

AGA 511

Métodos Computacionais em Astronomia

Segundo semestre de 2017

Informações gerais

Prof. Alex Cavaliéri Carciofi

Email: carciofi@usp.br

Ramal: 2712

Colaborador: Carlos Eduardo Paladini

Email: carlos.paladini@iag.usp.br

Ramal: 2743

Programa resumido

1. Programação (boas práticas, algoritmos científicos, etc.)
2. HPC - High Performance Computing
3. MPI - Message passing interface
4. OpenMP - Open Multi-Processing
5. CUDA - Compute Unified Device Architecture
6. OpenCL (plataformas heterogêneas)

Proposta Geral

Curso não tradicional:

- (praticamente) sem aulas expositivas
- Focado em trabalhos práticos e estudos dirigidos
- Project oriented (orientado por projeto)

Proposta Geral

1. Escolher um **projeto interessante**, mas curto o suficiente para que possa ser resolvido em poucas semanas (**próxima aula!**)
2. Escrever um programa tradicional (**serial**) em C++ ou Fortran que resolva este problema
 - * Exercício individual
 - * Código bem estruturado: comentado, compartimentalizado, eficiente

Proposta Geral

3. Escolher uma das técnicas abaixo, e **refazer o programa** incorporando técnicas de HPC. visando aumentar a eficiência do código:

- a) **MPI - C++**
- b) **MPI - FORTRAN**
- c) **OpenMP - C++ ou FORTRAN**
- d) **CUDA - C++**
- e) **OpenCL**

* Esta parte do curso será feita em duplas

Avaliação

- EP 1 - O problema da agulha
- EP 2 - Versão serial do projeto (**atividade individual**)
- EP 3 - Versão HPC do projeto (**atividade em duplas**)
 - ▶ Duas reuniões comigo ao longo do semestre
 - ▶ Apresentação ao final do curso, para toda a turma, contendo:
 - Descrição do projeto desenvolvido
 - Descrição da técnica de HPC empregada
 - Descrição detalhada da implementação final
 - Discussão sobre o aumento de eficiência com relação ao código tradicional
- **Nota final:** $0,1 \text{ EP1} + 0,3 \text{ EP2} + 0,3 \text{ EP3} + 0,3 \text{ Apresentação}$
- Frequência será monitorada (lista)

Calendário

- 9/8 - Aula inicial
- 16/8 - Aula sobre possíveis projetos
- 23/8 - DEFINIÇÃO DOS PROJETOS - Introdução à HPC
- 30/8 - Introdução à HPC - Entrega do EP1
- 11/9 - Entrega do EP 2 (SEGUNDA-FEIRA)
- 13/9 - Correção do EP 2 em aula / FORMAÇÃO DAS DUPLAS DE TRABALHO
- 20/9 - Reunião Inicial: Duplas 1 e 2
- 27/9 - Reunião Inicial: Duplas 3 e 4
- 4/10 - Reunião Inicial: Dupla 5
- 11/10 - Reunião de acompanhamento: Duplas 1 e 2
- 18/10 - Reunião de acompanhamento: Duplas 3 e 4
- 25/10 - Reunião de acompanhamento: Dupla 5
- 1/11 - Não haverá aula
- 8/11 - Apresentação final: Dupla 1
- 22/11 - Apresentação final: Dupla 2
- 29/11 - Apresentação final: Dupla 3
- 6/12 - Apresentação final: Dupla 4
- 13/12 - Apresentação final: Dupla 5

A vertente computacional da graduação em astronomia

Composta por cinco disciplinas:

AGA0503 – Métodos Numéricos em Astronomia (5º semestre, obrigatória, 4 c.)

AGA0505 – Análise de Dados em Astronomia I (7º semestre, obrigatória, 2 c.)

AGA0511 – Métodos Computacionais em Astronomia (6º semestre, optativa, 2 c.)

AGA0512 – Análise de Dados em Astronomia II (8º semestre, optativa, 2 c.)

AGA0513 - e-Science em Astronomia (optativa, 2 c.)

O supercomputador do DA

Composto de três unidades.

1) Uma unidade Altix ICE 8400 com

2304 núcleos de processamento AMD opteron 6172

2 GB de RAM por núcleo (4.6 TB no total)

Interconect infiniband com topologia hypercube

2) Uma unidade Altix UV 100 com 64 núcleos Intel Xeon com 512 GB de RAM

3) 2 unidades com GPUS (24 processadores tradicionais e 4 placas Tesla M5020, com 512 núcleos de processamento)

Capacidade teórica de processamento: **22 Tflops!**

(ou 10^{12} operações de ponto flutuante por segundo...)



O cálculo numérico e a Astronomia

“(...) l’Astronomie nous apprenait à ne pas nous effrayer des grands nombres.”

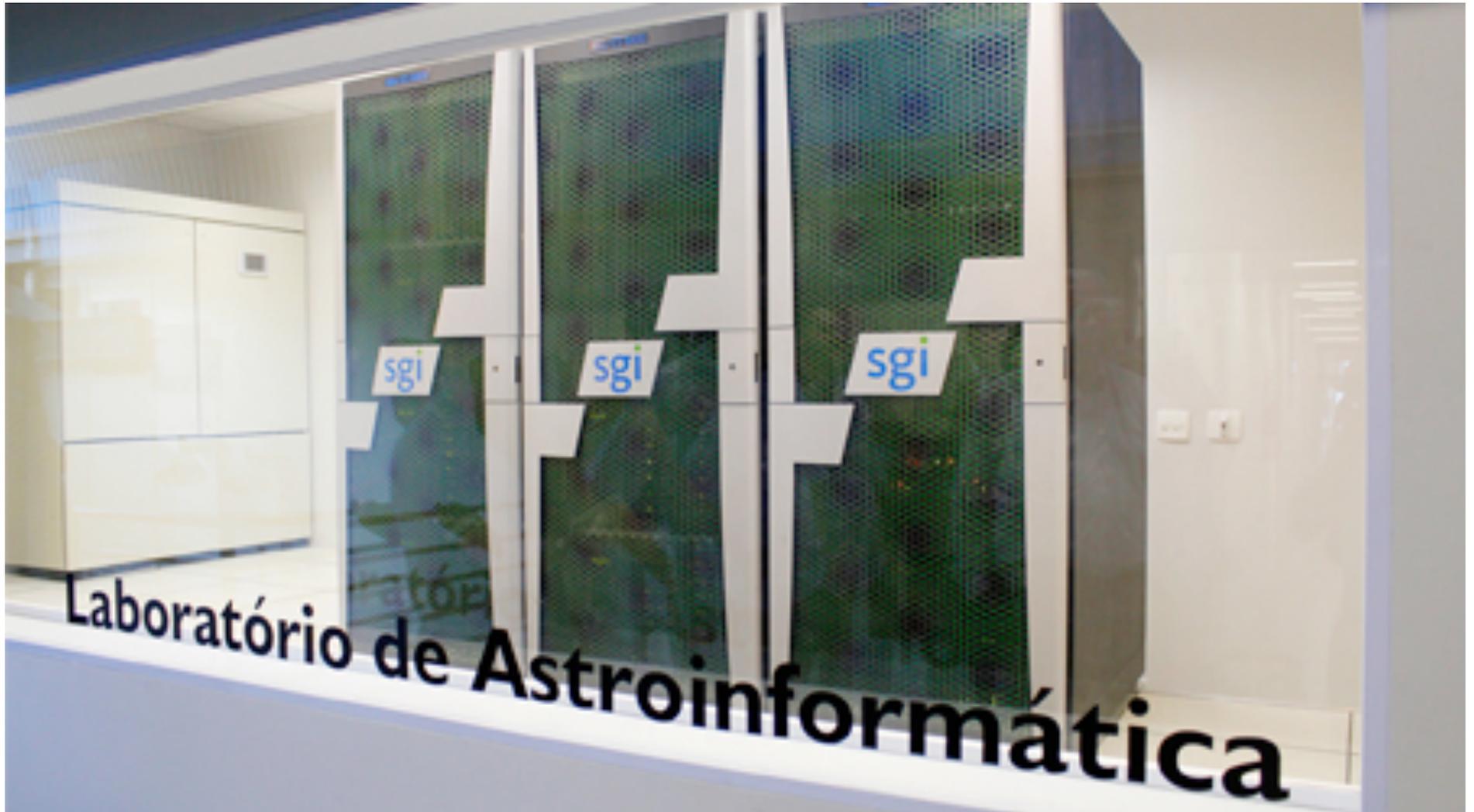
Henri Poincaré

No decorrer da história, a Astronomia teve várias vezes o importante papel de quebrar paradigmas da humanidade, ao ponto de Poincaré ter declarado que é ela a responsável por *“nos transformar em uma alma capaz de compreender a natureza”*

Alguns exemplos:

- Copérnico, Kepler
- Newton e o cálculo diferencial
- Einstein e a relatividade geral (avanço do periélio de Mercúrio)

Novo ramo da Astronomia: **astroinformática**



Um exemplo da biologia



in vivo

Um exemplo da biologia



in vivo



in vitro

Um exemplo da biologia

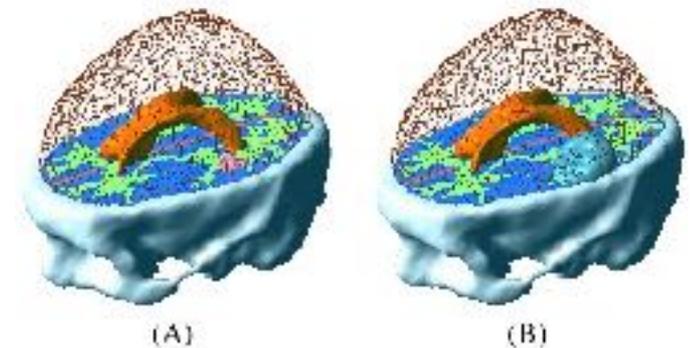
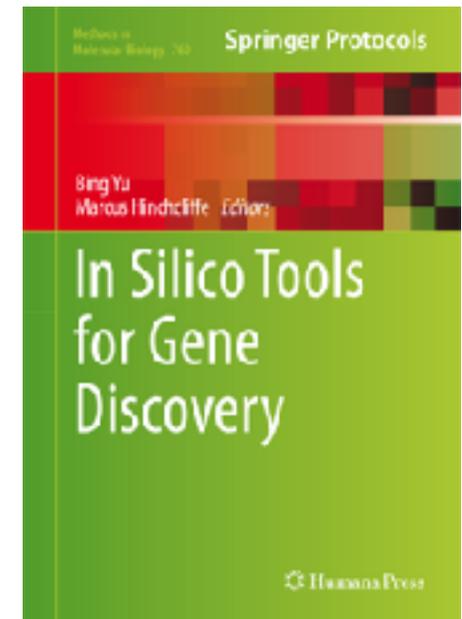


in vivo

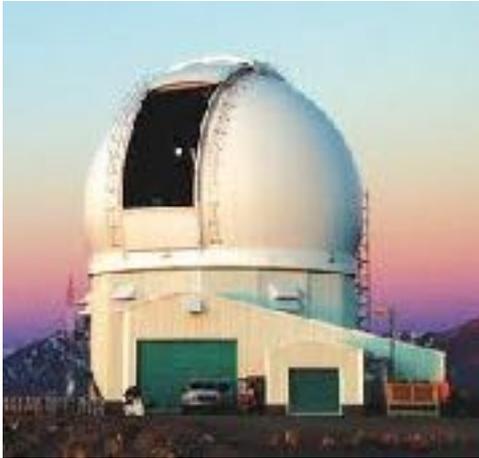


in vitro

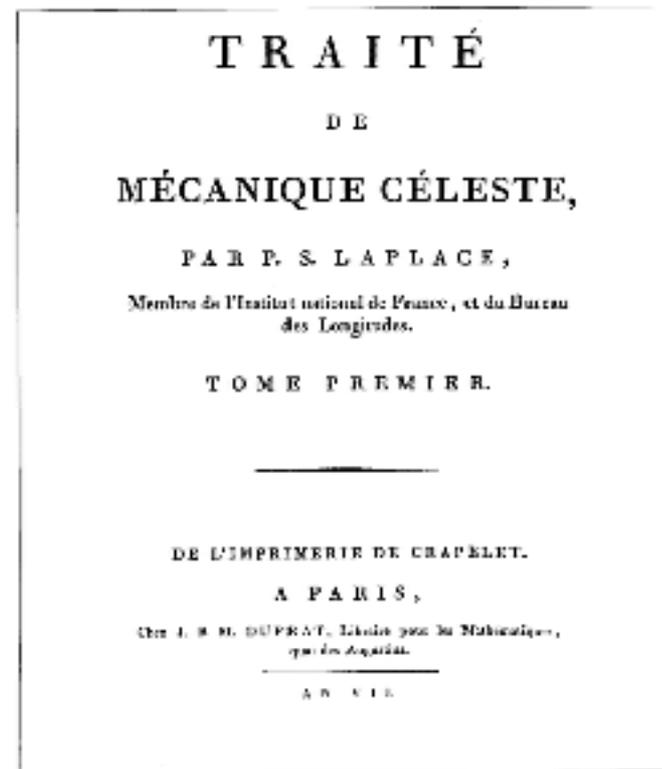
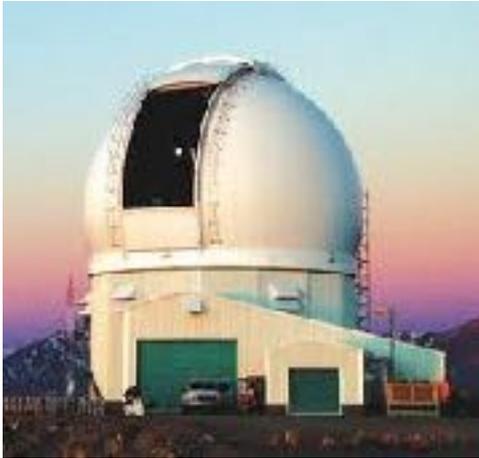
in silico



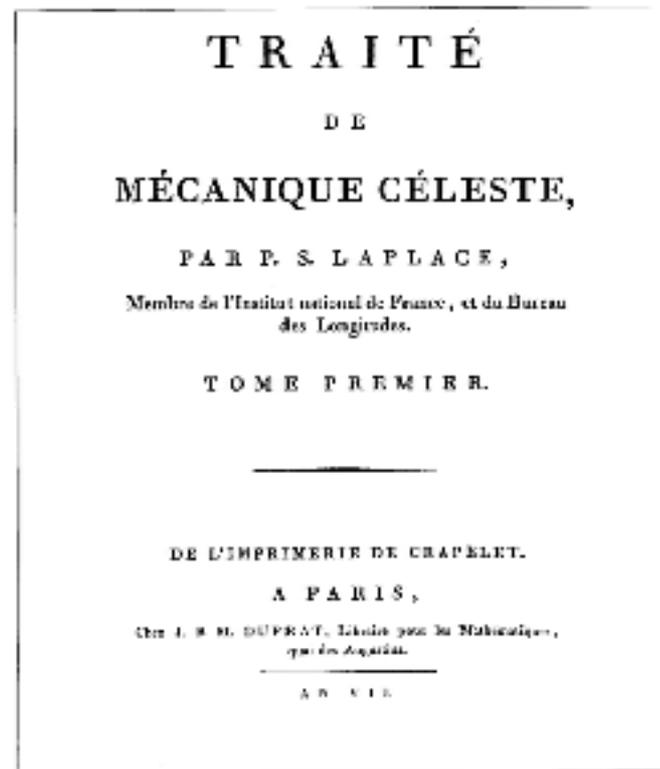
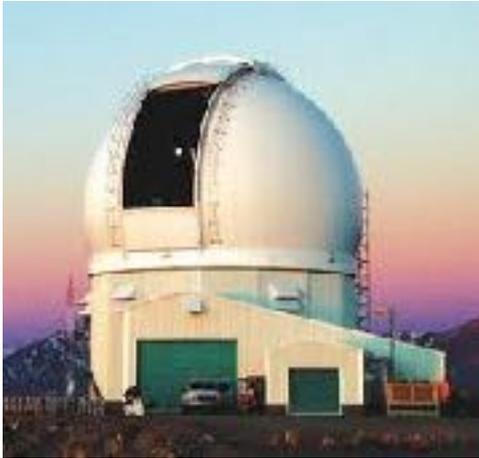
Da mesma forma, em astronomia...



Da mesma forma, em astronomia...



Da mesma forma, em astronomia...



O que essas abordagens têm em comum?

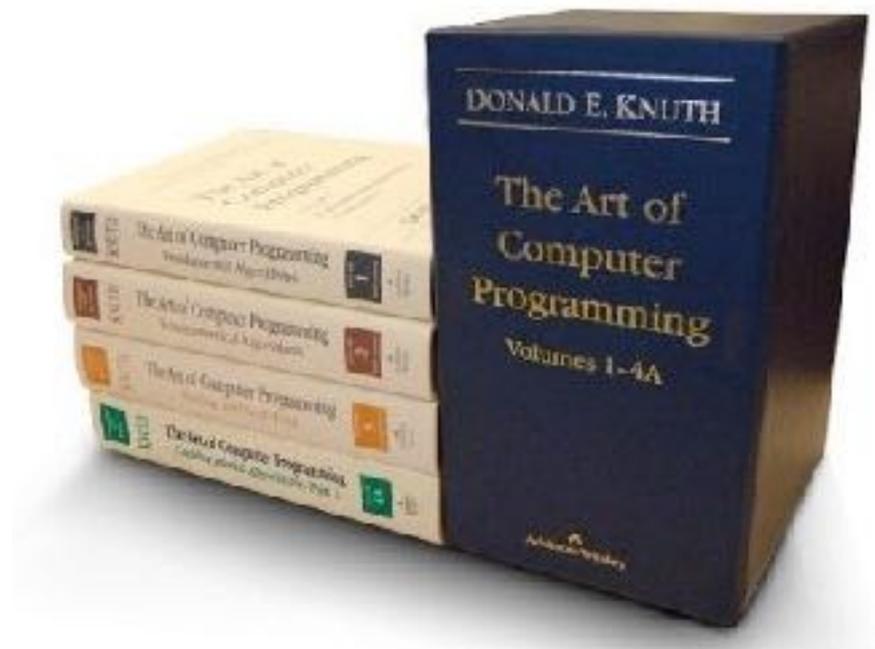
Data, data, data!

O que essas abordagens têm em comum?

Data, data, data!



Mas com os dados...



deve vir o software!

Para que aprender a escrever um código numérico?

A resposta é muito simples: grande parte da pesquisa moderna em Astronomia necessita de computadores. Alguns exemplos:

- **Simulações numéricas** feitas hoje são capazes de resolver problemas que antes eram absolutamente impossíveis. Elas são usadas para se comparar modelos com observações e desta forma testar teorias físicas (Ex: simulação do milênio)
- Dados astronômicos cada vez mais complexos requerem sofisticadas ferramentas de **visualização** para poderem ser estudados (ex: video)
- A **análise de dados** é uma atividade rotineira em astronomia. Frequentemente essa análise implica no uso de técnicas numéricas para se desempenhar uma série de tarefas.

Exemplo: simulação do milênio

Simulação numérica que usou 10 bilhões de partículas para estudar a distribuição de matéria em um espaço cúbico de 2 bilhões de anos-luz de lado.

Levou mais de um mês no principal computador do Instituto Max Planck de Garching, Alemanha (levaria um quarto do tempo no LAi...)

Objetivo científico: estudar a formação de estruturas em larga escala a partir do modelo Λ CDM.

Video...

Cálculo numérico e a pesquisa em Astronomia

Objetivo fundamental da pesquisa científica: confrontar **teorias físico-matemáticas** com as **observações** para se compreender os processos físicos que regem os sistemas naturais.

Uma pesquisa eficiente frequentemente requer:

- Uso de novas **ferramentas computacionais** para tornar os códigos rápidos o suficiente. Ex: Paralelismo, GPUs (AGA503, AGA510)
- **Ferramentas numéricas** para extração de informações dos dados observacionais (AGA503, AGA505 e AGA511). Ex: análise de períodos
- Implementação de ferramentas para **automação de tarefas** (AGA503, AGA505, AGA510 e AGA511). Ex: aplicação de técnicas de filtragem em milhares de espectros estelares
- Resolver problemas corriqueiros (AGA503, AGA505, AGA510 e AGA511).
- Etc.

Cientistas e seus *softwares*

Zeeya Merali 2010, Nature 467, p. 775

Uma das conclusões da pesquisa foi que cientistas, apesar de usarem e desenvolverem cada vez mais softwares em sua pesquisa, eles “não têm idéia de quão ruins são seus códigos”.

Razões:

- falta de treinamento formal (boa parte é auto-didata).
- cientistas não acompanham os desenvolvimentos recentes em computação.

As consequências são as piores possíveis (ver adiante)

...SCIENTISTS AND THEIR SOFTWARE

A survey of nearly 2,000 researchers showed how coding has become an important part of the research toolkit, but it also revealed some potential problems.

> **45%** said scientists spend more time today developing software than five years ago.”

> **38%** of scientists spend at least one fifth of their time developing software.

> Only **47%** of scientists have a good understanding of software testing.

> Only **34%** of scientists think that formal training in developing software is important.

Resolução numérica de problemas

O motivo para escrever um programa, qualquer programa, é **resolver um problema**. A tarefa de escrever um problema pode ser dividida em três partes:

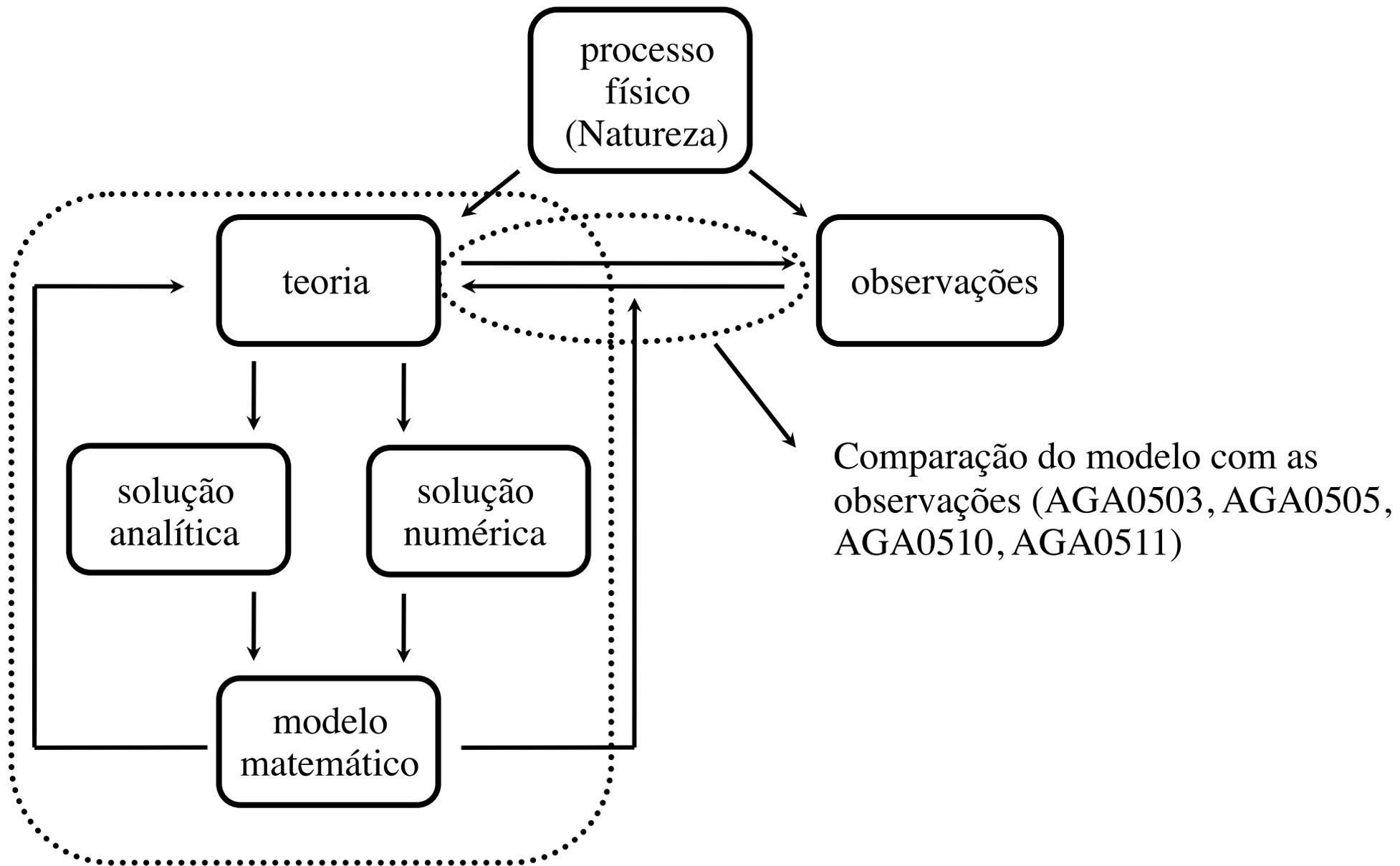
1. **Especificar o problema de forma clara** (difícil!)
2. Analisar o problema e **reduzi-lo em seus elementos fundamentais**
3. **Codificar o programa** de acordo com o plano desenvolvido no passo 2

Frequentemente há um quarto passo que, via de regra, é o mais difícil de todos:

4. Testar o problema exaustivamente, e repetir passos 2 e 3 até que o programa funcione corretamente em todas as situações previstas.

Método numérico: conjunto de regras escritas sob a forma de uma sequência de operações elementares que levam à uma solução do problema.

Em seu nível mais fundamental (ver capítulo 2), utiliza-se somente as quatro operações aritméticas $+$, $-$, \times , \div



Comparação do modelo com as observações (AGA0503, AGA0505, AGA0510, AGA0511)

Criação de um modelo matemático (física, cálculo, fismat, AGA0503, AGA0510)