interm tax rev #;
 (change)(all,c,COM)(all,s,SRC)(all,i,IND) delV2TAX(c,s,i)  # Invest tax rev #;
 (change)(all,c,COM)(all,s,SRC)            delV3TAX(c,s)    # H'hold tax rev #;
 (change)(all,c,COM)                       delV4TAX(c)      # Export tax rev #;
 (change)(all,c,COM)(all,s,SRC)            delV5TAX(c,s)    # Govmnt tax rev #;
Update
 (change)(all,c,COM)(all,s,SRC)(all,i,IND)  V1TAX(c,s,i) = delV1TAX(c,s,i);
 (change)(all,c,COM)(all,s,SRC)(all,i,IND)  V2TAX(c,s,i) = delV2TAX(c,s,i);
 (change)(all,c,COM)(all,s,SRC)             V3TAX(c,s)   = delV3TAX(c,s);
 (change)(all,c,COM)                        V4TAX(c)     = delV4TAX(c);
 (change)(all,c,COM)(all,s,SRC)             V5TAX(c,s)   = delV5TAX(c,s);
```

Note: equations defining delV#TAX tax variables appear later; they depend on type of tax;

Primary Factors, etc

	Size	1 Producers	2 Investors	3 Household	4 Export	5 Government	6 Inventories
		← I →	← I →	← 1 →	← 1 →	← 1 →	← 1 →
Basic Flows	C×S ↕	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	C×S×M ↕	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	C×S ↕	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	O ↕	V1LAB					
Capital	1 ↕	V1CAP					
Land	1 ↕	V1LND					
Production Tax	1 ↕	V1PTX					
Other Costs	1 ↕	V1OCT					

C = Number of Commodities

I = Number of Industries

S = 2: Domestic,Imported

O = Number of Occupation Types

M = Number of Commodities used as Margins

Size	Joint Production Matrix
	← I →
↑ C ↓	MAKE

Size	Import Duty
	← 1 →
↑ C ↓	V0TAR

Excerpt 5: Primary Factors etc

Capital example

```
Coefficient   (all,i,IND)     V1CAP(i)     # Capital rentals #;

Read V1CAP from file BASEDATA header "1CAP";

Variable    (all,i,IND)  x1cap(i)    # Current capital stock #;
            (all,i,IND)  p1cap(i)    # Rental price of capital #;

Update      (all,i,IND)  V1CAP(i)  = p1cap(i)*x1cap(i);
```

Excerpt 5a: Primary Factors etc

```
Coefficient
 (all,i,IND)(all,o,OCC)  V1LAB(i,o)  # Wage bill matrix #;
 (all,i,IND)              V1CAP(i)    # Capital rentals #;
 (all,i,IND)              V1LND(i)    # Land rentals #;
 (all,i,IND)              V1PTX(i)    # Production tax #;
 (all,i,IND)              V1OCT(i)    # Other cost tickets #;
Read
 V1LAB from file BASEDATA header "1LAB";
 V1CAP from file BASEDATA header "1CAP";
 V1LND from file BASEDATA header "1LND";
 V1PTX from file BASEDATA header "1PTX";
 V1OCT from file BASEDATA header "1OCT";
```

Different skills

Note: V1PTX is ad valorem, V1OCT is specific

Excerpt 5b: Primary Factors etc

```
Variable
 (all,i,IND)(all,o,OCC)   x1lab(i,o)  # Employment by industry and occupation #;
 (all,i,IND)(all,o,OCC)   p1lab(i,o)  # Wages by industry and occupation #;
 (all,i,IND)  x1cap(i)        # Current capital stock #;
 (all,i,IND)  p1cap(i)        # Rental price of capital #;
 (all,i,IND)  x1lnd(i)        # Use of land #;
 (all,i,IND)  p1lnd(i)        # Rental price of land #;
 (change)(all,i,IND) delV1PTX(i) # Ordinary change in production tax revenue #;
 (all,i,IND)  x1oct(i)        # Demand for "other cost" tickets #;
 (all,i,IND)  p1oct(i)        # Price of "other cost" tickets #;
Update
 (all,i,IND)(all,o,OCC)  V1LAB(i,o)  = p1lab(i,o)*x1lab(i,o);
 (all,i,IND)             V1CAP(i)    = p1cap(i)*x1cap(i);
 (all,i,IND)             V1LND(i)    = p1lnd(i)*x1lnd(i);
(change)(all,i,IND)      V1PTX(i)    = delV1PTX(i);
 (all,i,IND)             V1OCT(i)    = p1oct(i)*x1oct(i);
```

equation later

Excerpt 5c: Tariffs

```
Coefficient (all,c,COM) V0TAR(c)  # Tariff revenue #;
Read V0TAR from file BASEDATA header "0TAR";
Variable (all,c,COM) (change)
    delV0TAR(c) # Ordinary change in tariff revenue #;
Update (change)  (all,c,COM) V0TAR(c) = delV0TAR(c);
```

Note: tariff is independent of user, unlike V#TAX matrices.

Excerpt 6a: purchaser's values (basic + margins + taxes)

```
Coefficient
 (all,c,COM)(all,s,SRC)(all,i,IND)  V1PUR(c,s,i) # Intermediate purch. value #;
 (all,c,COM)(all,s,SRC)(all,i,IND)  V2PUR(c,s,i)  # Investment purch. value #;
 (all,c,COM)(all,s,SRC)             V3PUR(c,s)  # Households purch. value #;
 (all,c,COM)                        V4PUR(c)    # Export purch. value #;
 (all,c,COM)(all,s,SRC)             V5PUR(c,s)  # Government purch. value #;
Formula
 (all,c,COM)(all,s,SRC)(all,i,IND)
   V1PUR(c,s,i)  = V1BAS(c,s,i) + V1TAX(c,s,i)
                       + sum{m,MAR, V1MAR(c,s,i,m)};
. . . . . . . . . . . . .
 (all,c,COM)(all,s,SRC)
   V5PUR(c,s)    = V5BAS(c,s)   + V5TAX(c,s)
                       + sum{m,MAR, V5MAR(c,s,m)};
```

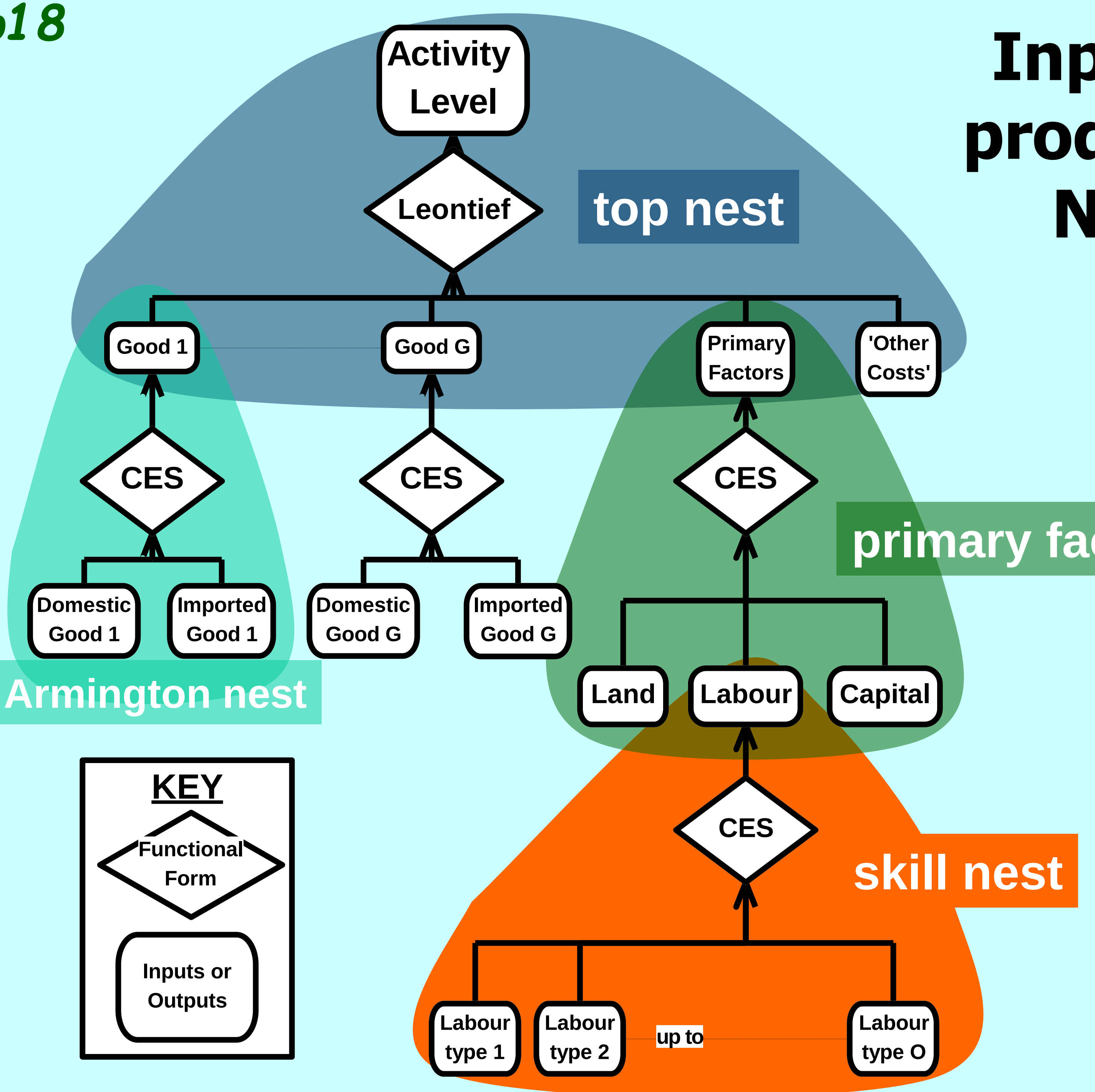
Excerpt 6b: purchaser's prices

```
Variable
 (all,c,COM)(all,s,SRC)(all,i,IND) p1(c,s,i) # Purchaser's price, intermediate #;
 (all,c,COM)(all,s,SRC)(all,i,IND) p2(c,s,i) # Purchaser's price, investment #;
 (all,c,COM)(all,s,SRC)             p3(c,s)   # Purchaser's price, household #;
 (all,c,COM)                        p4(c)     # Purchaser's price, exports, loc$ #;
 (all,c,COM)(all,s,SRC)             p5(c,s)   # Purchaser's price, government #;
```

Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- → **Production: input decisions**
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- Government demands
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- Aggregates and indices
- Investment allocation
- Labour market
- Decompositions
- Closure
- Regional extension

Inputs to production: Nests



Nested Structure of production

In each industry: Output = function of inputs:

output = F(inputs) = F(Labour, Capital, Land, dom goods, imp goods)

Separability assumptions simplify the production structure:

output = F(primary factor composite, composite goods)

where:

primary factor composite = CES(Labour, Capital. Land)

labour = CES(Various skill grades)

composite good (i) = CES(domestic good (i), imported good (i))

All industries share common production structure.

BUT: Input proportions and behavioural parameters vary.

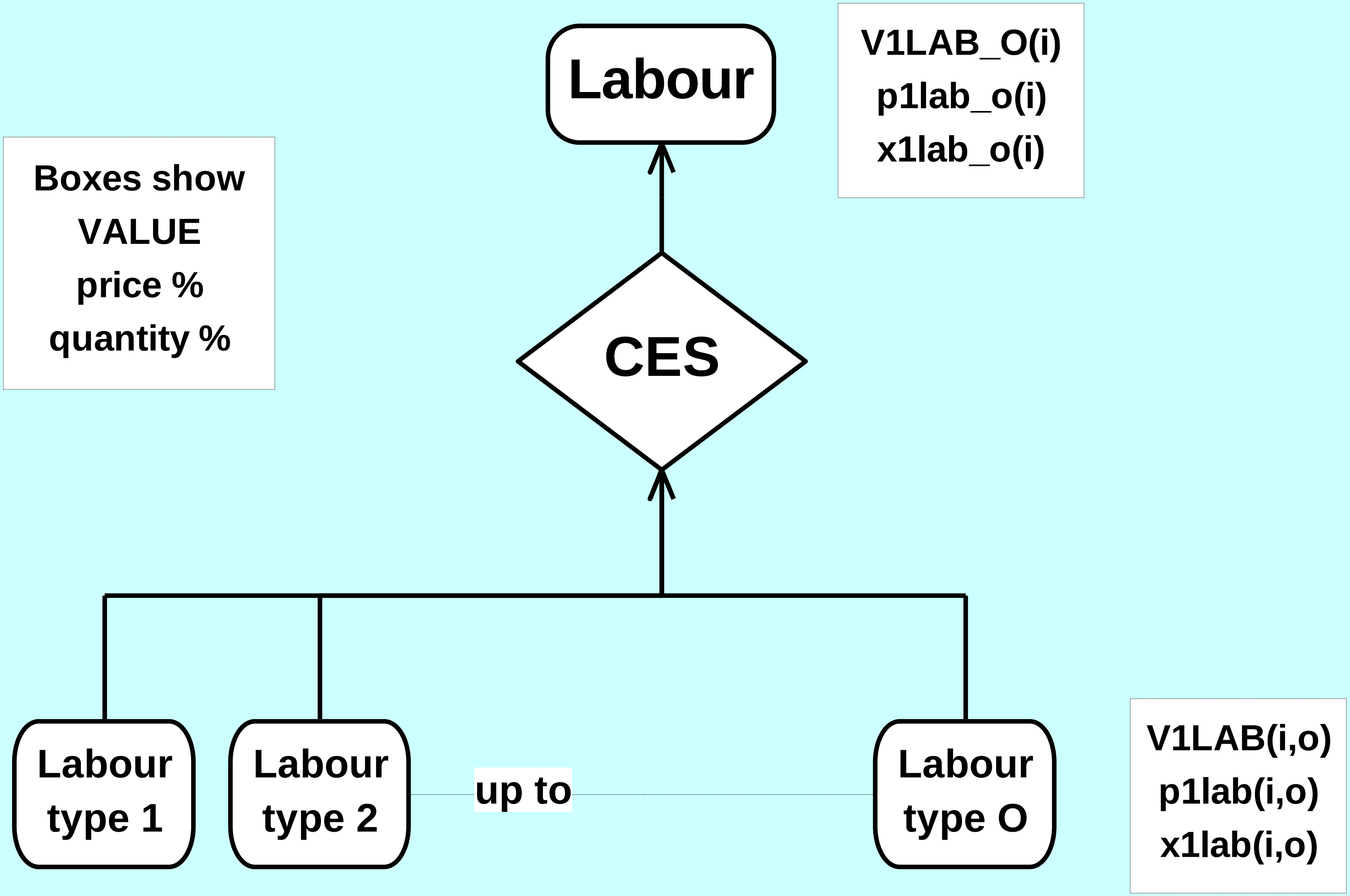
Nesting is like staged decisions:

First decide how much leather to use—based on output.

Then decide import/domestic proportions, depending on the relative prices of local and foreign leather.

Each nest requires 2 or 3 equations.

Excerpt 7: Skill Mix



Excerpt 7: Skill Mix

Problem: for each industry i, choose labour inputs X1LAB(i,o)
to minimize labour cost:

sum{o,OCC, P1LAB(i,o)*X1LAB(i,o)}

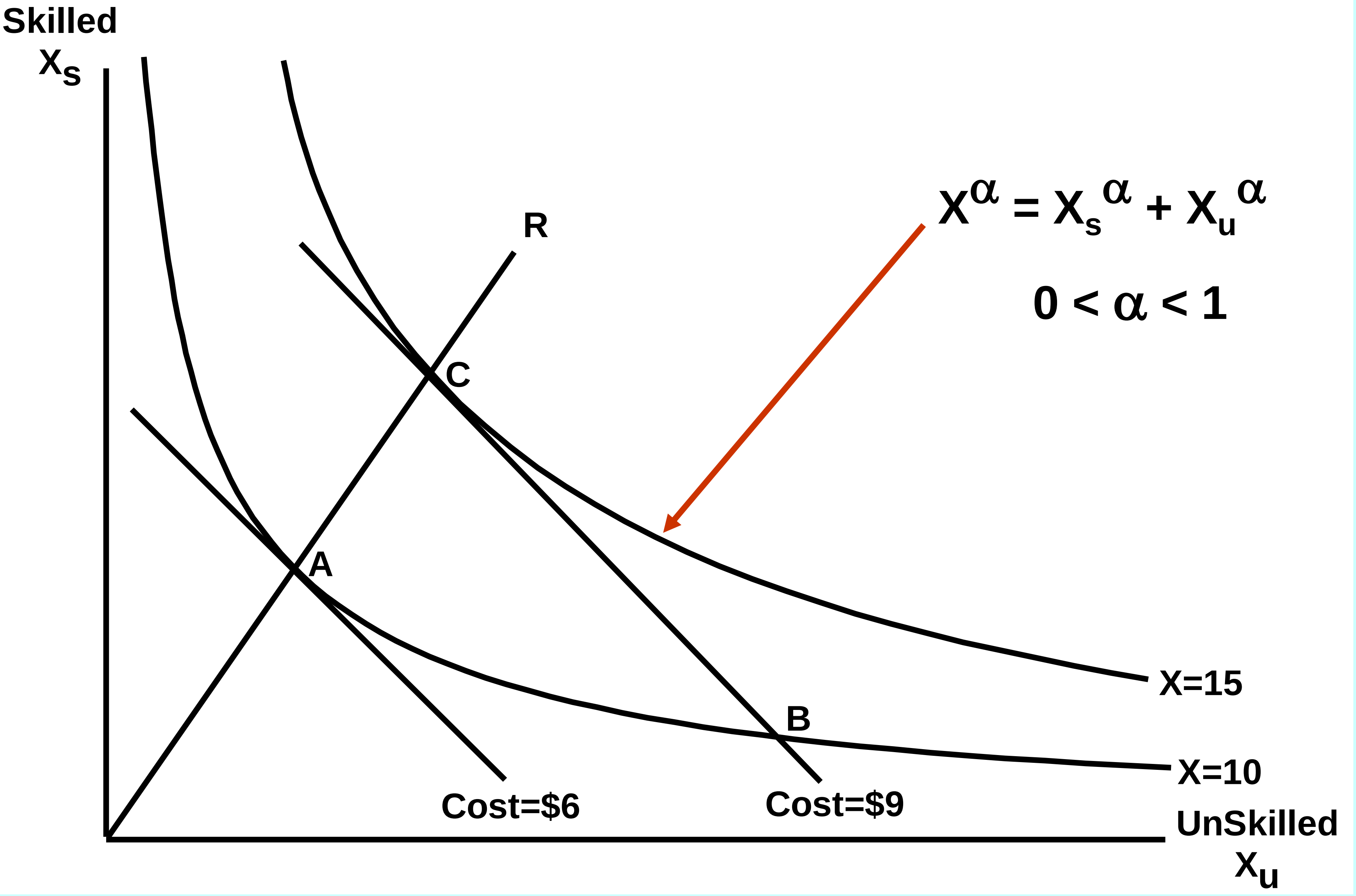
such that X1LAB_O(i) = CES(All,o,OCC: X1LAB(i,o))

given

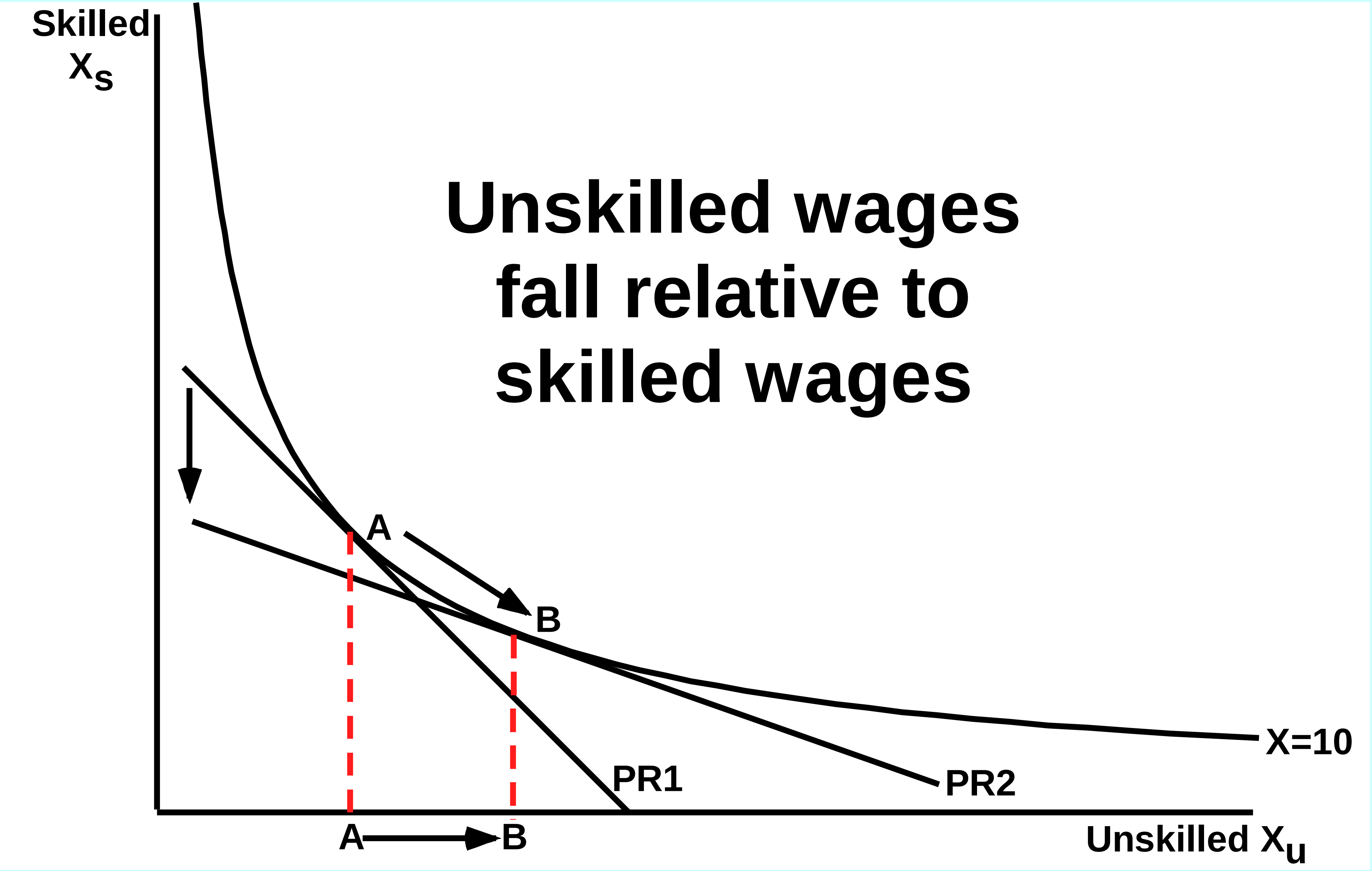
Coefficient
(all,i,IND) SIGMA1LAB(i) # CES substitution between skills #;
(all,i,IND) V1LAB_O(i) # Total labour bill in industry i #;
TINY# Small number to prevent zerodivides or singular matrix #;
Read SIGMA1LAB from file BASEDATA header "SLAB";
Formula (all,i,IND) V1LAB_O(i) = sum{o,OCC, V1LAB(i,o)};
TINY = 0.000000000001;

add over OCC

CES Skill Substitution



Effect of Price Change



Deriving the CES demand equations

See ORANI-G document Appendix A

Excerpt 7: Skill Mix

Variable

(all,i,IND) p1lab_o(i) # Price to each industry of labour composite #;

(all,i,IND) x1lab_o(i) # Effective labour input #;

Equation

E_x1lab # Demand for labour by industry and skill group #

(all,i,IND)(all,o,OCC)

x1lab(i,o) = x1lab_o(i) - SIGMA1LAB(i)*[p1lab(i,o) - p1lab_o(i)];

E_p1lab_o # Price to each industry of labour composite #

(all,i,IND) [TINY+V1LAB_O(i)]*p1lab_o(i)

= sum{o,OCC, V1LAB(i,o)*p1lab(i,o)};

MEMORIZE $x_o = x_{average} - \sigma[p_o - p_{average}]$

CES PATTERN $p_{average} = \Sigma s_o.p_o$

relative price term

The many faces of CES

normal nest form

$x_1 = x_{ave} - \sigma[p_1 - p_{ave}]$

$x_2 = x_{ave} - \sigma[p_2 - p_{ave}]$

$x_3 = x_{ave} - \sigma[p_3 - p_{ave}]$

$p_{ave} = S_1p_1 + S_2p_2 + S_3p_3$

multiply by share

$S_1x_1 = S_1x_{ave} - \sigma S_1 [p_1 - p_{ave}]$

$S_2x_2 = S_2x_{ave} - \sigma S_2 [p_2 - p_{ave}]$

$S_3x_3 = S_3x_{ave} - \sigma S_3 [p_3 - p_{ave}]$

add all three (price terms vanish)

$$S_1x_1 + S_2x_2 + S_3x_3 = x_{ave}$$

concentrated or pre-optimized production function

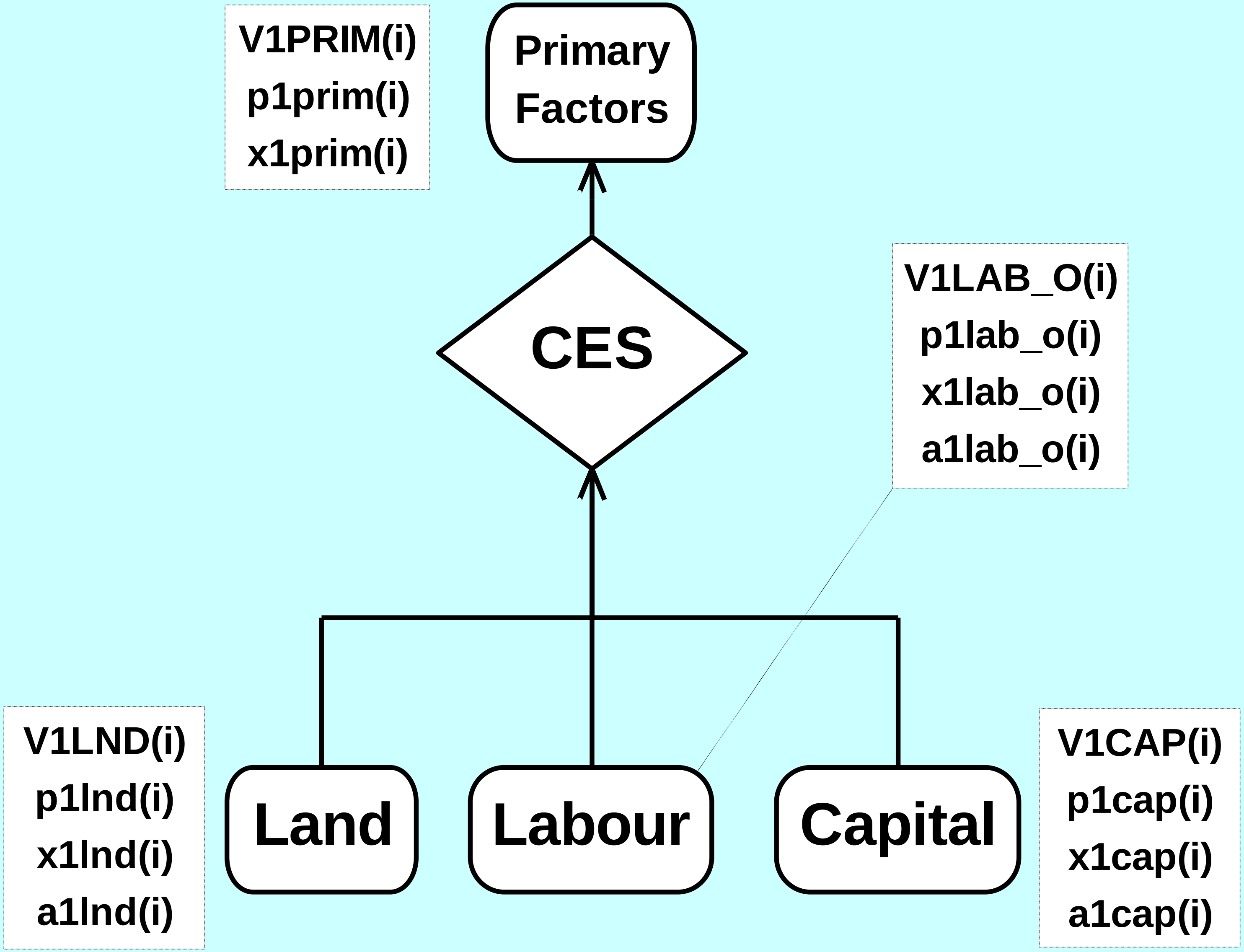
subtract

$$x_2 - x_3 = -\sigma[p_2 - p_3]$$

each new equation can be used to replace one original equation

p20

Excerpt 8: Primary factor Mix



Excerpt 8a: Primary factor Mix

```
X1PRIM(i) =  CES(       X1LAB_O(i)/A1LAB_O(i),
                        X1CAP(i)/A1CAP(i),
                        X1LND(i)/A1LND(i)    )
```

quantity-augmenting technical change

```
Coefficient (all,i,IND) SIGMA1PRIM(i) # CES substitution, primary factors #;
Read SIGMA1PRIM from file BASEDATA header "P028";
Coefficient (all,i,IND) V1PRIM(i) # Total factor input to industry i#;
Formula     (all,i,IND) V1PRIM(i) = V1LAB_O(i)+ V1CAP(i) + V1LND(i);
Variable
 (all,i,IND) p1prim(i)  # Effective price of primary factor composite #;
 (all,i,IND) x1prim(i)  # Primary factor composite #;
 (all,i,IND) a1lab_o(i) # Labor-augmenting technical change #;
 (all,i,IND) a1cap(i)   # Capital-augmenting technical change #;
 (all,i,IND) a1lnd(i)   # Land-augmenting technical change #;
(change)(all,i,IND) delV1PRIM(i)#Ordinary change, cost of primary factors#;
```

Excerpt 8b: Primary factor Mix

(x-a): effective input

Equation

E_x1lab_o # Industry demands for effective labour #

(all,i,IND) x1lab_o(i) - a1lab_o(i) =

x1prim(i) - SIGMA1PRIM(i)*[p1lab_o(i) + a1lab_o(i) - p1prim(i)];

E_p1cap # Industry demands for capital #

(p+a): price of effective input

(all,i,IND) x1cap(i) - a1cap(i) =

x1prim(i) - SIGMA1PRIM(i)*[p1cap(i) + a1cap(i) - p1prim(i)];

E_p1lnd # Industry demands for land #

(all,i,IND) x1lnd(i) - a1lnd(i) =

x1prim(i) - SIGMA1PRIM(i)*[p1lnd(i) + a1lnd(i) - p1prim(i)];

E_p1prim # Effective price term for factor demand equations #

(all,i,IND) V1PRIM(i)*p1prim(i) = V1LAB_O(i)*[p1lab_o(i) + a1lab_o(i)]

+ V1CAP(i)*[p1cap(i) + a1cap(i)] + V1LND(i)*[p1lnd(i) + a1lnd(i)];

Excerpt 8: Primary Factor Mix

Original $x_o = x_{average} - \sigma[p_o - p_{average}]$

CES Pattern $p_{average} = \Sigma S_o.p_o$

$x \rightarrow x\text{-}a \qquad p \rightarrow p+a$

With $x_f - a_f = x_{average} - \sigma[p_f + a_f - p_{average}]$

Tech Change $p_{average} = \Sigma S_f.[p_f + a_f]$

Excerpt 8c: Cost of Primary factors

Equation

E_delV1PRIM # Ordinary change in cost, primary factors #

(all,i,IND) 100*delV1PRIM(i) =

V1CAP(i) * [p1cap(i) + x1cap(i)]

+ V1LND(i) * [p1lnd(i) + x1lnd(i)]

+ sum{o,OCC, V1LAB(i,o)* [p1lab(i,o) + x1lab(i,o)]};

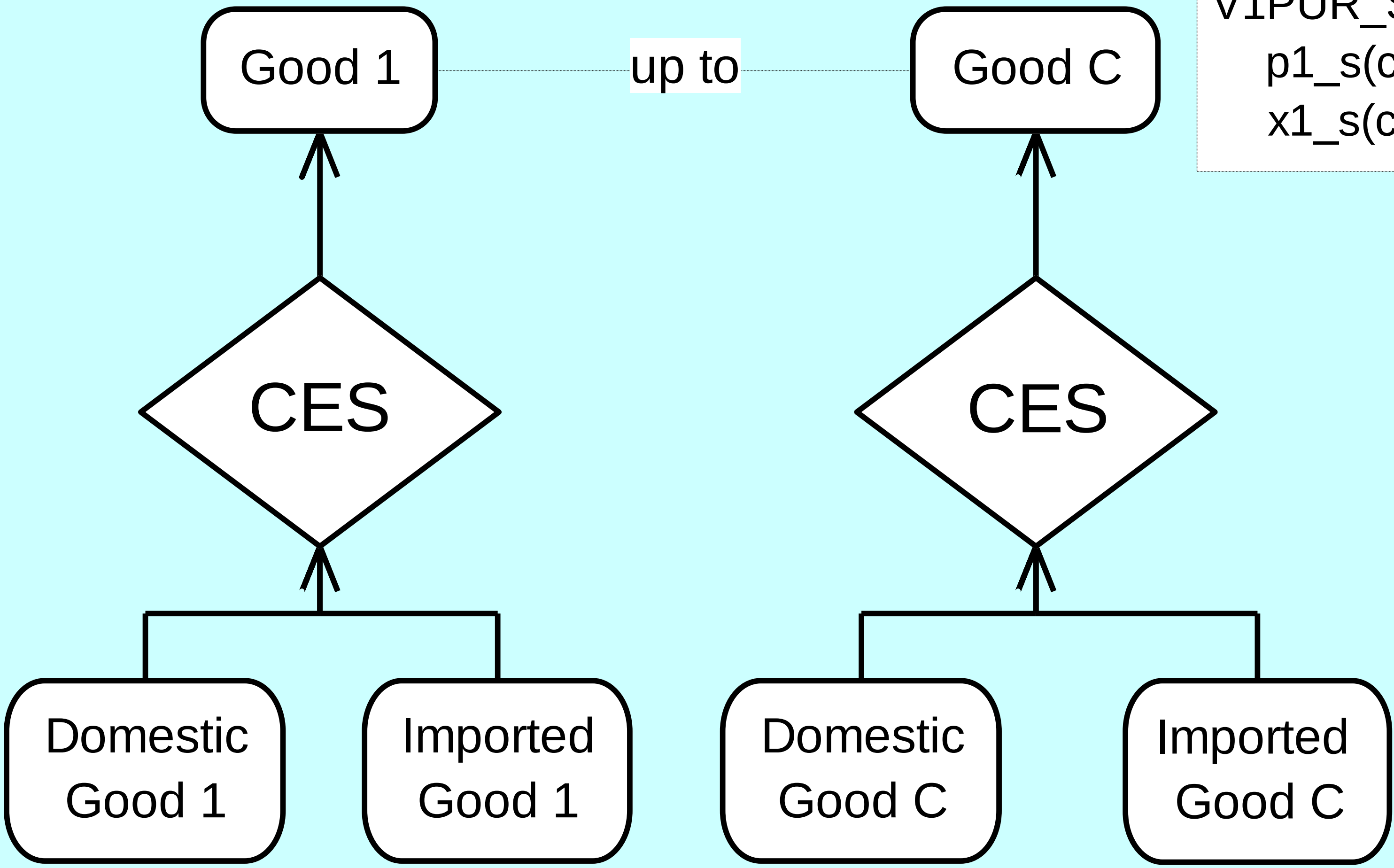
100 times change in value

V = value = P.X so v = p + x

V.v = 100 times change in V = V*[p+x]

. . . will prove a convenient representation for the zero pure profit equation . . .

Excerpt 9a: Intermediate Sourcing



V1PUR_S(c,i)
p1_s(c,i)
x1_s(c,i)

Boxes show
VALUE
price %
quantity %

V1PUR(c,s,i)
p1(c,s,i)
x1(c,s,i)
a1(c,s,i)

Excerpt 9a: Intermediate Sourcing

```
X1_S(c,i) = CES( All,s,SRC: X1(c,s,i)/A1(c,s,i) )
Variable
 (all,c,COM)(all,s,SRC)(all,i,IND) a1(c,s,i) # Intermediate basic tech change #;
 (all,c,COM)(all,i,IND) x1_s(c,i)    # Intermediate use of imp/dom composite #;
 (all,c,COM)(all,i,IND) p1_s(c,i)    # Price, intermediate imp/dom composite #;
Coefficient
 (all,c,COM)                SIGMA1(c)    # Armington elasticities: intermediate #;
 (all,c,COM)(all,i,IND) V1PUR_S(c,i) # Dom+imp intermediate purch. value #;
 (all,c,COM)(all,s,SRC)(all,i,IND)   S1(c,s,i) # Intermediate source shares #;
Read SIGMA1 from file BASEDATA header "1ARM";
Zerodivide default 0.5;
Formula
 (all,c,COM)(all,i,IND)          V1PUR_S(c,i) = sum{s,SRC, V1PUR(c,s,i)};
 (all,c,COM)(all,s,SRC)(all,i,IND) S1(c,s,i)    = V1PUR(c,s,i) / V1PUR_S(c,i);
Zerodivide off;
```

alternative to TINY

Excerpt 9b: Intermediate Sourcing

X1_S(c,i) = CES(All,s,SRC: X1(c,s,i)/A1(c,s,i))

Equation E_x1 # Source-specific commodity demands #
(all,c,COM)(all,s,SRC)(all,i,IND)
x1(c,s,i)-a1(c,s,i) =
x1_s(c,i) -SIGMA1(c)*[p1(c,s,i) +a1(c,s,i) -p1_s(c,i)];

x-a

Equation E_p1_s # Effective price, commodity composite #
(all,c,COM)(all,i,IND)
p1_s(c,i) = sum{s,SRC, S1(c,s,i)*[p1(c,s,i) + a1(c,s,i)]};

p+a

$$x_s - a_s = x_{average} - \sigma[p_s + a_s - p_{average}]$$

$$p_{average} = \Sigma S_s.[p_s + a_s]$$

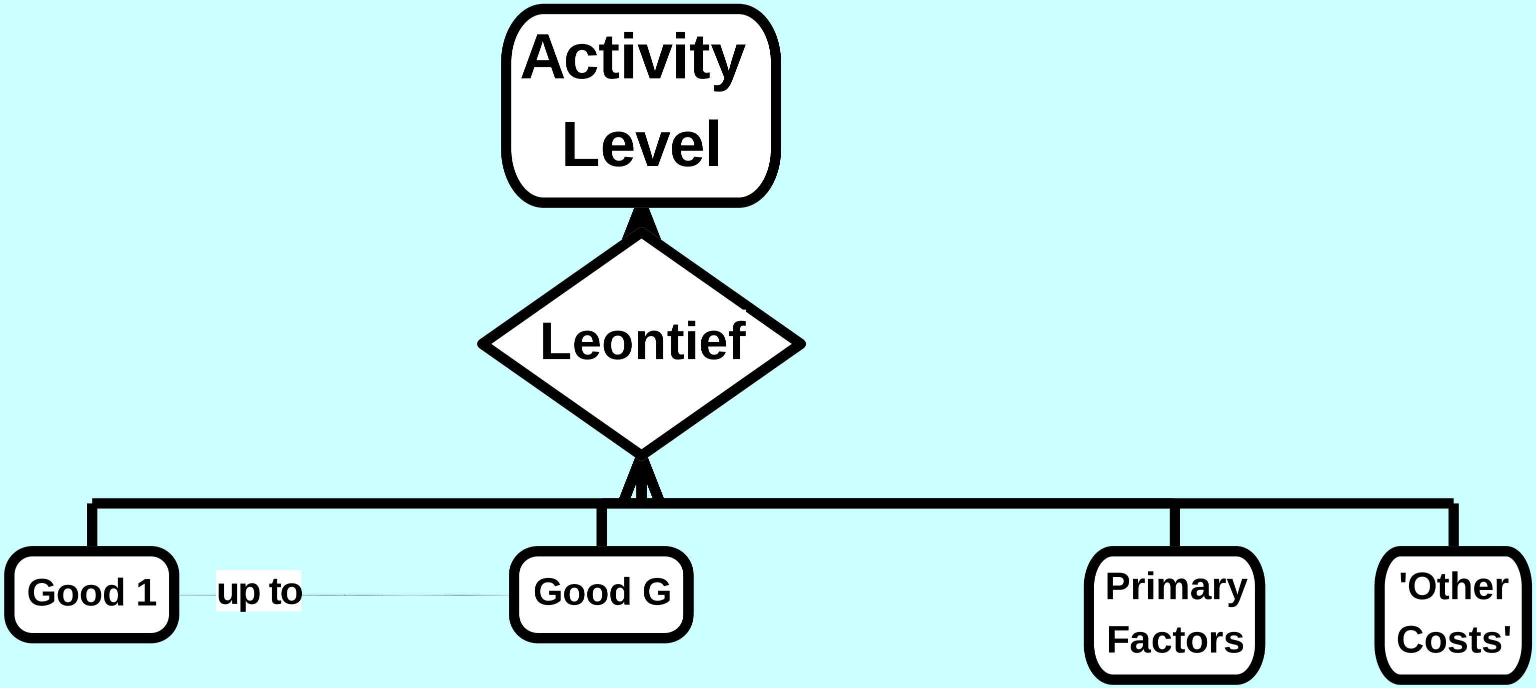
Excerpt 9: Intermediate Cost Index

```
Variable (all,i,IND)  p1mat(i)     # Intermediate cost price index #;
Coefficient  (all,i,IND) V1MAT(i)
                                   # Total intermediate cost for industry i #;
Formula
 (all,i,IND)   V1MAT(i)     = sum{c,COM, V1PUR_S(c,i)};

Equation E_p1mat  # Intermediate cost price index #
 (all,i,IND)
  [TINY+V1MAT(i)]*p1mat(i) =
                  sum{c,COM, sum{s,SRC, V1PUR(c,s,i)*p1(c,s,i)}};
```

Optional, could be useful for understanding results

Also p1var = average all input prices EXCEPT capital and land

Excerpt 10: Top nest of industry inputs



X1TOT(i) = MIN(All,c,COM: X1_S(c,i)/[A1_S(c,s,i)*A1TOT(i)],
X1PRIM(i)/[A1PRIM(i)*A1TOT(i)],
X1OCT(i)/[A1OCT(i)*A1TOT(i)])

Excerpt 10: Top nest of industry inputs

```
Variable
 (all,i,IND)  x1tot(i)     # Activity level or value-added #;
 (all,i,IND)  a1prim(i)  # All factor augmenting technical change #;
 (all,i,IND)  a1tot(i)     # All input augmenting technical change #;
 (all,i,IND)  p1tot(i)     # Average input/output price #;
 (all,i,IND)  a1oct(i)    # "Other cost" ticket augmenting techncal change#;
 (all,c,COM)(all,i,IND)
                  a1_s(c,i)  #Tech change, int'mdiate imp/dom composite#;

Equation E_x1_s  # Demands for commodity composites #
 (all,c,COM)(all,i,IND)  x1_s(c,i) - [a1_s(c,i) + a1tot(i)] = x1tot(i);

Equation E_x1prim  # Demands for primary factor composite #
 (all,i,IND)  x1prim(i) - [a1prim(i) + a1tot(i)] = x1tot(i);

Equation E_x1oct  # Demands for other cost tickets #
 (all,i,IND)  x1oct(i) - [a1oct(i) + a1tot(i)] = x1tot(i);
```

Excerpt 11a: Total Cost and Production Tax

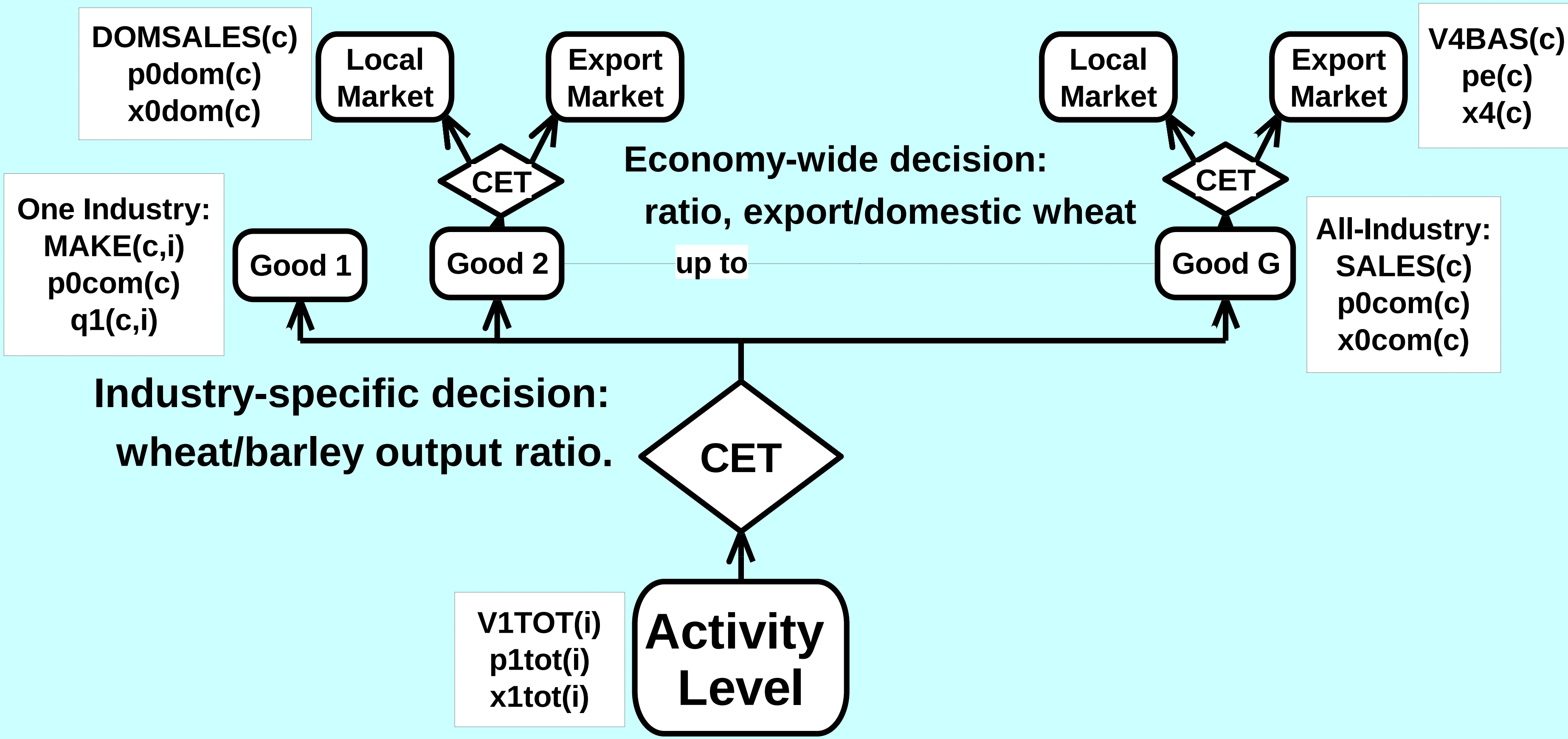
```
Coefficient
 (all,i,IND)  V1CST(i)    # Total cost of industry i #;
 (all,i,IND)  V1TOT(i)    # Total industry cost plus tax #;
 (all,i,IND) PTXRATE(i)   # Rate of production tax #;
Formula
 (all,i,IND)  V1CST(i)    = V1PRIM(i) + V1OCT(i) + V1MAT(i);
 (all,i,IND)  V1TOT(i)    = V1CST(i) + V1PTX(i);
 (all,i,IND)  PTXRATE(i)  = V1PTX(i)/V1CST(i); !  VAT: V1PTX/V1PRIM   !
Write PTXRATE to file SUMMARY header "PTXR";

Variable
 (change)(all,i,IND) delV1CST(i) # Change in ex-tax cost of production #;
 (change)(all,i,IND) delV1TOT(i) # Change in tax-inc cost of production #;
 (change)(all,i,IND) delPTXRATE(i) # Change in rate of production tax #;
```

Excerpt 11b: Total Cost and Production Tax

```
Equation
 E_delV1CST  (all,i,IND) delV1CST(i) = delV1PRIM(i) +
   0.01*sum{c,COM,sum{s,SRC, V1PUR(c,s,i)*[p1(c,s,i) + x1(c,s,i)]}}
                  +  0.01*V1OCT(i)*[p1oct(i)  + x1oct(i)];

 E_delV1PTX  (all,i,IND) delV1PTX(i) =
                    PTXRATE(i)*delV1CST(i)  + V1CST(i) * delPTXRATE(i);
 ! VAT alternative:
      PTXRATE(i)*delV1PRIM(i) + V1PRIM(i)* delPTXRATE(i); !

 E_delV1TOT  (all,i,IND) delV1TOT(i) = delV1CST(i) + delV1PTX(i);

 E_p1tot     (all,i,IND) V1TOT(i)*[p1tot(i) + x1tot(i)] = 100*delV1TOT(i);
```

Progress so far . . .

Excerpt 12: Industry Output mix



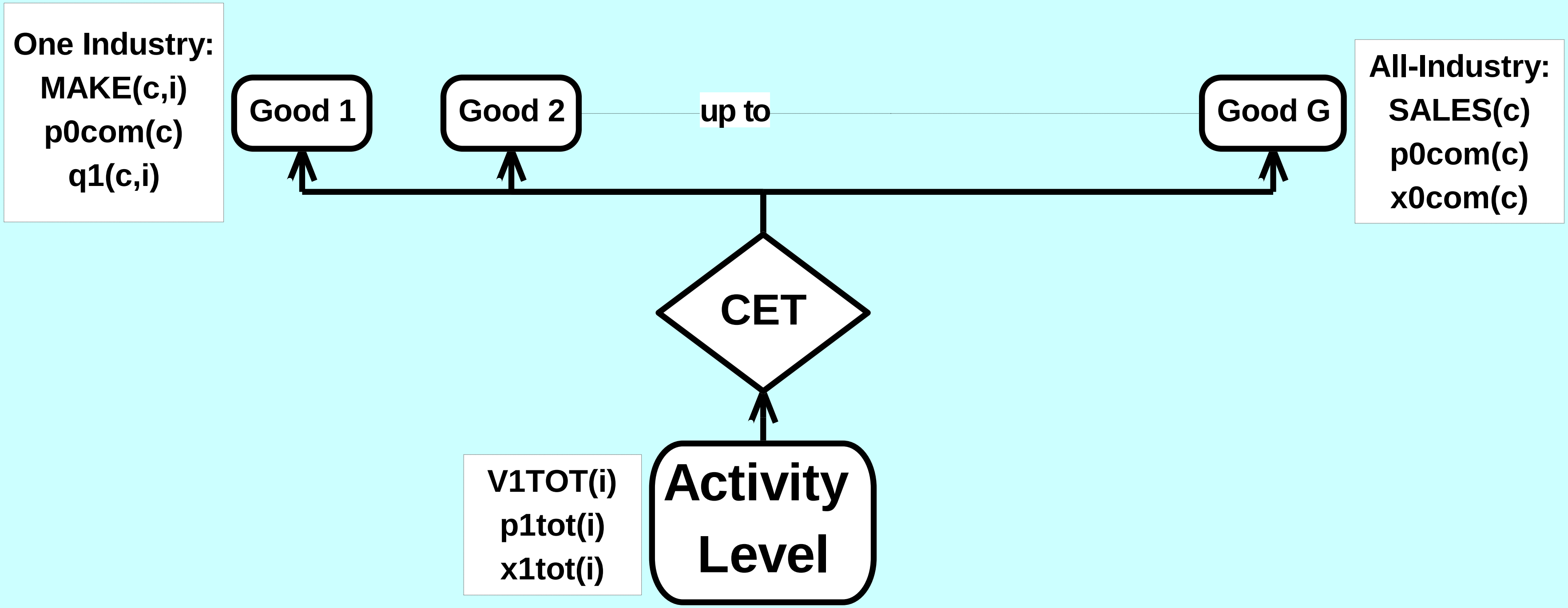
Export/domestic ratio for wheat is same, whichever industry made it.

In practice, often not so complex:

most industries make just one good

export/local CET usually not active

Excerpt 12: Multiproduction Commodity Mix



Industry 7 might produce Commodities 6, 7, and 8.

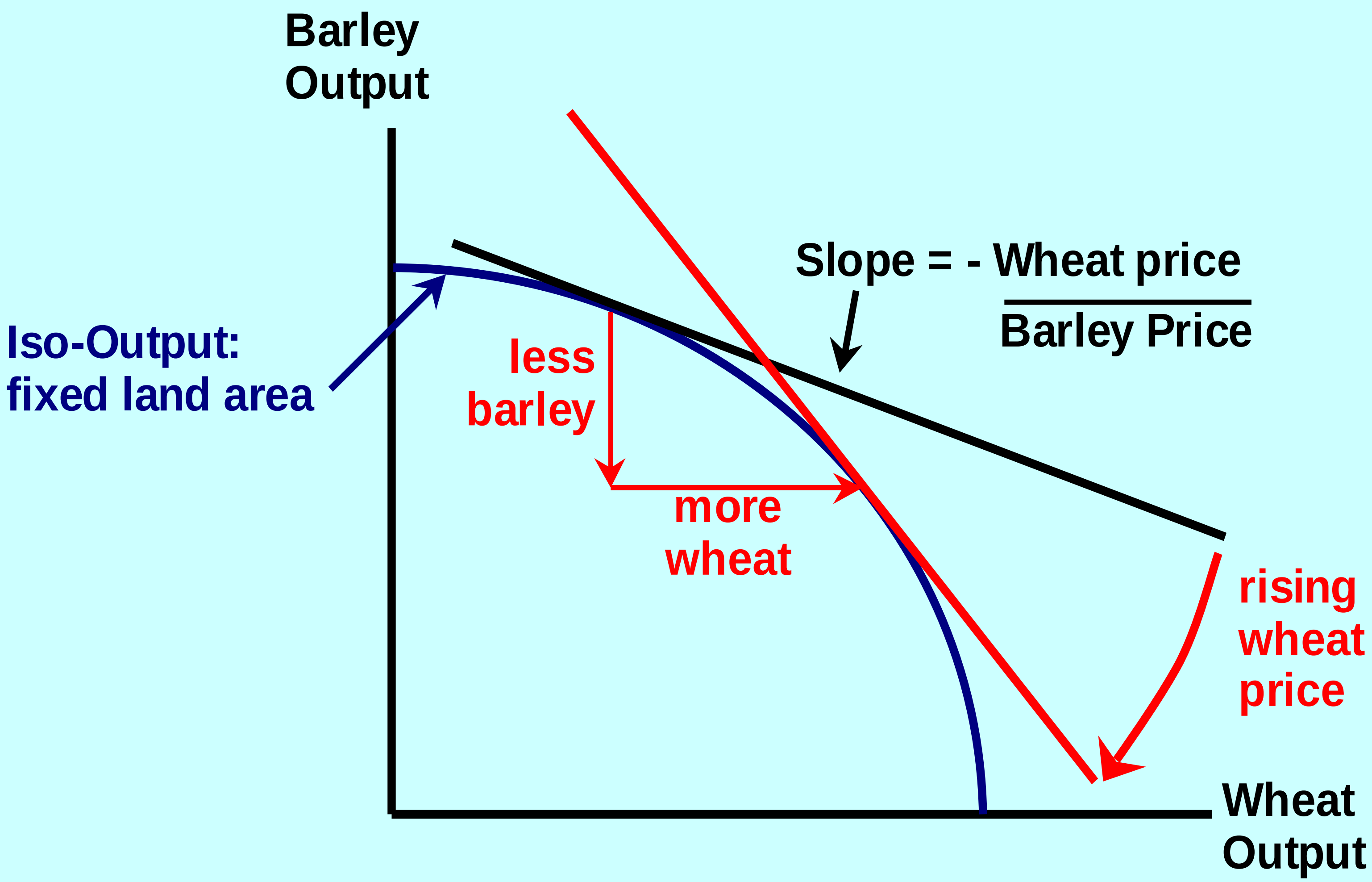
Commodity 3 might be produced by industries 3 and 9.

MAKE(COM,IND) shows which industry produces what.

Every industry that produces wheat get the same wheat price.

As wheat price rises, industries make more wheat and less barley

Excerpt 12: CET transformation frontier



As wheat price rises, industry makes more wheat and less barley.

Algebra same as CES, but substitution elasticity has opposite sign

Australian invention: Powell/Gruen

Do we need Multiproduction?

Competing technologies for producing one commodity:

oil-burning and nuclear plants both make electricity (Taiwan)

zonal agriculture: intensive or extensive beef-production (Australia)

Alternative outputs for a single industry:

Milk/Cattle/Pigs making milk, butter, pork and beef

Supplied MAKE may have many small off-diagonal elements:

IO tables: commodity-industry

Establishment definition:

a shoe factory is one that makes MAINLY shoes, but maybe belts too.

Commodity supplies vector not quite equal to industry output vector,

but MAKE row sums = commodity supplies vector,

and MAKE col sums = industry output vector.

Don't want to adjust data so that MAKE is diagonal,

ie, form commodity-commodity or industry-industry IO table.

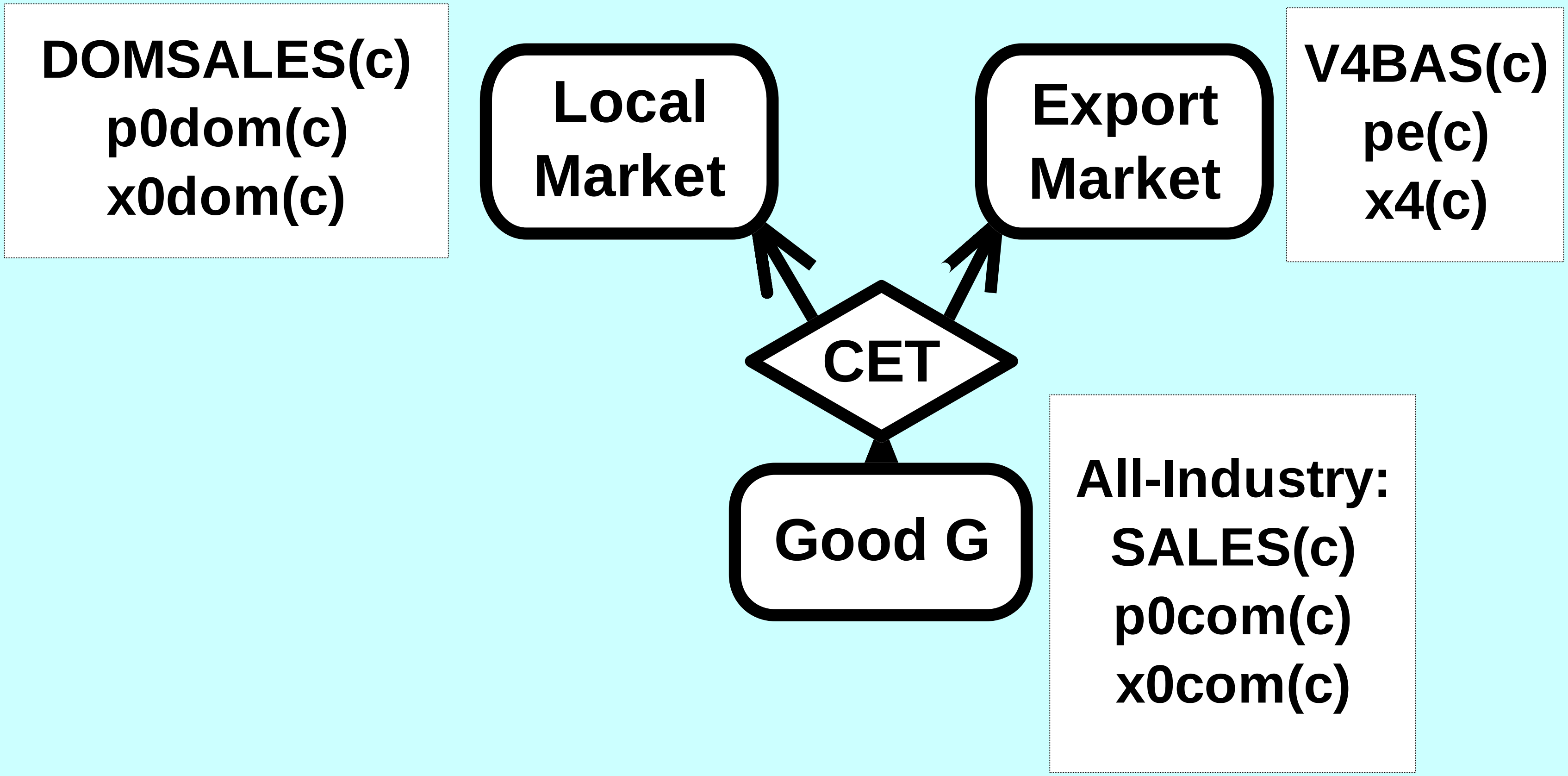
Excerpt 12a: Industry Output mix

```
Coefficient (all,c,COM)(all,i,IND) MAKE(c,i) # Multiproduction matrix #;
Variable (all,c,COM)(all,i,IND)  q1(c,i)   # Output by com and ind #;
 (all,c,COM)  p0com(c)        # Output price of locally-produced com #;

Read MAKE from file BASEDATA header "MAKE";
Update (all,c,COM)(all,i,IND) MAKE(c,i)= p0com(c)*q1(c,i);

Variable
 (all,c,COM)  x0com(c)        # Output of commodities #;

Coefficient (all,i,IND) SIGMA1OUT(i) # CET transformation elasticities #;
Read SIGMA1OUT from file BASEDATA header "SCET";
```

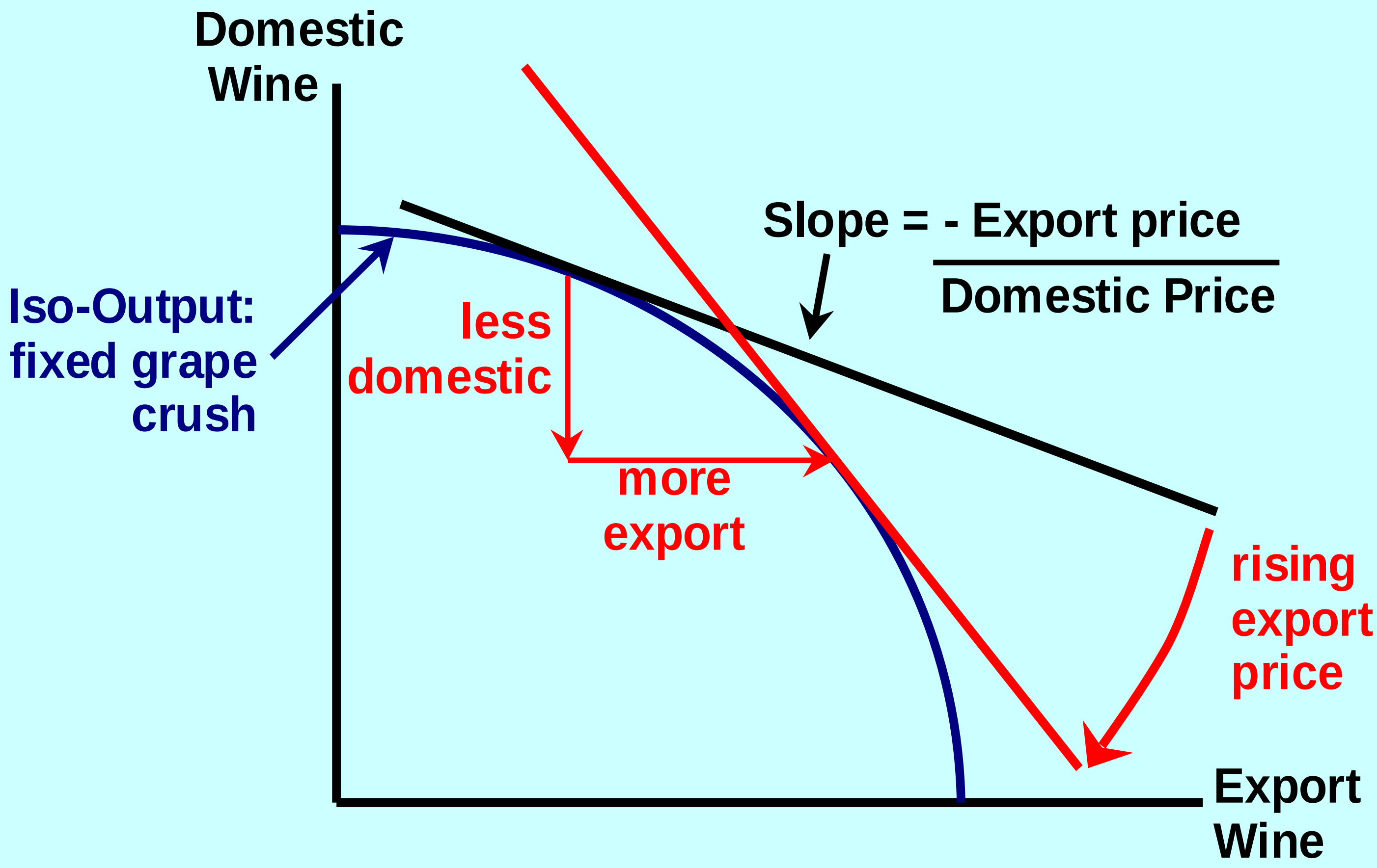
Excerpt 12b: Industry Output mix

```
Equation E_q1  # Supplies of commodities by industries #
 (all,c,COM)(all,i,IND)
  q1(c,i) = x1tot(i) + SIGMA1OUT(i)*[p0com(c) - p1tot(i)];
Coefficient
 (all,i,IND) MAKE_C(i) # All production by industry i #;
 (all,c,COM) MAKE_I(c) # Total production of commodities #;
Formula
 (all,i,IND)    MAKE_C(i)  = sum{c,COM, MAKE(c,i)};
 (all,c,COM) MAKE_I(c)  = sum{i,IND,    MAKE(c,i)};

Equation E_x1tot # Average price received by industries #
 (all,i,IND) MAKE_C(i)*p1tot(i) = sum{c,COM,
    MAKE(c,i)*p0com(c)};
Equation E_x0com # Total output of commodities #
 (all,c,COM) MAKE_I(c)*x0com(c) = sum{i,IND, MAKE(c,i)*q1(c,i)};
```

Excerpt 13: Local/Export Mix



Excerpt 13: CET Export/Domestic mix



As export price rises, industry diverts production towards exports.

Not in ORANI; favoured by Americans; probably wrong

Why do we need Local/Export CET?

Over-specialization: the longrun flip-flop problem

all factors mobile between industries

-- very flat supply curves

Elastic or flat export demand schedules

Australia producing only chocolate

fixed by CET

Americans think long-run
Australians think short-run

Alternatives:

Industry-specific permanently fixed factors (ORANI)

Agricultural Land

Fish or Ore Stocks

-- lead to upwardly sloping supply curves

good for primary products

Less elastic export demand schedules (manufacturing, services)

History or ABARE forecasts: local and export prices may diverge

fixed by CET

Excerpt 13: Local/Export Mix

p0dom	x0dom	price and quantity for local market
pe	x4	price and quantity for export market
p0com	x0com	average price and total quantity

X0COM = CET(X0DOM,X4)

usual 3 nest equations

$$x0dom = x0com + \sigma(p0dom - p0com)$$

subtract

$$x4 = x0com + \sigma(p4 - p0com)$$

$$p0com = S_{local}p0dom + S_{export}p4$$

implying

alternate 3 nest equations

$$x0com = S_{local}x0dom + S_{export}x4$$

and

$$x0dom - x4 = \sigma(p0dom - p4)$$

$$\tau = 1/\sigma$$

$$\tau(x0dom - x4) = p0dom - p4$$

Switching off the Local/Export CET

p0dom	x0dom	price and quantity for local market
pe	x4	price and quantity for export market
p0com	x0com	average price and total quantity

Set τ to zero

$\tau = 1/\sigma = 0$ ie $\sigma = \infty$ (perfect substitutes)

$\tau(x0dom - x4) = 0 = p0dom - p4$

so $p0dom = p4$

$p0com = S_{local}p0dom + S_{export}p4 = p0dom = p4$

$x0com = S_{local}x0dom + S_{export}x4$

Excerpt 13: Local/Export Mix

```
Variable  (all,c,COM)  x0dom(c)  # Output of commodities for local market #;
Coefficient
 (all, c,COM) EXPSHR(c) # Share going to exports #;
 (all, c,COM) TAU(c)    # 1/Elast. of transformation, exportable/locally used #;
Zerodivide default 0.5;
Formula
 (all,c,COM) EXPSHR(c) = V4BAS(c)/MAKE_I(c);
 (all,c,COM) TAU(c) = 0.0; ! if zero, p0dom = pe, and CET is nullified !
Zerodivide off;
Equation E_x0dom  # Supply of commodities to export market #
 (all,c,COM) TAU(c)*[x0dom(c) - x4(c)] = p0dom(c) - pe(c);
Equation E_pe     # Supply of commodities to domestic market #
 (all,c,COM) x0com(c) = [1.0-EXPSHR(c)]*x0dom(c) + EXPSHR(c)*x4(c);
Equation E_p0com  # Zero pure profits in transformation #
 (all,c,COM) p0com(c) = [1.0-EXPSHR(c)]*p0dom(c) + EXPSHR(c)*pe(c);
```

Excerpt 13: Local/Export Mix

CET is joint by-products: imagine τ is large (fixed proportions):

Australian pork products: meat (export) sausages(domestic)

rise in foreign demand for meat floods domestic market with sausages

so export price rises , while domestic price falls.

Australian fisheries: prawns, lobster(export) southern fish(domestic)

rise in foreign demand for lobster domestic market with fish ???

so export price rises , while domestic price falls.

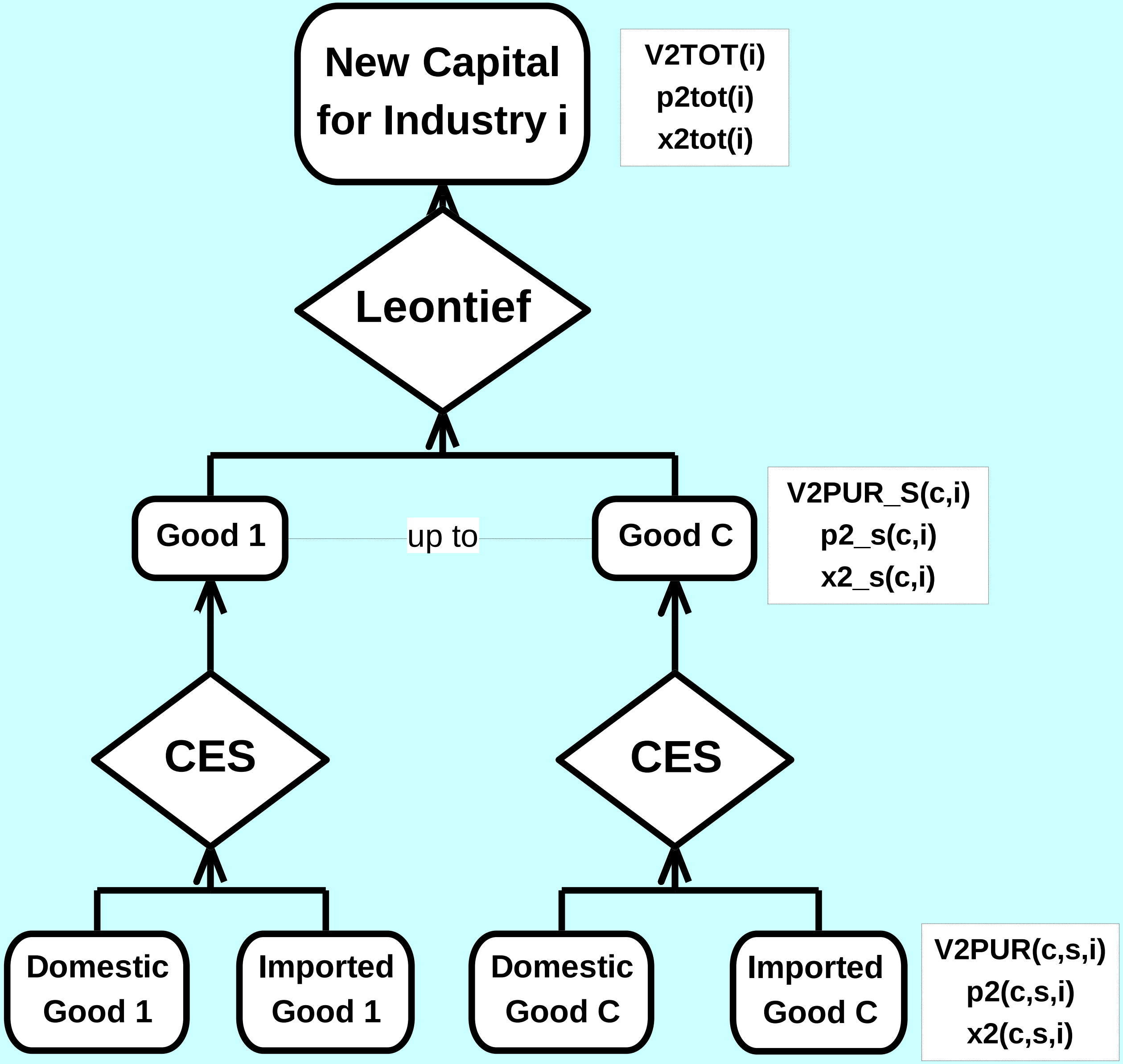
A case for disaggregation

Progress so far . . .

Introduction
Database structure
Solution method
TABLO language
Production: input decisions
Production: output decisions
→ **Investment: input decisions**
Household demands
Export demands
Government demands

Inventory demands
Margin demands
Market clearing
Price equations
Aggregates and indices
Investment allocation
Labour market
Decompositions
Closure
Regional extension

Excerpt 14: Composition of Investment



Excerpt 14a: Composition of Investment

```
Variable
 (all,c,COM)(all,i,IND)  x2_s(c,i)  # Investment use of imp/dom composite #;
 (all,c,COM)(all,i,IND)  p2_s(c,i)  # Price, investment imp/dom composite #;
 (all,c,COM)(all,s,SRC)(all,i,IND)  a2(c,s,i) # Investment basic tech change #;

Coefficient (all,c,COM) SIGMA2(c) # Armington elasticities: investment #;
Read SIGMA2 from file BASEDATA header "2ARM";
Coefficient ! Source Shares in Flows at Purchaser's prices !
 (all,c,COM)(all,i,IND)     V2PUR_S(c,i) # Dom+imp investment purch. value #;
 (all,c,COM)(all,s,SRC)(all,i,IND)  S2(c,s,i) # Investment source shares #;
Zerodivide default 0.5;
Formula
  (all,c,COM)(all,i,IND)     V2PUR_S(c,i)  = sum{s,SRC, V2PUR(c,s,i)};
  (all,c,COM)(all,s,SRC)(all,i,IND)  S2(c,s,i) = V2PUR(c,s,i) / V2PUR_S(c,i);
Zerodivide off;
```

Excerpt 14b: Composition of Investment

```
Equation E_x2  # Source-specific commodity demands #
(all,c,COM)(all,s,SRC)(all,i,IND)
x2(c,s,i)-a2(c,s,i) - x2_s(c,i)
                  = - SIGMA2(c)*[p2(c,s,i)+a2(c,s,i) - p2_s(c,i)];

Equation E_p2_s #Effective price of commodity composite #
(all,c,COM)(all,i,IND)
p2_s(c,i) = sum{s,SRC, S2(c,s,i)*[p2(c,s,i)+a2(c,s,i)]};
```

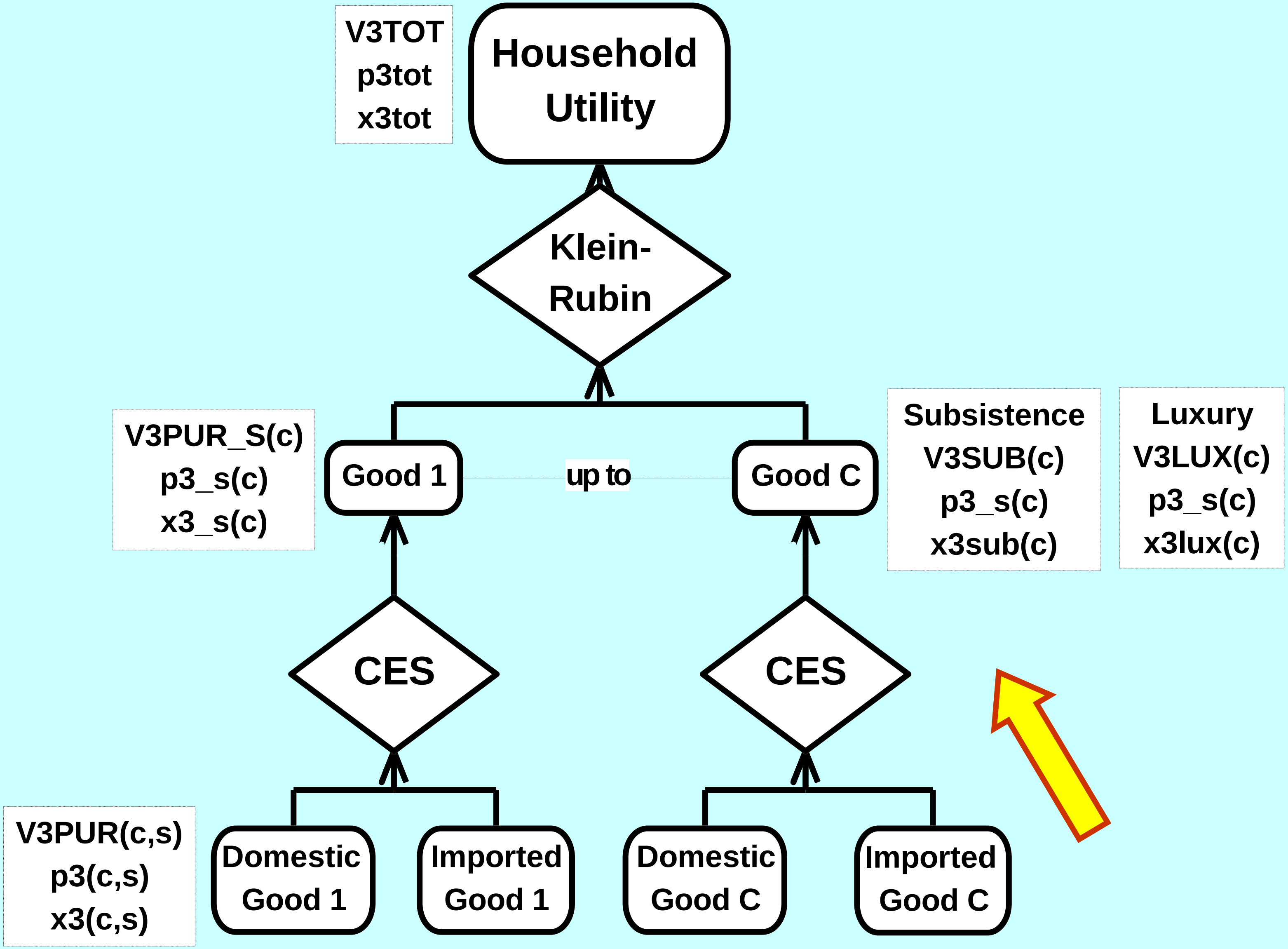
Excerpt 14c: Composition of Investment

```
! Investment top nest !
!$  X2TOT(i) = MIN( All,c,COM: X2_S(c,i)/[A2_S(c,i)*A2TOT(i)] ) !

Variable
 (all,i,IND)  a2tot(i)       # Neutral technical change - investment #;
 (all,i,IND)  p2tot(i)       # Cost of unit of capital #;
 (all,i,IND)  x2tot(i)       # Investment by using industry #;
 (all,c,COM)(all,i,IND) a2_s(c,i) # Tech change, investment imp/dom
    composite #;
Coefficient (all,i,IND) V2TOT(i)  # Total capital created for industry i #;
Formula     (all,i,IND) V2TOT(i)  = sum{c,COM, V2PUR_S(c,i)};
Equation
 E_x2_s  (all,c,COM)(all,i,IND)  x2_s(c,i) - [a2_s(c,i) + a2tot(i)] = x2tot(i);
 E_p2tot (all,i,IND)    V2TOT(i)*p2tot(i)
      = sum{c,COM, V2PUR_S(c,i)*[p2_s(c,i) +a2_s(c,i) +a2tot(i)]};
```

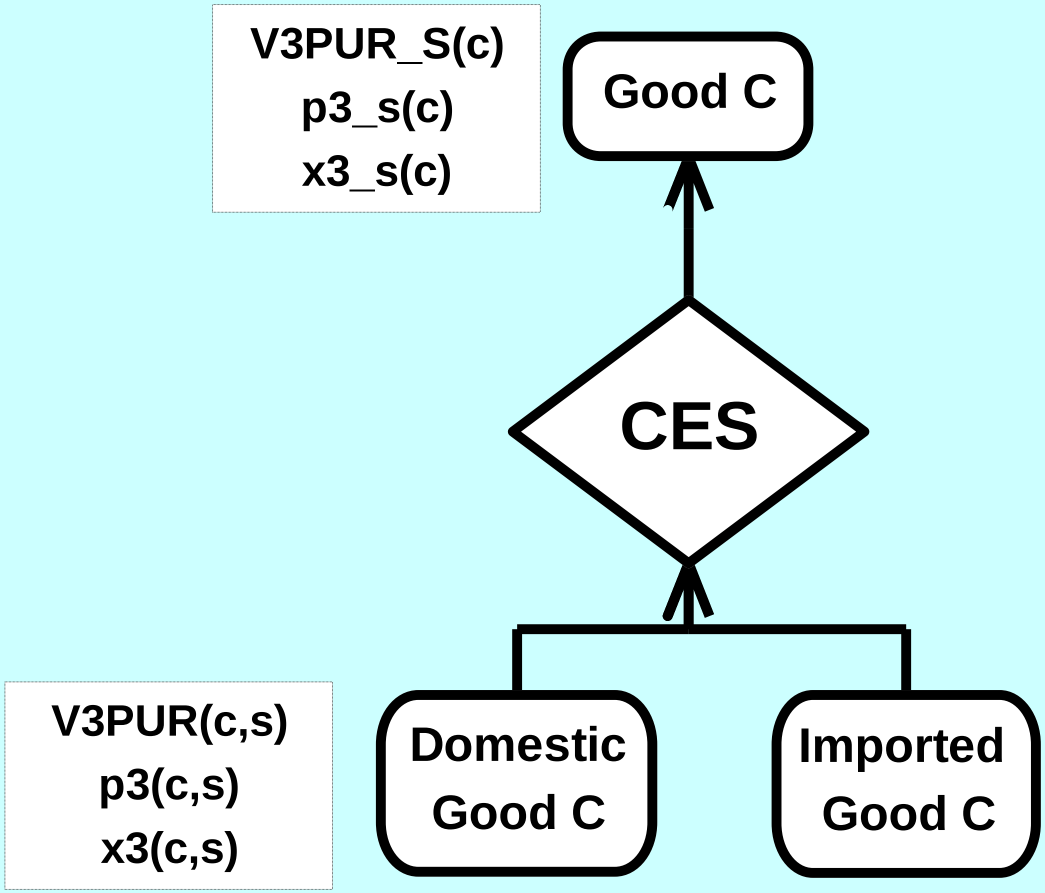
Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- → **Household demands**
- Export demands
- Government demands
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- Aggregates and indices
- Investment allocation
- Labour market
- Decompositions
- Closure
- Regional extension

Household Demands



Household imp/dom sourcing



Excerpt 15a: household imp/dom sourcing

```
Variable
(all,c,COM)(all,s,SRC)  a3(c,s)   # Household basic taste change #;
(all,c,COM)             x3_s(c)   # Household use of imp/dom composite #;
(all,c,COM)             p3_s(c)   # Price, household imp/dom composite #;
Coefficient (all,c,COM) SIGMA3(c) # Armington elasticity: households #;
Read SIGMA3 from file BASEDATA header "3ARM";

Coefficient ! Source Shares in Flows at Purchaser's prices !
 (all,c,COM)           V3PUR_S(c) # Dom+imp households purch. value #;
 (all,c,COM)(all,s,SRC)   S3(c,s) # Household source shares #;
Zerodivide default 0.5;
Formula
 (all,c,COM)           V3PUR_S(c) = sum{s,SRC, V3PUR(c,s)};
 (all,c,COM)(all,s,SRC)   S3(c,s) = V3PUR(c,s)   / V3PUR_S(c);
Zerodivide off;
```

Excerpt 15b: household imp/dom sourcing

```
Equation E_x3  # Source-specific commodity demands #
(all,c,COM)(all,s,SRC)
x3(c,s)-a3(c,s) = x3_s(c)
                    - SIGMA3(c)*[ p3(c,s)+a3(c,s) - p3_s(c) ];

Equation E_p3_s  # Effective price of commodity composite #
(all,c,COM) p3_s(c) = sum{s,SRC, S3(c,s)*[p3(c,s)+a3(c,s)]};
```

Numerical Example of CES demands

feel for numbers

$p = S_d p_d + S_m p_m$ average price of dom and imp Food

$x_d = x - \sigma(p_d - p)$ demand for domestic Food

$x_m = x - \sigma(p_m - p)$ demand for imported Food

Let p_m=-10%, $x=p_d=0$

Let S_m=0.3 and σ =2. This gives:

$p = -0.3*10 = -3$

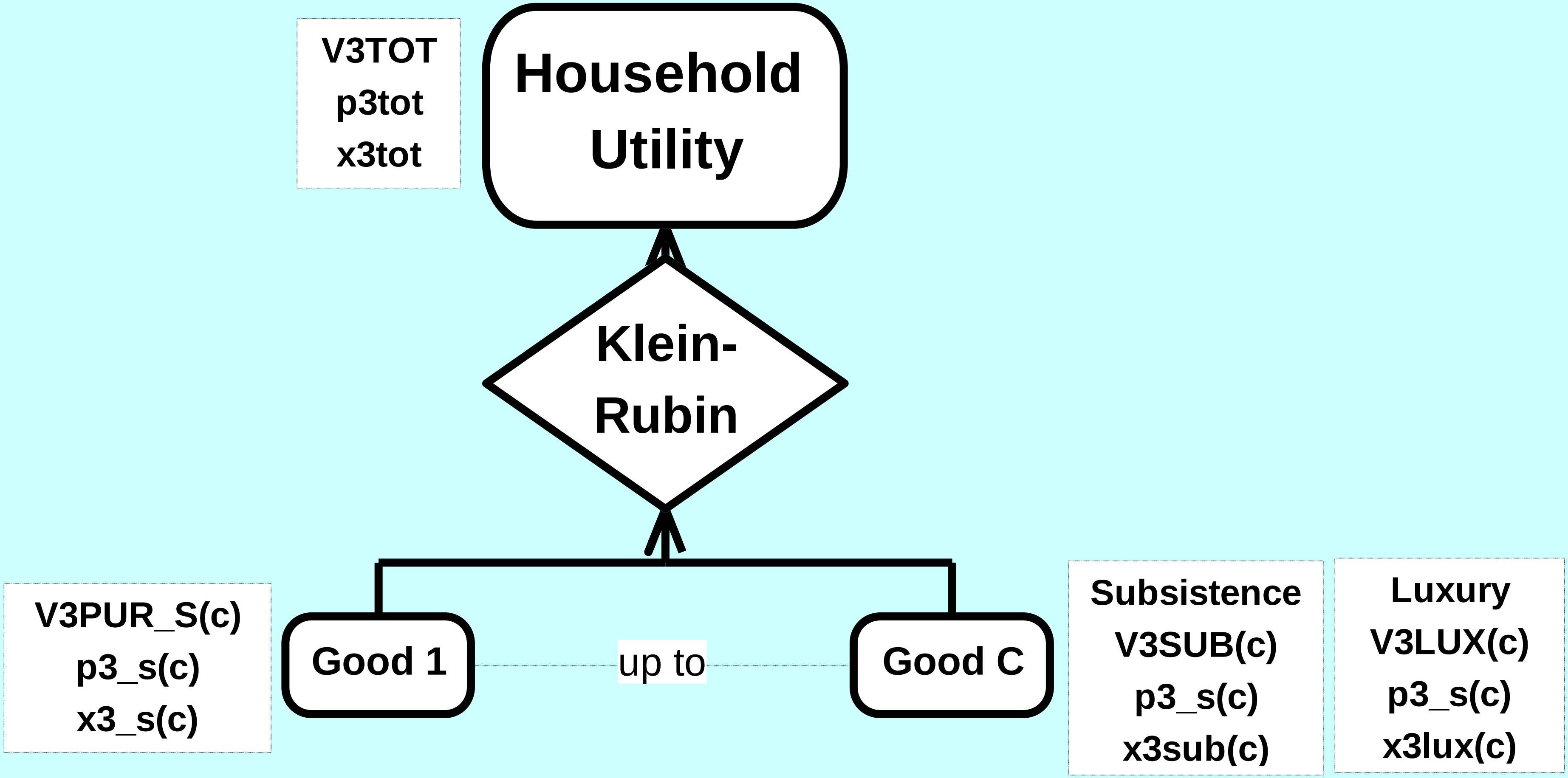
$x_d = - 2(- -3) = -6$

$x_m = -2(-10 - - 3) = 14$

Cheaper imports cause 14% increase in import volumes and 6% fall in domestic demand.

Effect on domestic sales is proportional to both S_m and σ.

Top Nest of Household Demands



Klein-Rubin:
a non-homothetic utility function

Homothetic means:

budget shares depend only on prices, not incomes

eg: CES, Cobb-Douglas

Non-homothetic means:

rising income causes budget shares to change even with price ratios fixed.

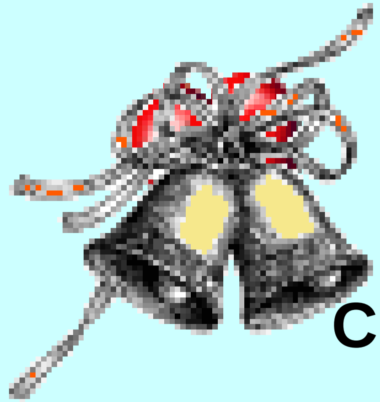
Non-unitary expenditure elasticities:

1% rise in total expenditure might cause food expenditure to rise by 1/2%; air travel expenditure to rise by 2%.

See Green Book for algebraic derivation (complex). Explained here by a metaphor.

Two Happy Consumers

Mr Klein



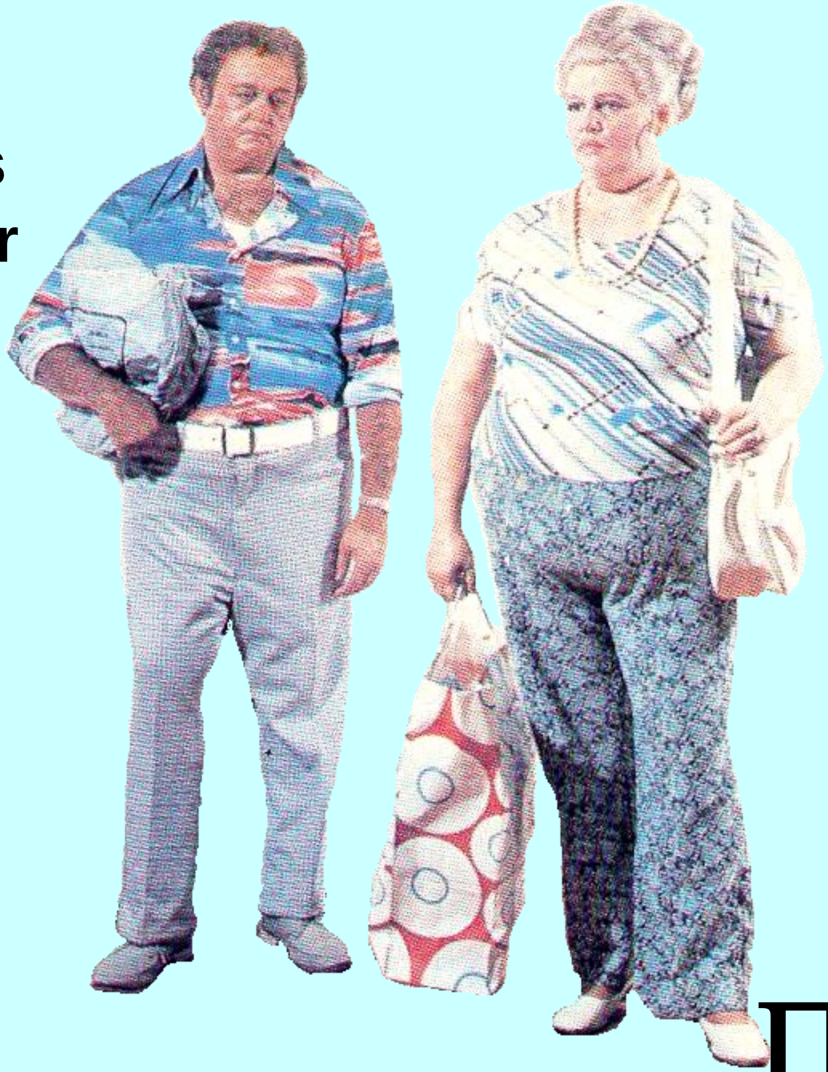
Miss Rubin

**Cobb-Douglas:
constant
budget
shares:
30% clothes
70% food**

**weekly:
300 cigarettes
30 bottles beer**

The Klein-Rubin Household

First buy:
300 cigarettes
30 bottles beer
subsistence (constant)
X3SUB(c)



Allocate remaining money:
clothes 30%
food 70%
luxury (goes with income)
X3LUX(c)

Utility =

$$\prod \{X3LUX(c)\}^{S3LUX(c)}$$

Total consumption good c

X3_S(c) = X3SUB(c) + X3LUX(c)

Also called Linear Expenditure System

Total expenditure = subsistence cost + luxury expenditure

supernumerary

P3_S(c) *X3_S(c) = P3_S(c) *X3SUB(c) + S3LUX(c) *V3LUX_C

P3_S(c) *X3_S(c) = P3_S(c) *X3SUB(c)

+ S3LUX(c) *[V3TOT - Σ {P3_S(c) *X3SUB(c)}]

all subsistence costs

Expenditure on each good is a linear function of prices and income

How many parameters -degree of flexibility

No of parameters =
extra numbers needed to specify percent change form
IF EXPENDITURE VALUES ARE ALREADY KNOWN

Example, CES=1:
with input values known, 1 number, σ, is enough.

Example, CobbDouglas=0:
with input values known, we know all.

In levels, more parameters are needed.

Example, Leontief=0:
with input values known, we know all.

How many parameters is Klein-Rubin/LES ?

We need to divide expenditure on each good
into subsistence and luxury parts.

(all,c,COM) B3LUX(c) # Ratio,supernumerary/total expenditure#;

One B3LUX parameter for each commodity.

These "parameters" change !

not in doc

Deriving B3LUX from literature estimates

Normally expressed as:

EPS = Expenditure elasticities for each good

= marginal/average budget shares

$$= \frac{\text{(share this good in luxury spending)}}{\text{(share this good in all spending)}}$$

and 1969, Tinbergen

Frisch "parameter" = - 1.82

$$= - \frac{\text{(total spending)}}{\text{(total luxury spending)}}$$

= 1 + C numbers ! **but average of EPS = 1**

S3_S(c) = V3PUR_S(c)/V3TOT **average shares**

B3LUX(c) = -EPS(c)/FRISCH **share of luxury**

S3LUX(c) = EPS(c)*S3_S(c) **marginal budget shares**

Excerpt 16a: household demands

```
Variable
     p3tot # Consumer price index #;
     x3tot # Real household consumption #;
   w3lux # Total nominal supernumerary household expenditure #;
    w3tot # Nominal total household consumption #;
          q # Number of households #;
   utility # Utility per household #;
 (all,c,COM) x3lux(c) # Household - supernumerary demands #;
 (all,c,COM) x3sub(c) # Household - subsistence demands #;

(all,c,COM) a3lux(c) # Taste change, supernumerary demands #;
(all,c,COM) a3sub(c) # Taste change, subsistence demands #;
(all,c,COM)  a3_s(c) # Taste change, h'hold imp/dom composite #;
```

Excerpt 16b: household demands

```
Coefficient
                  V3TOT   # Total purchases by households #;
                 FRISCH   # Frisch LES 'parameter'= - (total/luxury)#;
 (all,c,COM)   EPS(c)    # Household expenditure elasticities #;
 (all,c,COM)  S3_S(c)   # Household average budget shares #;
 (all,c,COM) B3LUX(c) # Ratio,supernumerary/total expenditure#;
 (all,c,COM) S3LUX(c) # Marginal household budget shares #;
Read   FRISCH from file BASEDATA header "P021";
            EPS from file BASEDATA header "XPEL";
Update
(change)            FRISCH = FRISCH*[w3tot - w3lux]/100.0;
(change)(all,c,COM)
          EPS(c) = EPS(c)*[x3lux(c)-x3_s(c)+w3tot-w3lux]/100.0;
```

Excerpt 16c: household demands

```
Formula
                V3TOT = sum{c,COM, V3PUR_S(c)};
 (all,c,COM)  S3_S(c) = V3PUR_S(c)/V3TOT;
 (all,c,COM) B3LUX(c) = -EPS(c)/FRISCH;
 (all,c,COM) S3LUX(c) = EPS(c)*S3_S(c);
Write  S3LUX   to file SUMMARY header "LSHR";
       S3_S    to file SUMMARY header "CSHR";
```

Excerpt 16d: household demands

Equation

E_x3sub # Subsistence demand for composite commodities #
(all,c,COM) x3sub(c) = q + a3sub(c);

E_x3lux # Luxury demand for composite commodities #
(all,c,COM) x3lux(c) + p3_s(c) = w3lux + a3lux(c);

E_x3_s # Total household demand for composite commodities #
(all,c,COM) x3_s(c) = B3LUX(c)*x3lux(c)
+ [1-B3LUX(c)]*x3sub(c);

E_utility # Change in utility disregarding taste change terms #
utility + q = sum{c,COM, S3LUX(c)*x3lux(c)};

Excerpt 16e: household demands

```
E_a3lux # Default setting for luxury taste shifter #
(all,c,COM) a3lux(c) = a3sub(c) - sum{k,COM,
  S3LUX(k)*a3sub(k)};

E_a3sub # Default setting for subsistence taste shifter #
(all,c,COM) a3sub(c) = a3_s(c) - sum{k,COM, S3_S(k)*a3_s(k)};

E_x3tot # Real consumption #
V3TOT*x3tot = sum{c,COM, sum{s,SRC, V3PUR(c,s)*x3(c,s)}};

E_p3tot # Consumer price index #
V3TOT*p3tot = sum{c,COM, sum{s,SRC, V3PUR(c,s)*p3(c,s)}};

E_w3tot # Household budget constraint: determines w3lux #
w3tot = x3tot + p3tot;
```

Quiz Question

Fact: with σ = 1, CES is same as Cobb-Douglas.

Question: With all expenditure elasticities = 1, is Klein-Rubin same as Cobb-Douglas ?

Answer: No. Would be Cobb-Douglas if Frisch parameter = -1 [totally luxury]. Own-price demand elasticity for Cobb-Douglas = -1; average own-price demand elasticity for Klein-Rubin is share of luxury in total spending (maybe 0.5). Tendency towards inelastic demand.

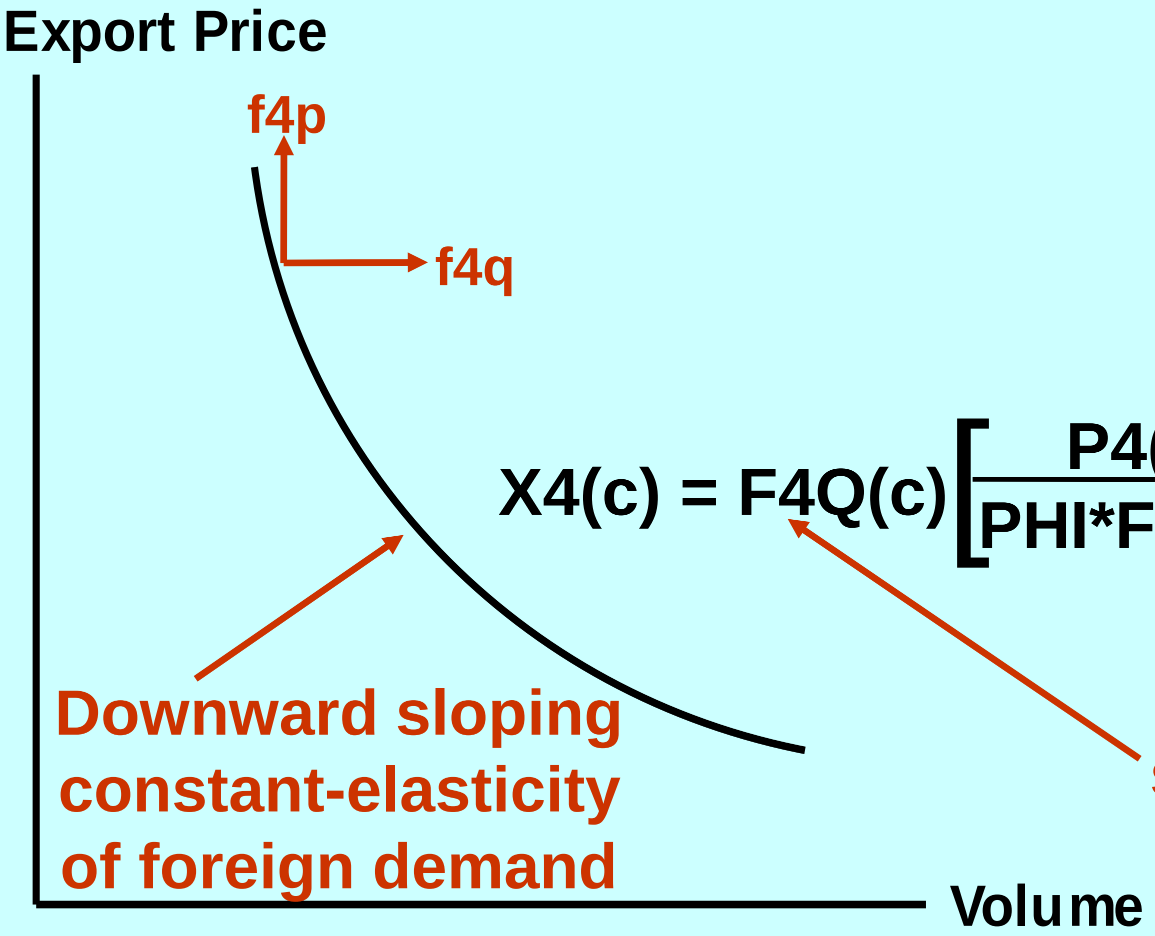
Stone-Geary = another name for Klein-Rubin

Progress so far . . .

Introduction
Database structure
Solution method
TABLO language
Production: input decisions
Production: output decisions
Investment: input decisions
Household demands
→ **Export demands**
Government demands

Inventory demands
Margin demands
Market clearing
Price equations
Aggregates and indices
Investment allocation
Labour market
Decompositions
Closure
Regional extension

Excerpt 17: Individual Export demands



$$X4(c) = F4Q(c)\left[\frac{P4(c)}{PHI*F4P(c)}\right]^{EXP\_ELAST(c)}$$

shift variables

In original ORANI, only applied to main (primary) export commodities.

The rest (collective exports) are bundled together as an aggregate, with a shared demand curve.

Excerpt 17a: Export demands

Variable phi # Exchange rate, local currency/$world #;
(all,c,COM) f4p(c) # Price (upward) shift in export demands #;
(all,c,COM) f4q(c) # Quantity (right) shift in export demands #;
Coefficient (all,c,COM) EXP_ELAST(c)
Export demand elasticities: typical value -5.0 #;
Read EXP_ELAST from file BASEDATA header "P018";
Equation E_x4A # Individual export demand functions #
(all,c,TRADEXP)

x4(c) - f4q(c) = EXP_ELAST(c)*[p4(c) - phi - f4p(c)];

levels:

$$X4(c) = F4Q(c)\left[\frac{P4(c)}{PHI*F4P(c)}\right]^{EXP\_ELAST(c)}$$

Excerpt 17b: Export demands

```
Set NTRADEXP # Collective Export Commodities #
          = COM - TRADEXP;
Write (Set) NTRADEXP  to file SUMMARY header "NTXP";
Variable
 x4_ntrad    # Quantity, collective export aggregate #;
 f4p_ntrad  # Upward demand shift, collective export aggregate #;
 f4q_ntrad  # Right demand shift, collective export aggregate #;
 p4_ntrad   # Price, collective export aggregate #;

Coefficient V4NTRADEXP # Total collective export earnings #;
Formula     V4NTRADEXP = sum{c,NTRADEXP, V4PUR(c)};
```

Excerpt 17c: Export demands

Equation E_X4B # Collective export demand functions #

(all,c,NTRADEXP) x4(c) - f4q(c) = x4_ntrad; **all move together**

Equation E_p4_ntrad # Average price of collective exports #

[TINY+V4NTRADEXP]*p4_ntrad

= sum{c,NTRADEXP, V4PUR(c)*p4(c)};

Coefficient EXP_ELAST_NT # Collective export demand elast #;

Read EXP_ELAST_NT from file BASEDATA header "EXNT";

Equation E_x4_ntrad # Demand for collective export aggregate #

x4_ntrad - f4q_ntrad = EXP_ELAST_NT*[p4_ntrad - phi - f4p_ntrad];

Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- → **Government demands**
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- Aggregates and indices
- Investment allocation
- Labour market
- Decompositions
- Closure
- Regional extension

Excerpt 18a: Government demands

```
Variable
 f5tot  # Overall shift term for government demands #;
 f5tot2 # Ratio between f5tot and x3tot #;
 (all,c,COM)(all,s,SRC) f5(c,s) # Government demand shift #;
 (change)
 (all,c,COM)(all,s,SRC)  fx6(c,s) # Shifter on stocks rule #;
Equation
 E_x5 # Government demands #
  (all,c,COM)(all,s,SRC)  x5(c,s) = f5(c,s) + f5tot;
 E_f5tot # Overall government demands shift #
   f5tot = x3tot + f5tot2;
```

Cunning use of shift variables

(all,c,COM)(all,s,SRC) x5(c,s) = f5(c,s) + f5tot;

f5tot = x3tot + f5tot2;

Shift variables f5tot and f5tot2
used to switch between two rules:

With f5tot2 exogenous, f5tot endogenous, we get

(all,c,COM)(all,s,SRC) x5(c,s) = f5(c,s) + x3tot + f5tot2;

ie: gov. demands follow real household consumption

with f5tot exogenous, f5tot2 endogenous, we get

(all,c,COM)(all,s,SRC) x5(c,s) = f5(c,s) + f5tot;

ie: gov. demands are exogenous

Progress so far . . .

Introduction
Database structure
Solution method
TABLO language
Production: input decisions
Production: output decisions
Investment: input decisions
Household demands
Export demands
Government demands

→ **Inventory demands**
Margin demands
Market clearing
Price equations
Aggregates and indices
Investment allocation
Labour market
Decompositions
Closure
Regional extension

Excerpt 18b: Inventory demands

Useful to endogenously calculate the change in the volume of goods going to inventory. (Eg. Real homogeneity test)

. . . However we have no theory to explain changes in inventory demands . . .

so we adopt a simple rule: % change inventory demand = % change in domestic production

BUT: Inventory demand can change sign - rate of change variable

x6(c,s) = x(c)

100 . [dX6(c,s) / X6(c,s)] = x(c)

100 . dX6(c,s) = X6(c,s) . x(c)

[100 . P6(c,s)] . dX6(c,s) = [P6(c,s) . X6(c,s)] . x(c) ***E_delx6***

V6BAS

Change in quantity

Excerpt 18b: Inventory demands

Coefficient (all,c,COM)(all,s,SRC)
LEVP0(c,s) # Levels basic prices #;

must specify units for ordinary change in quantities

Formula (initial) (all,c,COM)(all,s,SRC)
LEVP0(c,s) = 1; ! arbitrary setting !

Update (all,c,COM)(all,s,SRC) LEVP0(c,s) = p0(c,s);

change in quantity at "current" prices

Equation
E_delx6 # Stocks follow domestic output #

or exogenous

(all,c,COM)(all,s,SRC)
100*LEVP0(c,s)*delx6(c,s) = V6BAS(c,s)*x0com(c) + fx6(c,s);

Excerpt 18b: Inventory demands

Recall that the update of inventory demands is via a change variable.

. . . this is defined by E_delV6 . . .

E_delV6 # Update formula for stocks #

(all,c,COM)(all,s,SRC)

delV6(c,s) = 0.01*V6BAS(c,s)*p0(c,s)+ LEVP0(c,s)*delx6(c,s);

Derivation of E_delV6

V6(c,s) = P0(c,s) . X6(c,s)

dV6 = dP0 . X6 + P0 . dX6

dV6 = [0.01] . [P0 X6] . [100 dP0 / P0] + P0 . dX6

dV6 = [0.01 . V6] . p0 + [P0] . dX6 ***E_delV6***

Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- Government demands
- Inventory demands
- → **Margin demands**
- Market clearing
- Price equations
- Aggregates and indices
- Investment allocation
- Labour market
- Decompositions
- Closure
- Regional extension

Excerpt 19: Margin demands

Intermediate only - see text for rest

```
Variable
 (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
   a1mar(c,s,i,m)  # Intermediate margin tech change #;
Equation
 E_x1mar  # Margins to producers #
   (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
  x1mar(c,s,i,m) = x1(c,s,i) + a1mar(c,s,i,m);
```

normally exogenous = 0

```
Coefficient (all,c,COM) MARSALES(c) # Total usage,margins purposes #;
Formula     (all,n,NONMAR) MARSALES(n) = 0.0;
 (all,m,MAR)    MARSALES(m) = sum{c,COM, V4MAR(c,m) +
                          sum{s,SRC, V3MAR(c,s,m) + V5MAR(c,s,m) +
                           sum{i,IND, V1MAR(c,s,i,m) + V2MAR(c,s,i,m) }}};
```

Progress so far . . .

Excerpt 20a: Sales Aggregates

```
Set DEST # Sale Categories #
(Interm, Invest, HouseH, Export, GovGE, Stocks, Margins);

Coefficient (all,c,COM)(all,s,SRC)(all,d,DEST)
        SALE(c,s,d) # Sales aggregates #;
Formula
 (all,c,COM)(all,s,SRC) SALE(c,s,"Interm")  = sum{i,IND, V1BAS(c,s,i)};
 (all,c,COM)(all,s,SRC) SALE(c,s,"Invest")  = sum{i,IND, V2BAS(c,s,i)};
 (all,c,COM)(all,s,SRC) SALE(c,s,"HouseH")         = V3BAS(c,s);
 (all,c,COM)            SALE(c,"dom","Export")  = V4BAS(c);
 (all,c,COM)            SALE(c,"imp","Export")   = 0;
 (all,c,COM)(all,s,SRC) SALE(c,s,"GovGE")          = V5BAS(c,s);
 (all,c,COM)(all,s,SRC) SALE(c,s,"Stocks")          = V6BAS(c,s);
 (all,c,COM)            SALE(c,"dom","Margins") = MARSALES(c);
 (all,c,COM)            SALE(c,"imp","Margins") = 0;
Write SALE to file SUMMARY header "SALE";
```

Excerpt 20b: Sales Aggregates

```
Coefficient (all,c,COM) V0IMP(c) # Total basic-value imports, good c #;
Formula     (all,c,COM) V0IMP(c) = sum{d,DEST, SALE(c,"imp",d)};

Coefficient (all,c,COM) SALES(c) # Total sales,domestic commodities#;
Formula     (all,c,COM) SALES(c) = sum{d,DEST, SALE(c,"dom",d)};

Coefficient (all,c,COM) DOMSALES(c) # Total sales to local market #;
Formula     (all,c,COM) DOMSALES(c) = SALES(c) - V4BAS(c);
```

Excerpt 21a: Market clearing

Commodity Supply = Commodity Demand

Commodity Demands: **intermediate, investment, household, export, government, stocks, margins.**

It will prove handy later (*see p. 47 - 49*) to measure now each of these changes in demand as changes in physical quantities valued at current prices.

$$dS = P \,.\, dX$$

$$dS = [X \,.\, P \,/\, 100] \,.\, (dX \,/\, X) \,.\, 100$$

$$dS = [\, 0.01 \,.\, VBAS \,] \,.\, x$$ **standard form**

Excerpt 21a: Market clearing

Standard form

```
Variable (change)
 (all,c,COM)(all,s,SRC)(all,d,DEST)
          delSale(c,s,d) # Sales aggregates #;

Equation
E_delSaleA (all,c,COM)(all,s,SRC) delSale(c,s,"Interm") =
               0.01*sum{i,IND,V1BAS(c,s,i)*x1(c,s,i)};

E_delSaleB (all,c,COM)(all,s,SRC) delSale(c,s,"Invest") =
               0.01*sum{i,IND,V2BAS(c,s,i)*x2(c,s,i)};

E_delSaleC (all,c,COM)(all,s,SRC)
   delSale(c,s,"HouseH")=0.01*V3BAS(c,s)*x3(c,s);
```

Excerpt 21a: Market clearing

Standard form

```
E_delSaleD (all,c,COM)
   delSale(c,"dom","Export")=0.01*V4BAS(c)*x4(c);
```

No imported exports

```
E_delSaleE (all,c,COM)
                    delSale(c,"imp","Export")=   0;
```

Standard form

```
E_delSaleF (all,c,COM)(all,s,SRC)
   delSale(c,s,"GovGE") =0.01*V5BAS(c,s)*x5(c,s);
```

```
E_delSaleG (all,c,COM)(all,s,SRC) delSale(c,s,"Stocks") =
   LEVP0(c,s)*delx6(c,s);
```

Initial form

Excerpt 21b: Market clearing

```
E_delSaleH (all,m,MAR) delSale(m,"dom","Margins")=0.01*
  ! note nesting of sum parentheses !
sum{c,COM, V4MAR(c,m)*x4mar(c,m) + sum{s,SRC,
   V3MAR(c,s,m)*x3mar(c,s,m) + V5MAR(c,s,m)*x5mar(c,s,m)
  + sum{i,IND, V1MAR(c,s,i,m)*x1mar(c,s,i,m) +
   V2MAR(c,s,i,m)*x2mar(c,s,i,m)   }}};
```

Standard form

NONMAR not used as Margin

```
E_delSaleI (all,n,NONMAR)   delSale(n,"dom","Margins") = 0;
```

No imported margins

```
E_delSaleJ (all,c,COM)        delSale(c,"imp","Margins") = 0;
```

Excerpt 21c: Market clearing

Equation E_p0A: Sets supply of each domestic commodity to the local market equal to the sum of local demands . . .

$$X0(i) = \Sigma_{user} X(i,user)$$

$$dX0(i) = \Sigma_{user} dX(i,user)$$

$$[X0(i).P0(i)/100].[100.dX0(i)/X0(i)] = \Sigma_{user} dX(i,user).P0(i)$$

$$[X0(i).P0(i)/100].x0(i) = \Sigma_{user} delSales(i,user) \qquad \text{E\_p0A}$$

E_x0imp has same basic form, but equates demand for imports with supply of imports.

Excerpt 21c: Market clearing

```
Set LOCUSER # Non-export users #
(Interm, Invest, HouseH, GovGE, Stocks,Margins);
Subset LOCUSER is subset of DEST;

Equation E_p0A # Supply = Demand for domestic goods #
(all,c,COM)   0.01*[TINY+DOMSALES(c)]*x0dom(c)
=sum{u,LOCUSER,delSale(c,"dom",u)};

Variable (all,c,COM)  x0imp(c)  # Total supplies of imports #;
Equation E_x0imp # Import volumes #
 (all,c,COM)    0.01*[TINY+V0IMP(c)]*x0imp(c) =
sum{u,LOCUSER,delSale(c,"imp",u)};
```

Progress so far . . .

Excerpt 22: Purchasers prices

All purchaser's price equations have the same basic form:

Purchaser's value of commodity c used by user N

Value of margins associated with the purchase

$$PN_c \,.\, XN_c = P0c \,.\, XN_c \,.\, T_c + \Sigma_{mar}\, X_{mar,\,c} \,.\, P_{mar}$$

value preservation

Basic value of commodity c used by user N

Power of tax (= 1 + rate of tax) Eg. 1.03

. . . linearising (and dropping subscripts) . . .

$$[P.X]\,(p + x) = [P0.X.T]\,(p0 + x + t) + \Sigma_{mar}[X_{mar}.P_{mar}]\,(x_{mar} + p_{mar})$$

. . . noting that demand for margins is: $x_{mar} = x + a_{mar}$

$$[P.X]\,p = [P0.X.T]\,(p0 + t) + \Sigma_{mar}[X_{mar}.P_{mar}]\,(a_{mar} + p_{mar})$$

Standard form

Excerpt 22: Purchasers prices

```
Variable  ! example Government  !
 (all,c,COM)(all,s,SRC)  t5(c,s)   # Power of tax on government #;

Equation E_p5 # Zero pure profits in distribution to government #
 (all,c,COM)(all,s,SRC)
  [V5PUR(c,s)+TINY]*p5(c,s) =
   [V5BAS(c,s)+V5TAX(c,s)]*[p0(c,s)+ t5(c,s)]
    + sum{m,MAR, V5MAR(c,s,m)*[p0dom(m)+a5mar(c,s,m)]};

! alternate form    Equation E_p5q
(all,c,COM)(all,s,SRC) [V5PUR(c,s)+TINY]*p5(c,s) =
  [V5BAS(c,s)+V5TAX(c,s)]*p0(c,s)
+ 100*V5BAS(c,s)*delt5(c,s)
+ sum{m,MAR, V5MAR(c,s,m)*[p0dom(m)+a5mar(c,s,m)]};  !
```

Excerpt 23: Tax rate equations

```
Variable  ! example Intermediate  !
f1tax_csi    # Uniform %change in power of tax on intermediate usage #;
(all,c,COM)  f0tax_s(c)      # General sales tax shifter #;
Equation
E_t1  # Power of tax on sales to intermediate #
  (all,c,COM)(all,s,SRC)(all,i,IND)  t1(c,s,i)  = f0tax_s(c) + f1tax_csi;
```

power of tax =
1 + ad valorem rate:
1.2 means 20% tax

default rule:
modeller could
change for special
experiment

Excerpt 24: Tax Updates

Before: ! example Intermediate !

```
Coefficient (all,c,COM)(all,s,SRC)(all,i,IND)
        V1TAX(c,s,i) # Taxes on intermediate #;
Read  V1TAX from file BASEDATA header "1TAX";
Variable  (change)(all,c,COM)(all,s,SRC)(all,i,IND)
     delV1TAX(c,s,i) # Interm tax rev #;
Update  (change)(all,c,COM)(all,s,SRC)(all,i,IND)
      V1TAX(c,s,i) = delV1TAX(c,s,i);
```

```
Equation
 E_delV1TAX (all,c,COM)(all,s,SRC)(all,i,IND)
```

original tax revenue
× proportional change (=%/100) in tax base

```
delV1TAX(c,s,i) = 0.01*V1TAX(c,s,i)* [x1(c,s,i) + p0(c,s)]
             +     0.01*[V1BAS(c,s,i)+V1TAX(c,s,i)]*t1(c,s,i);
```

change in tax rate
× the original [base + tax]

Excerpt 25: Import prices

Variable

(all,c,COM) pf0cif(c) # CIF foreign currency import prices #;

(all,c,COM) t0imp(c) # Power of tariff #;

Equation E_p0B # Zero pure profits in importing #

(all,c,COM) p0(c,"imp") = pf0cif(c) + phi + t0imp(c);

Equation E_delV0TAR (all,c,COM)

delV0TAR(c) = 0.01*V0TAR(c)*[x0imp(c)+pf0cif(c)+phi] + 0.01*V0IMP(c)*t0imp(c);

$$P_{imp} = P_f \Phi (1+V)$$

$$= P_f \Phi (T0IMP)$$

T0IMP = power = 1 + ad valorem rate

exchange rate (Φ, phi) = local dollars per foreign dollar

Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- Government demands
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- → **Aggregates and indices**
- Investment allocation
- Labour market
- Decompositions
- Closure
- Regional extension

Excerpt 26: Tax Revenue Totals

```
Coefficient
 V1TAX_CSI  # Total intermediate tax revenue #;
........................
 V0TAR_C    # Total tariff revenue #;
Formula
 V1TAX_CSI  = sum{c,COM, sum{s,SRC, sum{i,IND, V1TAX(c,s,i)}}};
........................
V0TAR_C    = sum{c,COM,   V0TAR(c)};
Variable
(change) delV1tax_csi # Agg. revenue from indirect taxes on
    intermediate #;
........................
(change) delV0tar_c   # Aggregate tariff revenue #;

Equation
E_delV1tax_csi
delV1tax_csi = sum{c,COM, sum{s,SRC, sum{i,IND, delV1TAX(c,s,i) }}};
........................
E_delV0tar_c   delV0tar_c   = sum{c,COM, delV0TAR(c)};
```

Excerpt 27: Factor incomes and GDP

Example Capital

```
Coefficient  V1CAP_I   # Total payments to capital #;
Formula      V1CAP_I   = sum{i,IND, V1CAP(i)};

Variable  w1cap_i   # Aggregate payments to capital #;
Equation  E_w1cap_i
 V1CAP_I*w1cap_i = sum{i,IND,
   V1CAP(i)*[x1cap(i)+p1cap(i)]};

E_w0gdpinc   V0GDPINC*w0gdpinc =
  V1LND_I*w1lnd_i + V1CAP_I*w1cap_i +
   V1LAB_IO*w1lab_io + 100*delV0tax_csi;
```

Excerpt 27: GDP - Production tax example

```
Coefficient   V1PTX_I   # Total production tax/subsidy #;
Formula      V1PTX_I   = sum{i,IND, V1PTX(i)};
Variable  (change) delV1PTX_i
# Ordinary change in all-industry production tax revenue #;
Equation E_delV1PTX_i
   delV1PTX_i=sum{i,IND,delV1PTX(i)};

E_delV0tax_csi # Total indirect tax revenue #
   delV0tax_csi =    delV1tax_csi + delV2tax_csi +
   delV3tax_cs + delV4tax_c + delV5tax_cs + delV0tar_c +
   delV1PTX_i + 0.01*V1OCT_I*w1oct_i;

E_w0gdpinc   V0GDPINC*w0gdpinc = V1LND_I*w1lnd_i +
   V1CAP_I*w1cap_i + V1LAB_IO*w1lab_io +
   100*delV0tax_csi;
```

Excerpt 28a: GDP expenditure aggregates

```
Coefficient ! Expenditure Aggregates at Purchaser's Prices !
 (all,c,COM) V0CIF(c) # Total ex-duty imports of good c #;
 V0CIF_C  # Total local currency import costs, excluding tariffs #;
 V0IMP_C  # Total basic-value imports (includes tariffs) #;
 V2TOT_I  # Total investment usage #;
  . . . . . . . . . . . . . . . . . .
 V0GDPEXP # Nominal GDP from expenditure side #;
Formula
 (all,c,COM) V0CIF(c) = V0IMP(c) - V0TAR(c);
 V0CIF_C   = sum{c,COM, V0CIF(c)};
 V0IMP_C   = sum{c,COM, V0IMP(c)};
 V2TOT_I   = sum{i,IND, V2TOT(i)};
 V4TOT     = sum{c,COM, V4PUR(c)};
 V5TOT     = sum{c,COM, sum{s,SRC, V5PUR(c,s)}};
 V6TOT     = sum{c,COM, sum{s,SRC, V6BAS(c,s)}};
 V0GDPEXP  = V3TOT + V2TOT_I + V5TOT + V6TOT + V4TOT - V0CIF_C;
```

Excerpt 28b: GDP expenditure aggregates

Investment example

```
Coefficient   V2TOT_I  # Total investment usage #;
Formula       V2TOT_I  = sum{i,IND, V2TOT(i)};
Variable
 x2tot_i     # Aggregate real investment expenditure #;
 p2tot_i     # Aggregate investment price index #;
 w2tot_i     # Aggregate nominal investment #;
Equation
 E_x2tot_i  V2TOT_I*x2tot_i = sum{i,IND, V2TOT(i)*x2tot(i)};
 E_p2tot_i  V2TOT_I*p2tot_i = sum{i,IND, V2TOT(i)*p2tot(i)};
 E_w2tot_i  w2tot_i = x2tot_i + p2tot_i;
```

Excerpt 28c: GDP expenditure aggregates

Inventory example

```
Coefficient   V6TOT   # Total value of inventories #;
Formula       V6TOT    = sum{c,COM, sum{s,SRC,
   V6BAS(c,s)}};

Variable
 x6tot     # Aggregate real inventories #;
 p6tot     # Inventories price index #;
 w6tot     # Aggregate nominal value of inventories #;

Equation
 E_x6tot [TINY+V6TOT]*x6tot
   =100*sum{c,COM,sum{s,SRC,LEVP0(c,s)*delx6(c,s)}};
 E_p6tot [TINY+V6TOT]*p6tot
   = sum{c,COM, sum{s,SRC, V6BAS(c,s)*p0(c,s)}};
 E_w6tot   w6tot = x6tot + p6tot;
```

Excerpt 28d: GDP expenditure aggregates

```
Coefficient V0GDPEXP # Nominal GDP from expenditure side #;
Formula     V0GDPEXP  = V3TOT + V2TOT_I + V5TOT + V6TOT +
    V4TOT - V0CIF_C;

Variable
 x0gdpexp   # Real GDP from expenditure side #;
 p0gdpexp   # GDP price index, expenditure side #;
 w0gdpexp  # Nominal GDP from expenditure side #;

Equation
 E_x0gdpexp   V0GDPEXP*x0gdpexp =
     V3TOT*x3tot + V2TOT_I*x2tot_i + V5TOT*x5tot
    + V6TOT*x6tot  + V4TOT*x4tot - V0CIF_C*x0cif_c;
 E_p0gdpexp V0GDPEXP*p0gdpexp =
     V3TOT*p3tot + V2TOT_I*p2tot_i + V5TOT*p5tot
     + V6TOT*p6tot  + V4TOT*p4tot - V0CIF_C*p0cif_c;
 E_w0gdpexp w0gdpexp = x0gdpexp + p0gdpexp;
```

Excerpt 29: Trade measures

```
Variable
 (change) delB # (Balance of trade)/GDP #;
        x0imp_c # Import volume index, duty-paid weights #;
       w0imp_c # Value of imports plus duty #;
       p0imp_c # Duty-paid imports price index, local currency #;
     p0realdev # Real devaluation #;
          p0toft # Terms of trade #;
Equation
 E_delB 100*V0GDPEXP*delB=V4TOT*w4tot -V0CIF_C*w0cif_c
    - (V4TOT-V0CIF_C)*w0gdpexp;
 E_x0imp_c  V0IMP_C*x0imp_c=sum{c,COM, V0IMP(c)*x0imp(c)};
 E_p0imp_c
    V0IMP_C*p0imp_c=sum{c,COM,V0IMP(c)*p0(c,"imp")};
 E_w0imp_c    w0imp_c = x0imp_c + p0imp_c;
 E_p0toft        p0toft = p4tot - p0cif_c;
 E_p0realdev  p0realdev =  p0cif_c - p0gdpexp;
```

Excerpt 30: Factor Aggregates

Variable ***(Selected)***

```
Variable ( Selected )
 (all,i,IND) employ(i) # Employment by industry #;
 employ_i   # Aggregate employment: wage bill weights #;
 x1cap_i    # Aggregate capital stock, rental weights #;
 x1prim_i   # Aggregate output: value-added weights #;
 p1lab_io   # Average nominal wage #;
 realwage   # Average real wage #;
Equation
 E_employ  (all,i,IND)    V1LAB_O(i)*employ(i)
                  = sum{o,OCC, V1LAB(i,o)*x1lab(i,o)};
 E_employ_i  V1LAB_IO*employ_i = sum{i,IND,
    V1LAB_O(i)*employ(i)};
 E_x1cap_i   V1CAP_I*x1cap_i   = sum{i,IND, V1CAP(i)*x1cap(i)};
 E_x1prim_i V1PRIM_I*x1prim_i = sum{i,IND, V1PRIM(i)*x1tot(i)};
 E_p1lab_io  V1LAB_IO*p1lab_io = sum{i,IND, sum{o,OCC,
    V1LAB(i,o)*p1lab(i,o)}};
 E_realwage realwage = p1lab_io - p3tot;
```

Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- Government demands
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- Aggregates and indices
- → **Investment allocation**
- Labour market
- Decompositions
- Closure
- Regional extension

Excerpt 31: Investment

For each industry i, investment x2tot(i) follows one of three rules:

1: Investment positively related to profit rate (short-run),
x2tot(i) = f(profit) + finv1(i) + invslack

2: Investment follows national investment, x2tot_i
x2tot(i) = x2tot_i + finv2(i)

3: Investment follows industry capital stock (long-run):
x2tot(i) = x1cap(i) + finv3(i) + invslack

For each industry i, one of the finv shift variables exogenous.

Optional extra: rules can accommodate fixed national investment.

One of invslack or x2tot_i exogenous.

Excerpt 31: Investment

RULE 1: Investment positively related to profit rate (short-run).

Equation E_gret

First, we define the net rate of return as:

NRET(i) = P1CAP(i)/P2TOT(i) - DEP(i) = GRET(i) - DEP(i) {levels}

nret(i) = [GRET(i) / NRET(i)] * gret(i) {% change}

Substituted into RHS of E_finv1 as 2.0 * gret(i)

Variable

gret(i) # Gross rate of return = Rental/[Price of new capital] #;

Equation E_gret gret(i) = p1cap(i) - p2tot(i);

Excerpt 31: Investment

Second, we define the gross growth rate of capital as:

GGRO(i) = X2TOT(i) / X1CAP(i) {levels}

Equation E_ggro ggro(i) = x2tot(i) - x1cap(i) {% change}

Third, we relate the gross growth rate to the net rate of return via

Equation E_finv1 # DPSV investment rule #
(all,i,IND) ggro(i) = finv1(i) + 0.33*[2.0*gret(i) - invslack];

ie. GRET = 2 x DEP

Sensitivity of capital growth to rates of return

Excerpt 31: "Exogenous" investment industries

RULE 2: Industry investment follows national investment.

This rule is applied in those cases where investment is not thought to be mainly driven by current profits (eg, Education)

```
Equation E_finv2
# Alternative rule for "exogenous" investment industries #
 (all,i,IND) x2tot(i) = x2tot_i + finv2(i);
```

BUT: Do not set ALL the finv2's exogenous: would conflict with:

```
Equation E_x2tot_i
      V2TOT_I*x2tot_i = sum{i,IND, V2TOT(i)*x2tot(i)};
```

At solve time: "singular matrix" error.

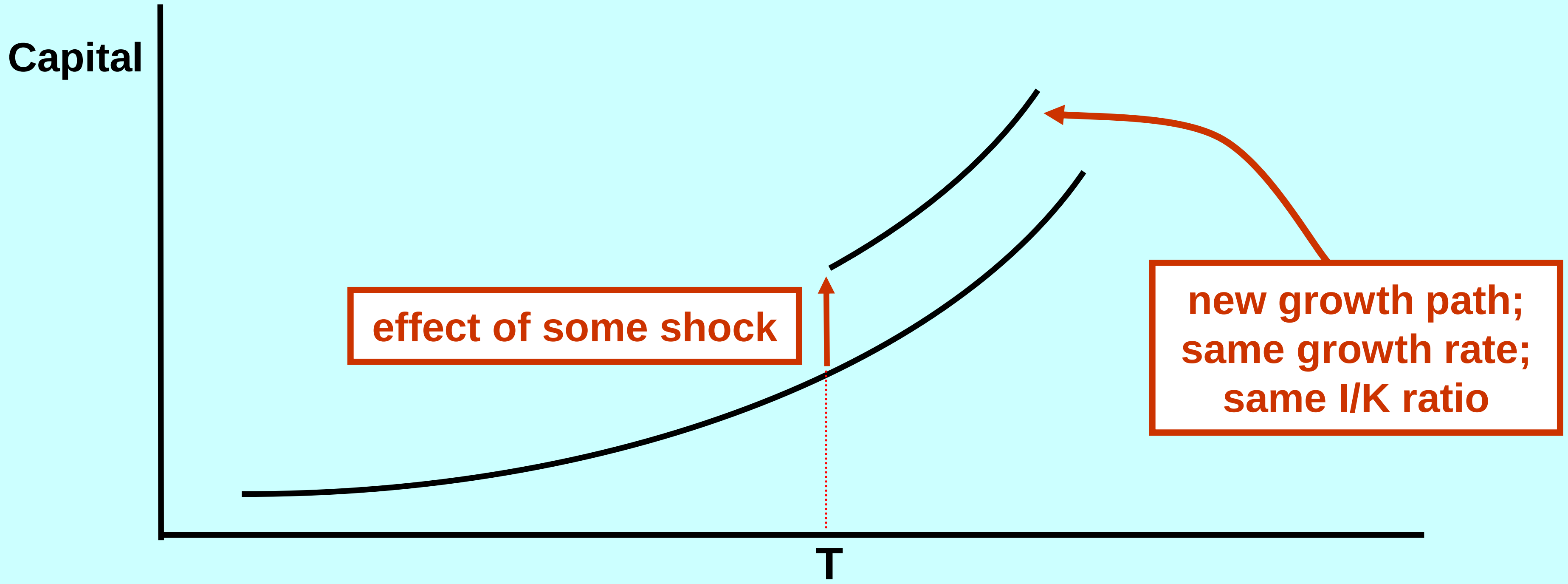
Excerpt 31: Longrun Investment Rule

RULE 3: investment/capital ratios are exogenous

Equation E_finv3 (all,i,IND) ggro(i) = finv3(i) + invslack

Recall:

ggro(i) # Gross growth rate of capital = Investment/capital #
= x2tot(i) - x1cap(i);



Excerpt 31: Aggregate Investment

Three ways to set aggregate investment in ORANI-G

1. x2tot endogenous (invslack exogenous)
   industry specific rules determine aggregate

2. x2tot exogenous (invslack endogenous)

3. x2tot linked to Cr (invslack endogenous)
   Variable f2tot # Ratio, investment/consumption #;
   Equation E_f2tot x2tot_i = x3tot + f2tot;

   Implemented by seting f2tot exog and invslack endog

Capital and Investment

ORANI-G: choice of 2 comp. stat. treatments

Shortrun: x1cap(i) fixed   x2tot(i) profit driven or exogenous

Longrun: gret(i) fixed   x2tot(i) follows x1cap(i)

NOT IN ORANI-G

Accumulation rule: Capital = function(investment)

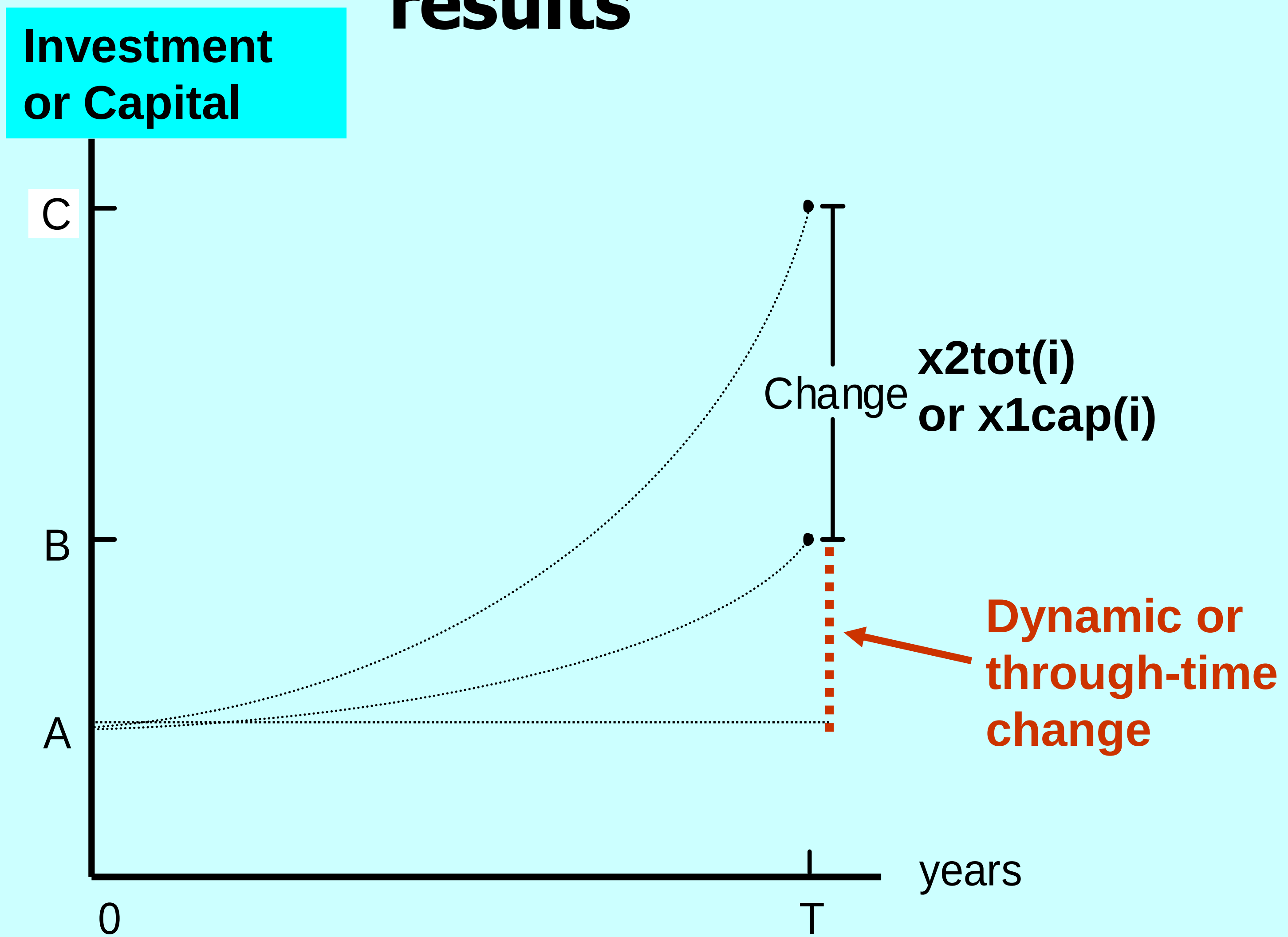
ΔX1CAP = X2TOT - Depreciation*(X1CAP)

MONASH: Series of shortruns:

x1cap(i) determined by previous period investment

x2tot(i) profit driven or exogenous

Comparative-static interpretation of results



Results refer to changes at some future point in time.

Fixed total capital, mobile between sectors

Equation E_fgret # *force rates of return to move together* #

(all,i,IND) gret(i) = fgret(i) + capslack;

Normally, capslack exogenous and zero, fgret endogenous:

fgret(i) = gret(i);

just determines fgret(i).

With capslack and gret endogenous,

x1cap_i and fgret(i) exogenous:

gret(i) = capslack;

all sectoral rates of return move together

p45

Summary of closure options

	Short-run	Long-run	Fixed capital
x1cap(i)	X	N (a)	N
finv1(i ∈ J)	X	N (b)	N
finv2(i ∉ J)	X	N (c)	N
finv3(i)	N	X (b) (c)	X
gret(i)	N	X (a)	N (a)
fgret(i)	N	N	X (a)
capslack	X	X	N (b)
x1cap_i	N	N	X (b)
x2tot(i)	N	N	N
finv1(i ∉ J)	N	N	N
finv2(i ∈ J)	N	N	N
invslack	N	N	N
x2tot_i	X	X	X

X:eXogenous
N:eNdogenous

(J : endogenous investment industries)

Progress so far . . .

Introduction
Database structure
Solution method
TABLO language
Production: input decisions
Production: output decisions
Investment: input decisions
Household demands
Export demands
Government demands

Inventory demands
Margin demands
Market clearing
Price equations
Aggregates and indices
Investment allocation
→ **Labour market**
Decompositions
Closure
Regional extension

Excerpt 32: Labour market

```
Variable
 (all,i,IND)(all,o,OCC) f1lab(i,o) # Wage shift variable #;
         (all,o,OCC) f1lab_i(o) # Occupation-specific wage shifter #;
         (all,i,IND)  f1lab_o(i) # Industry-specific wage shifter #;
                        f1lab_io   # Overall wage shifter #;

E_p1lab   # Wage setting #   (all,i,IND)(all,o,OCC)
   p1lab(i,o)= p3tot + f1lab_io + f1lab_o(i) + f1lab_i(o) + f1lab(i,o);
```

Short run: f1lab_io fixed, aggregate employment varies

Long run: f1lab_io varies, aggregate employment exogenous

```
E_x1lab_i  # Employment by occupation #  (all,o,OCC)
   V1LAB_I(o)*x1lab_i(o) = sum{i,IND, V1LAB(i,o)*x1lab(i,o)};
```

Excerpt 33: Miscellaneous

Variable (all,i,IND) f1oct(i) Shift in price of "other cost" tickets

Equation E_p1oct # Indexing of prices of "other cost" tickets #

(all,i,IND) p1oct(i) = p3tot + f1oct(i);   ! assumes full indexation !

Variable f3tot # Ratio, consumption/ GDP #;

Equation E_f3tot # Consumption function #

w3tot = w0gdpexp + f3tot;

Vector variables are easier to look at in results:

Basic price of domestic goods: p0dom(c) = p0(c,"dom");

Basic price of imported goods: p0imp(c) = p0(c,"imp");

Progress so far . . .

Introduction
Database structure
Solution method
TABLO language
Production: input decisions
Production: output decisions
Investment: input decisions
Household demands
Export demands
Government demands
Inventory demands
Margin demands
Market clearing
Price equations
Aggregates and indices
Investment allocation
Labour market
→ **Decompositions**
Closure
Regional extension

Variables to explain results

Decomposition breaks down a percent change into contributions due to various parts or causes.

3 Decompositions:

Sales Decomposition

breaks down sales change by different markets

Fan Decomposition (causal)

breaks sales change into

- growth of local market effect
- import/domestic competition effect
- export effect

Expenditure-side GDP Decomposition

breaks down GDP by main expenditure aggregates

Contributions in Decompositions

In explaining results, it is sometimes useful to be able to decompose the percentage change in *x* into the individual contributions of the RHS variables.

EG: X = A + B (Levels)

or PX = PA + PB (x through by common price, P)

Small % change: x = (PA/PX)a + (PB/PX)b

conta =(PA/PX)a

contb =(PB/PX)b

***contribution* of A to % change in X**

***contribution* of B to % change in X**

x = conta + contb

Would not add up right in multistep computation, if x, conta and contb were percent changes (compounded).

Contributions in Decompositions

Solution: define *conta* and *contb* as ordinary change variables, and make a new ordinary change variable, *q*.

EG: X = A + B

dX = dA + dB

[0.01 X0][100 .dX / X0] = dA + dB

initial *ordinary change*

[0.01 X0] q = dA + dB

changes: so will add

but: we want % changes

new change variable: q

multiply through by common price:

[P X0] q = 100 [P dA] + 100 [P dB]

q = [100 / P X0] [P dA] + [100 / P X0] [P dB]

NB: change in quantity valued @ current price

Decomp Equations

[P X0] conta = 100 [P dA]

[P X0] conta = [P A] a

Standard forms

Excerpt 34: Sales Decomposition

Breaks down %change in domestic sales into contributions from each main customer:

Say domestic shoe sales went up 4.1%

Intermediate	1%
Investment	0
Household	5%
Government	0.1%
Export	-2%
Inventories	0
Total	4.1%

Equation E_SalesDecompA

Excerpt 35: Fan Decomposition

Output of Shoes up 4.1% ..... why:

3 possible reasons:

Local Market Effect: demand for shoes (dom + imp) is up.

Domestic Share Effect: ratio (dom/imp) shoes is up.

Export Effect: shoe exports are up.

$X = L*S_{dom} + E$ L=all shoe sales $L*S_{dom}$=local sales dom shoes

$x = [L*S_{dom}/X][l + s_{dom}] + [E/X]e$ E=export sales

$x = [L*S_{dom}/X]\, l + [L*S_{dom}/X]s_{dom} + [E/X]e$

Local Market Domestic Share Export

Fan decomposition breaks down output change between these three components.

Very useful for understanding results.

Excerpt 36: Expenditure side GDP Decomposition

Shows contributions of main expenditure aggregates to % change in real GDP

NB: Standard form

```
INITGDP*contGDPexp("Consumption") = V3TOT*x3tot;
INITGDP*contGDPexp("Investment")  = V2TOT_I*x2tot_i;
INITGDP*contGDPexp("Government")  = V5TOT*x5tot;
INITGDP*contGDPexp("Stocks")      = V6TOT*x6tot;
INITGDP*contGDPexp("Exports")     = V4TOT*x4tot;
INITGDP*contGDPexp("Imports")     = - V0CIF_C*x0cif_c;
```

Initial GDP valued at current price

Change variable

Excerpt 36:
Income side GDP Decomposition

Shows contributions of

- **primary factor usage,**
- **indirect taxes, and**
- **technological change.**

to % change in real GDP

Excerpt 37 -42: The Summary file

Many useful aggregates

Regional Extension

covered in a later lecture

Progress so far . . .

- Introduction
- Database structure
- Solution method
- TABLO language
- Production: input decisions
- Production: output decisions
- Investment: input decisions
- Household demands
- Export demands
- Government demands
- Inventory demands
- Margin demands
- Market clearing
- Price equations
- Aggregates and indices
- Investment allocation
- Labour market
- Decompositions
- → **Closure**
- Regional extension

Closing the model

Each equation explains a variable.

More variables than equations.

Endogenous variables: explained by model

Exogenous variables: set by user

Closure: choice of exogenous variables

Many possible closures

Number of endogenous variables = Number of equations

One way to construct a closure:

(a) Find the variable that each equation explains; it is endogenous.

(b) Other variables, not explained by equations, are exogenous.

ORANI-G equations are named after the variable they SEEM to explain. TABmate uses equation names for automatic closure.

Variables not explained by any equation = possible exogenous list

1 Dimension	2 Variable Count	3 Equation Count	4 Exogenous Count	5 List of unexplained variables (Mechanical closure)
MACRO	70	56	14	f1lab_io f4p_ntrad f4q_ntrad f4tax_trad f4tax_ntrad *f5tot2* phi q *invslack* *w3lux* f1tax_csi f2tax_csi f3tax_cs f5tax_cs
COM	25	19	6	f0tax_s t0imp a3_s f4p f4q pf0cif
COM∗IND	7	5	2	a1_s a2_s
COM∗MAR	2	1	1	a4mar
COM∗SRC	14	11	3	f5 a3 *fx6*
COM∗SRC∗IND	10	8	2	a1 a2
COM∗SRC∗IND∗MAR	4	2	2	a1mar a2mar
COM∗SRC∗MAR	4	2	2	a3mar a5mar
IND	34	21	13	a1cap a1lab_o a1lnd a1oct a1prim a1tot f1lab_o f1oct *x2tot* x1lnd a2tot x1cap delPTXRate
IND∗OCC	3	2	1	f1lab
OCC	2	1	1	f1lab_i
COM∗SRC∗DEST	1	1	0	
COM∗DESTPLUS	1	1	0	
COM*FANCAT	1	1	0	
EXPMAC	1	1	0	
TOTAL	179	132	47	

The ORANI short-run closure

Exogenous variables constraining real GDP from the supply side	
x1cap x1lnd	**industry-specific endowments of capital and land**
a1cap a1lab_o a1lnd a1prim a1tot a2tot	**all technological change**
f1lab_io	**real wage shift variable**
Exogenous settings of real GDP from the expenditure side	
x3tot	**aggregate real private consumption expenditure**
x2tot_i	**aggregate real investment expenditure**
x5tot	**aggregate real government expenditure**
f5	**distribution of government demands**
delx6	**real demands for inventories by commodity**
Foreign conditions: import prices fixed; export demand curves fixed in quantity and price axes	
pf0cif	**foreign prices of imports**
f4p f4q	**individual exports**
f4p_ntrad f4q_ntrad	**collective exports**
All tax rates are exogenous	
delPTXRATE f0tax_s f1tax_csi f2tax_csi f3tax_cs f5tax_cs t0imp f4tax_trad f4tax_ntrad f1oct	
Distribution of investment between industries	
finv1(selected industries)	**investment related to profits**
finv2(the rest)	**investment follows aggregate investment**
Number of households and their consumption preferences are exogenous	
q	**number of households**
a3_s	**household tastes**
Numeraire assumption	
phi	**nominal exchange rate**

Length of run ,T

T is related to our choice of closure.

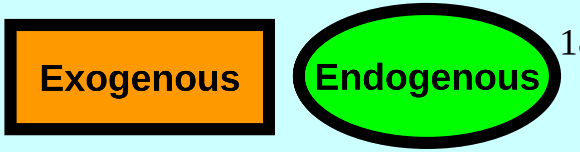
With shortrun closure we assume that:

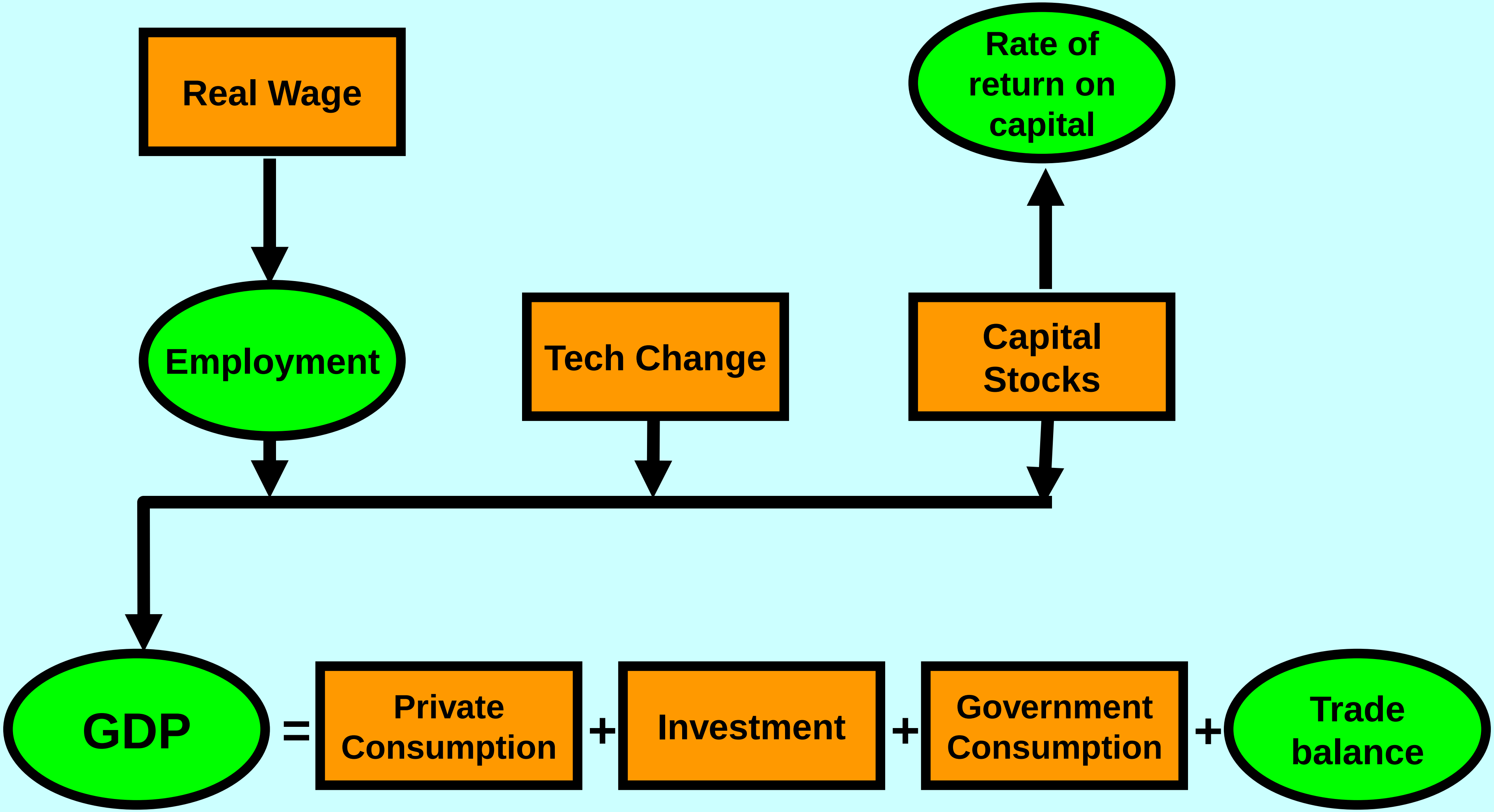
- **T is long enough for price changes to be transmitted throughout the economy, and for price-induced substitution to take place.**
- **T is not long enough for investment decisions to greatly affect the useful size of sectoral capital stocks. [New buildings and equipment take time to produce and install.]**

T might be 2 years. So results mean:

a 10% consumption increase might lead to employment in 2 years time being 1.24% higher than it would be (in 2 years time) if the consumption increase did not occur.

Causation in Short-run Closure





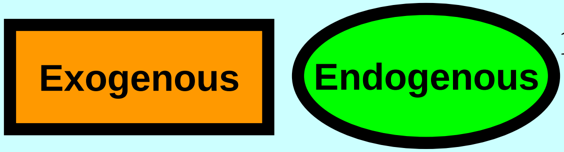
A possible long-run closure

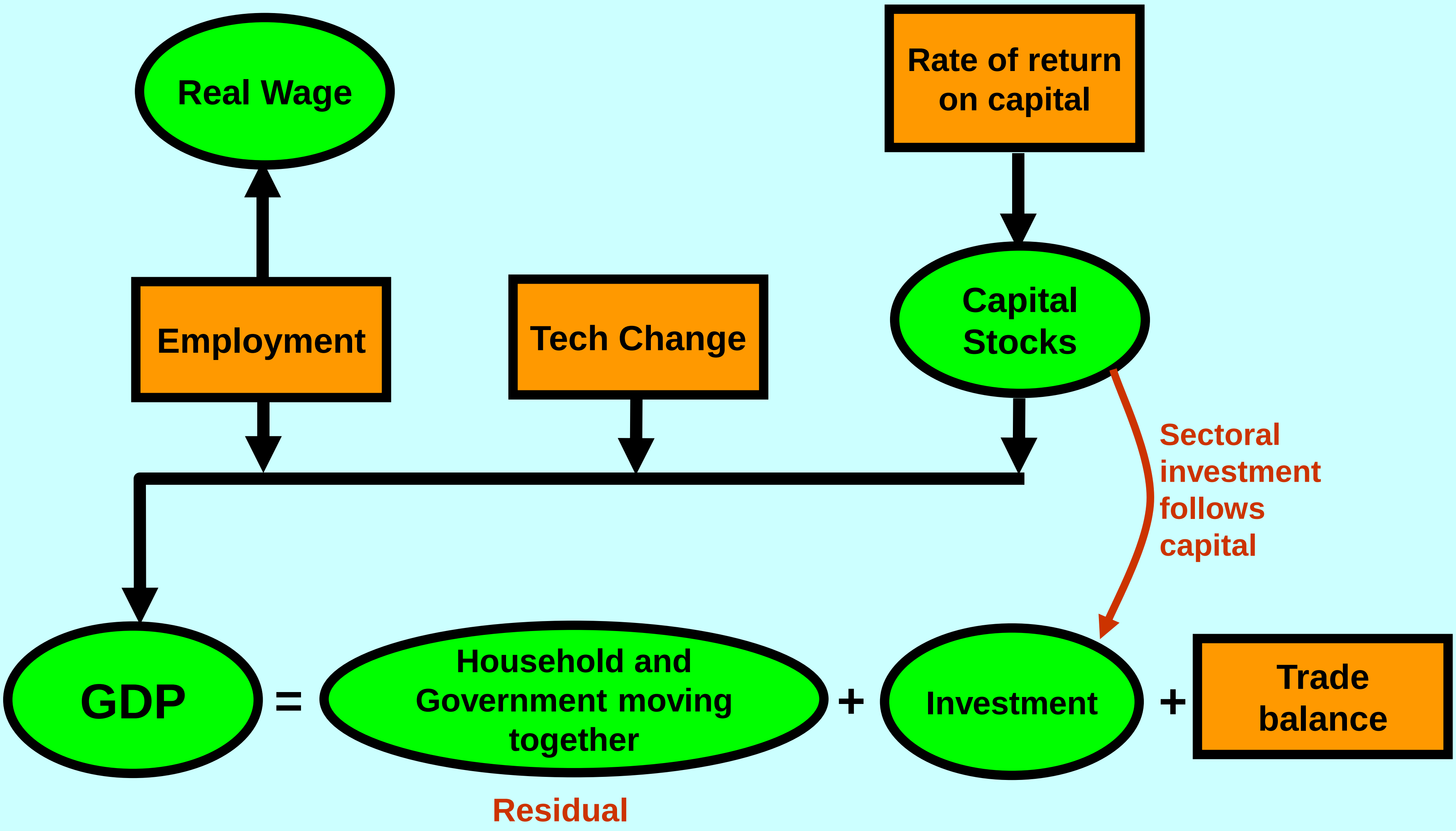
- Capital stocks adjust in such a way to maintain fixed rates of return (gret).
- Aggregate employment is fixed and the real wage adjusts.
- DelB fixed instead of x3tot (real household consumption)
- x3tot (household) and x5tot (government) linked to move together

Table 4: A possible long-run closure

Exogenous variables constraining real GDP from the supply side	
gret	***gross sectoral rates of return***
x1lnd	**industry-specific endowments of land**
a1cap a1lab_o a1lnd a1prim a1tot a2tot	**all technological change**
employ_i	***total employment - wage weights***
Exogenous settings of real GDP from the expenditure side	
delB	***balance of trade/GDP***
invslack	***aggregate investment determined by industry specific rules***
f5tot2	***link government demands to total household***
f5	**distribution of government demands**
delx6	**real demands for inventories by commodity**
Foreign conditions: import prices fixed; export demand curves fixed in quantity and price axes	
pf0cif	**foreign prices of imports**
f4p f4q	**individual exports**
f4p_ntrad f4q_ntrad	**collective exports**
All tax rates are exogenous	
delPTXRATE f0tax_s f1tax_csi f2tax_csi f3tax_cs f5 f5tax_cs t0imp f4tax_trad f4tax_ntrad f1oct	
Distribution of investment between industries	
finv3(selected industries)	***fixed investment/capital ratios***
finv2(the rest)	**investment follows aggregate investment**
Number of households and their consumption preferences are exogenous	
q	**number of households**
a3_s	**household tastes**
Numeraire assumption	
phi	**nominal exchange rate**

Causation in Long-run Closure





Different closures

Many closures might be used for different purposes.

No unique natural or correct closure.

Must be at least one exogenous variable measured in local currency units.

Normally just one — called the *numeraire*.

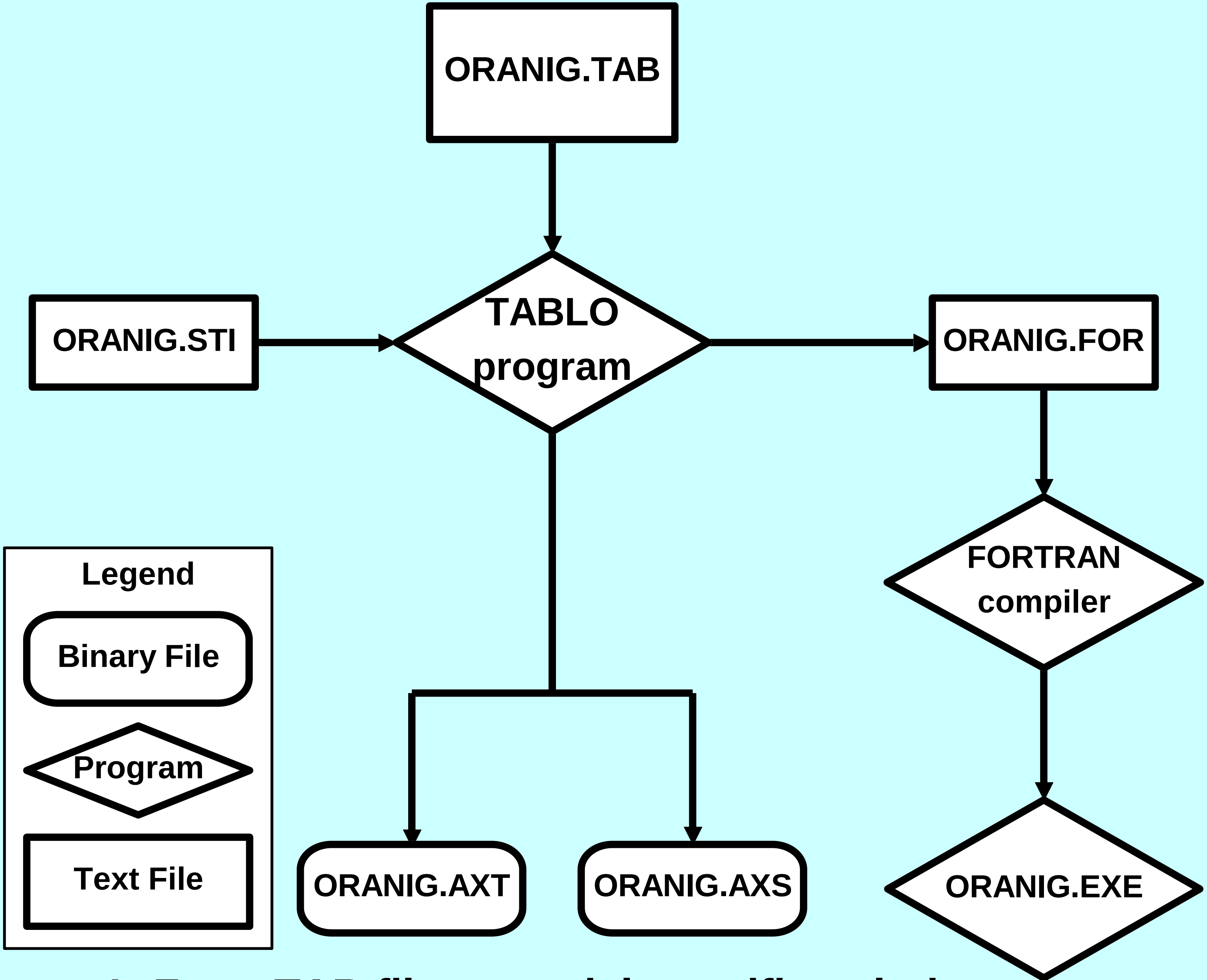
Often the exchange rate, phi, or p3tot, the CPI.

Some quantity variables must be exogenous, such as:

- **primary factor endowments**
- **final demand aggregates**

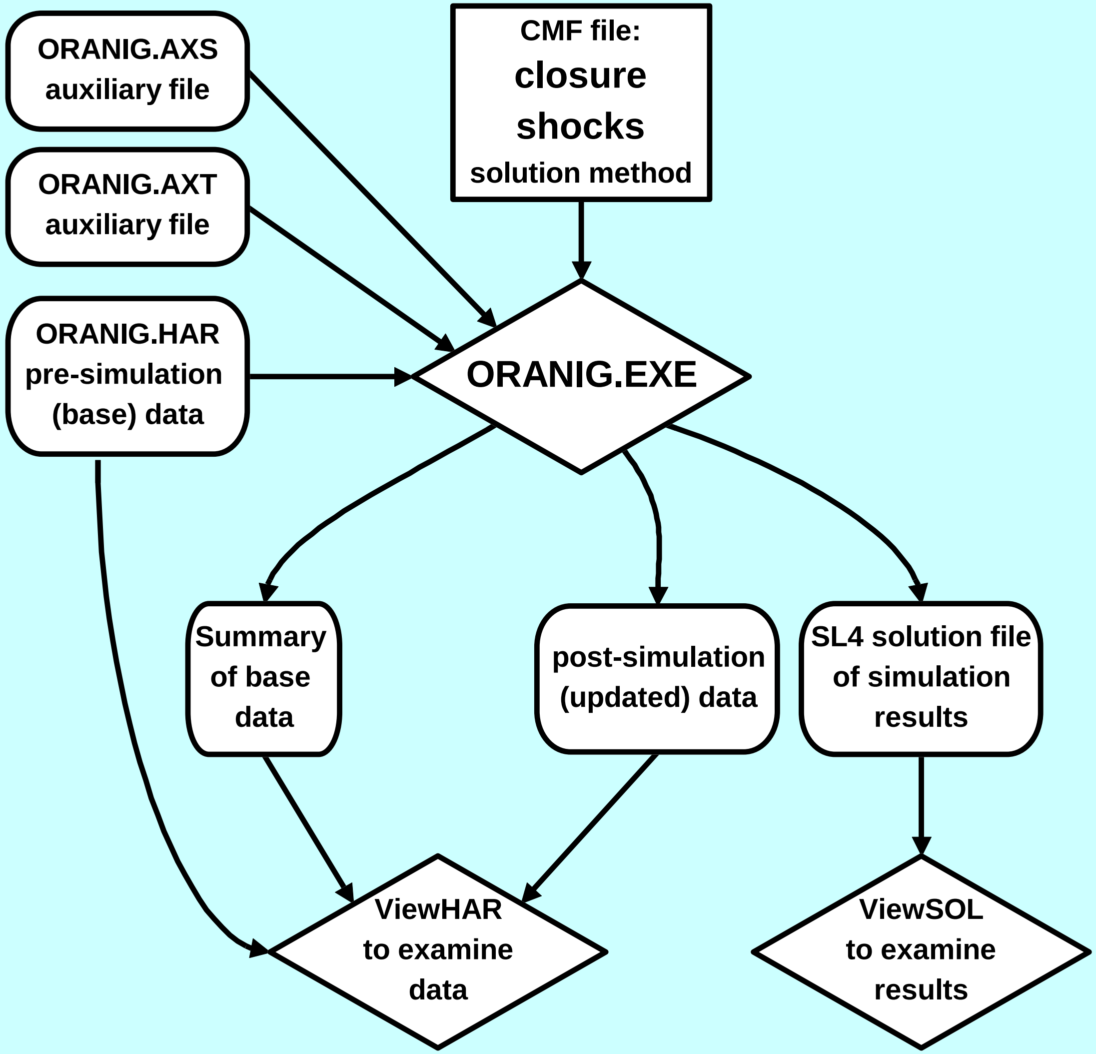
Three Macro Don't Knows

- ***Absolute price level.*** **Numeraire choice determines whether changes in the real exchange rate appear as changes in domestic prices or in changes in the exchange rate. Real variables unaffected.**
- ***Labour supply.*** **Closure determines whether labour market changes appear as changes in either wage or employment.**
- ***Size and composition of absorption.*** **Either exogenous or else adusting to accommodate fixed trade balance. Closure determines how changes in national income appear.**

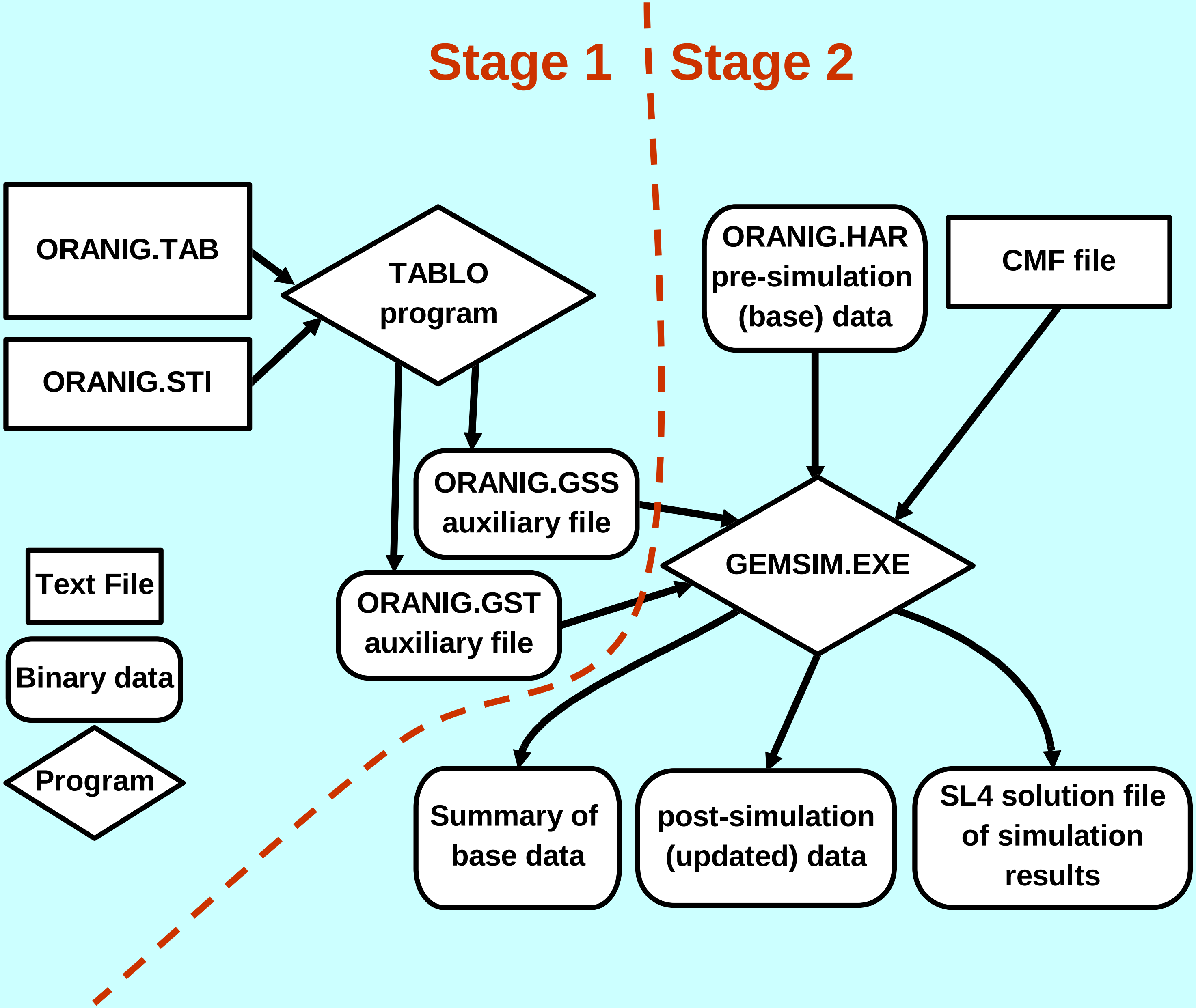


Stage 1: From TAB file to model-specific solution program

p61



Stage 2: Using the model-specific EXE to run a simulation



Using GEMSIM

The End