

Resumo da Fase 2: Implementação do Solver Numérico

Status:  EM PROGRESSO (85% completo)

Objetivos da Fase 2

Implementar o solver numérico para resolver as EDPs de transporte 1D, incluindo:

- Solver de difusão com diferenças finitas
- Termos fonte de aquecimento (ECRH, ICRH, NBI, Ôhmico, Fusão)
- Integração temporal implícita
- Validação do balanço de energia

Realizações

1. Módulo de Termos Fonte de Aquecimento

Arquivo: `src/transport/heating_sources.py`

Implementamos a classe `HeatingSource` com todos os sistemas de aquecimento:

Sistemas Implementados:

a) ECRH (Electron Cyclotron Resonance Heating)

- Deposição gaussiana localizada
- Parâmetros: $\rho_{\text{peak}} = 0.4$, $\sigma = 0.15$
- Potência máxima: 20 MW
- Teste:  Conservação de potência verificada (20.00 MW)

b) ICRH (Ion Cyclotron Resonance Heating)

- Deposição gaussiana para íons
- Parâmetros: $\rho_{\text{peak}} = 0.5$, $\sigma = 0.2$
- Potência máxima: 30 MW
- Teste:  Conservação de potência verificada (30.00 MW)

c) NBI (Neutral Beam Injection)

- Perfil de deposição: $(1-\rho^2)/(1+\rho^2)$
- Fração eletrônica: 30% (6 MW)
- Fração iônica: 70% (14 MW)
- Potência máxima: 33 MW
- **Teste:** Conservação de potência verificada

d) Aquecimento Ôhmico

- Baseado na resistividade de Spitzer: $\eta \propto T^{-3/2}$
- Perfil de corrente parabólico
- **Teste:** Potência típica: ~5-6 MW para $I_p = 15$ MA

e) Aquecimento por Fusão (Partículas Alfa)

- Reatividade D-T: parametrização simplificada
- $\langle \sigma v \rangle \approx 1.1 \times 10^{-24} T^2 / (1 + (T/25)^2) \text{ m}^3/\text{s}$
- Energia das alfas: 3.5 MeV
- **Teste:** Potência realista: ~87 MW para $T=5$ keV, $n=5 \times 10^{20} \text{ m}^{-3}$

f) Transferência Colisional Elétron-Íon

- $Q_{ei} = (3 m_e n_e / \tau_e) (T_i - T_e)$
- **Teste:** Implementado (valores pequenos para $T_e \approx T_i$)

2. Módulo do Solver de Transporte (com bug a corrigir)

Arquivo: `src/transport/transport_solver.py`

Implementamos a classe `TransportSolver` com:

Funcionalidades:

a) Método de Diferenças Finitas Implícito

- Discretização espacial: grade uniforme em ρ
- Esquema temporal: Euler implícito
- Sistema tridiagonal resolvido com algoritmo de Thomas

b) Condições de Contorno

- Centro ($\rho=0$): derivada nula (simetria)
- Borda ($\rho=1$): valor fixo (Dirichlet)

c) Resolução das EDPs

- Equação de $T_e(\rho, t)$
- Equação de $T_i(\rho, t)$
- Equação de $n_e(\rho, t)$

d) Algoritmo de Thomas

- Solver tridiagonal eficiente
- Complexidade $O(N)$ ao invés de $O(N^3)$

3. Diagnóstico de Balanço de Energia

Arquivo: `src/transport/test_energy_balance.py`

Criamos um script de diagnóstico detalhado que revelou:

Resultados do Diagnóstico:

Potências de Aquecimento (verificadas):

- $P_{ECRH} = 10.00 \text{ MW}$ 
- $P_{ICRH} = 15.00 \text{ MW}$ 
- $P_{NBI,e} = 6.00 \text{ MW}$ 
- $P_{NBI,i} = 14.00 \text{ MW}$ 
- $P_{ohm} = 5.70 \text{ MW}$ 
- $P_{fusion} = 87.19 \text{ MW}$ 
- **Total: 137.88 MW**

Coeficientes de Transporte:

- $\chi_e(\rho=0) = 2.6 \text{ m}^2/\text{s}$
- $\chi_e(\rho=0.5) = 50 \text{ m}^2/\text{s}$ (limitado)
- $\chi_i(\rho=0) = 4.4 \text{ m}^2/\text{s}$
- $\chi_i(\rho=0.5) = 50 \text{ m}^2/\text{s}$ (limitado)

Estimativa de Perdas:

- Fluxo de calor total: ~7 MW
- Tempo de confinamento estimado: $\tau_E \approx 1.4 \text{ s}$

⚠ PROBLEMA IDENTIFICADO:

Após 1 passo de tempo ($dt = 1 \text{ ms}$):

- **ΔW observado: -38.77 MJ **

- ΔW esperado: +0.14 MJ ✓
- Razão: -281 (erro de ~28000%!)

A energia está **decaindo drasticamente** ao invés de aumentar, indicando um erro grave no solver.

Análise do Problema

Possíveis Causas:

1. Erro na discretização da equação de difusão

- Os coeficientes da matriz tridiagonal podem estar incorretos
- O termo fonte pode não estar sendo aplicado corretamente

2. Erro nas unidades

- Conversão entre MW/m³ e a equação de energia
- Fator de capacidade térmica (3/2) pode estar errado

3. Erro nas condições de contorno

- A condição de simetria no centro pode estar mal implementada
- A condição de Dirichlet na borda pode estar causando perdas excessivas

4. Erro no cálculo de V'

- O termo V' (derivada do volume) pode estar incorreto
- A normalização pode estar errada

Diagnóstico Detalhado:

Comparando com a equação teórica:

$$\frac{3}{2} n_e \frac{\partial T_e}{\partial t} = \frac{1}{V'} \frac{\partial}{\partial \rho} \left(V' n_e \chi_e \frac{\partial T_e}{\partial \rho} \right) + Q_e$$

Forma discretizada (Euler implícito):

$$\frac{3}{2} n_e \frac{T_e^{n+1} - T_e^n}{\Delta t} = \text{RHS}(T_e^{n+1}) + Q_e$$

O problema está provavelmente na implementação do RHS (lado direito).

Próximos Passos

Correções Necessárias:

1. Revisar a discretização do operador de difusão

- Verificar os coeficientes α, β, γ da matriz tridiagonal
- Conferir a implementação do termo $(1/V') \partial / \partial \rho (V' \times \partial T / \partial \rho)$

2. Verificar as unidades

- Conferir conversão de MW/m^3 para $\text{keV}/(10^{20} \text{ m}^{-3} \text{ s})$
- Verificar fator 3/2 na capacidade térmica

3. Simplificar o teste

- Testar com transporte desligado ($\chi = 0$) e apenas aquecimento
- Verificar se a energia aumenta corretamente

4. Implementar validação analítica

- Testar com solução analítica conhecida (ex: difusão pura)
- Verificar convergência com refinamento de malha

Testes Planejados:

1. Teste 1: Aquecimento sem transporte

- $\chi = 0, D = 0$
- Apenas termos fonte
- Esperado: $dW/dt = P_{\text{input}}$

2. Teste 2: Difusão pura

- Sem termos fonte
- Perfil inicial gaussiano
- Comparar com solução analítica

3. Teste 3: Estado estacionário

- Simular até equilíbrio
- Verificar $P_{\text{input}} = P_{\text{loss}}$

Estrutura de Arquivos Criados

Plain Text

```
simulator_1d/src/transport/
├── __init__.py
└── transport_coefficients.py
└── heating_sources.py
```

```
|── transport_solver.py  
└── test_energy_balance.py
```

⚠ (com bug)
✓

Comparação com Modelo 0D

| Aspecto | Modelo 0D | Modelo 1D (atual) |
|--------------------|-------------------|-----------------------|
| Aquecimento | Potências globais | Perfis radiais ✓ |
| Transporte | τ_E global | $X(\rho), D(\rho)$ ✓ |
| Solver | RK4 explícito | Implícito (com bug) ⚠ |
| Balanço de energia | Conservado ✓ | Não conservado ✗ |
| Tempo de simulação | ~0.5 s para 50 s | ~1 s para 0.1 s |

Conclusão Parcial

A Fase 2 está **85% completa**:

- ✓ Termos fonte implementados e validados
- ✓ Estrutura do solver implementada
- ✓ Diagnóstico de balanço de energia funcional
- ⚠ Solver com bug crítico no balanço de energia
- ✗ Validação numérica pendente

O próximo passo crítico é **corrigir o bug no solver** para garantir conservação de energia.

Tempo Estimado para Conclusão

- Correção do solver: 1-2 horas
- Testes de validação: 1 hora
- Documentação final: 30 minutos
- **Total: 2.5-3.5 horas**

Data: 23 de Dezembro de 2025

Autor: Sistema NPE-PSQ

Status: Fase 2 em Progresso (85%)