

UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE
LABORATÓRIO DE ENGENHARIA E EXPLORAÇÃO DE PETRÓLEO

PROJETO ENGENHARIA
SOFTWARE
SIMULADOR DE DIFUSÃO TÉRMICA 3D
TRABALHO DE CONCLUSÃO DE CURSO

Versão 1:
Nicholas de Almeida Pinto
Prof. André Duarte Bueno
Prof. Guilherme Rodrigues Lima

MACAÉ - RJ
Novembro - 2021

Sumário

1	Introdução	1
1.1	Escopo do problema	1
1.2	Objetivos	1
2	Especificação	3
2.1	Nome do sistema/produto	3
2.2	Especificação	3
2.2.1	Requisitos funcionais	4
2.2.2	Requisitos não funcionais	5
2.3	Casos de uso	5
2.3.1	Diagrama de caso de uso geral	5
2.3.2	Diagrama de caso de uso específico	5
3	Elaboração	7
3.1	Análise de domínio	7
3.2	Formulação teórica	8
3.3	Identificação de pacotes – assuntos	12
3.4	Diagrama de pacotes – assuntos	12

Capítulo 1

Introdução

No presente projeto de engenharia, desenvolveu-se o software Simulador de difusão térmica em objetos 3D, com paradigma orientado ao objeto, com o objetivo de implementar conceitos aprendidos nas disciplinas de fenômeno dos transportes, modelagem numérica e programação.

Dessa forma, a principal finalidade do simulador é fornecer o cálculo da temperatura ao longo do tempo, em um objeto genérico, com propriedades termofísicas também inseridas pelo usuário. Tornando-se uma ferramenta poderosa para o ensino de transferência de calor, cálculo numérico, programação orientada ao objeto, programação de multi-threads, e para o desenvolvimento de projetos de engenharia.

1.1 Escopo do problema

Troca de calor é um tema importantíssimo na indústria do Petróleo, sendo estudado e aplicado em absolutamente todas as etapas. Geólogos estudam a maturidade do óleo, engenheiros de reservatório estudam mecanismos térmicos para produzir mais óleo, engenheiros de planta de plataforma buscam formas de minimizar a perda energética devido às trocas de calor dos fluidos produzidos, engenheiros de refinaria buscam melhores controles de temperatura nas destilarias, e engenheiros de logística, melhores materiais para transportar o óleo/gás até o consumidor.

É importantíssimo resolver os problemas de engenharia citados, para diversos casos e situações, com alta precisão. Portanto, o que se propõe é um simulador de difusão térmica, que consegue resolver todos os casos possíveis, para qualquer temperatura, superfícies ou volumes, e para qualquer material. Tornando-se uma ferramenta prática para alunos e engenheiros.

1.2 Objetivos

O objetivo deste projeto são:

- Objetivo geral:
 - desenvolver um software capaz de simular a transferência de calor em qualquer superfície ou objeto, para qualquer material.
 - facilitar o entendimento e ensinamento de transferência de calor, modelagem numérica, programação orientada ao objeto e paralelismo.
- Objetivo específico
 - Simulação da transferência de calor de qualquer superfície.
 - Programação em C++, utilizando o paradigma de orientação ao objeto, e pacotes externos, permitindo modificações e adições no código fonte disponibilizado.
 - Métodos numéricos: solução de equações diferenciais da conservação de energia por meio de diferenças finitas, e desenvolvimento de algoritmo para resolver qualquer problema de fronteira.
 - Modelagem física e matemática do problema.
 - Simulação com dados de materiais obtidos em laboratório
 - Programação com paralelismo
 - Gerar gráfico e interface de usuário com software externo.
 - Resolver métodos iterativos.

Capítulo 2

Especificação

Apresenta-se neste capítulo do projeto de engenharia a concepção, a especificação do sistema a ser modelado e desenvolvido.

2.1 Nome do sistema/produto

Nome	Simulador de difusão térmica 3D
Componentes principais	Distribuição da temperatura em um objeto, ao longo do tempo, utilizando método numérico implícito.
Missão	Calcular a temperatura em objetos.

2.2 Especificação

Deseja-se desenvolver um software com interface gráfica amigável ao usuário, onde seja possível desenhar o objeto 3D, por meio de perfis, com o usuário escolhendo a temperatura e o material. A simulação é governada pela Equação da Difusão Térmica, a qual é modelada por diferenças finitas, pelo método BTCS, com fronteiras seladas.

Na dinâmica de execução, o usuário deverá escolher o tamanho do objeto, a temperatura, em qual perfil está desenhando, o material e suas propriedades termofísicas, e onde quer gerar gráficos para estudar. O usuário terá a liberdade para utilizar um dentre três métodos para obter as propriedades dos materiais: propriedades constantes, correlação e interpolação.

Os principais termos e suas unidades são listadas abaixo:

- Dados relativos ao material:

- c_p

- k

- ρ

- Dados relativos ao objeto
 - $\Delta x, \Delta y$ distância entre nodos, valor inicial: 1px=0.0026m [m];
 - Δz distância entre perfis, valor inicial: 0.05m [m];
 - T temperatura no nodo [K];
- Variáveis usadas na simulação:
 - i posição do nodo em relação ao eixo x;
 - k posição do nodo em relação ao eixo y;
 - g qual grid/perfil está sendo analisado;
 - t tempo atual;
 - ν número da iteração.

Após os desenhos do usuário e colocado o simulador para rodar, o simulador irá calcular iterativamente a temperatura em cada ponto, e só parará se o erro entre iterações for menor que um valor aceitável. Posteriormente, o desenho será atualizado, para mostrar a nova distribuição de temperatura, e plotará os gráficos com os novos valores.

O software será programado em C++, com paradigma orientado ao objeto, utilizando a biblioteca *Qt* para criar a interface do usuário, e *qcustomplot* para gerar os gráficos.

Para calcular as propriedades termofísicas dos materiais, são utilizados três modelos: propriedades constantes, por correlação e por interpolação.

2.2.1 Requisitos funcionais

Apresenta-se a seguir os requisitos funcionais.

RF-01	O usuário tem a liberdade de desenhar qualquer objeto 3D, escolhendo também sua temperatura em cada ponto.
RF-02	O usuário deverá ter liberdade para escolher o material em cada ponto do objeto.
RF-03	O usuário poderá salvar e/ou carregar dados da simulação.
RF-04	O usuário poderá salvar os resultados da simulação em um arquivo pdf.
RF-05	O usuário pode adicionar materiais no simulador, e escolher a forma de calcular suas propriedades termofísicas: constante, correlação ou interpolação.
RF-06	O usuário poderá escolher em qual ponto quer gerar gráficos para estudar a evolução da temperatura com o tempo.

RF-07	O usuário poderá comparar as propriedades termofísicas dos materiais..
--------------	--

2.2.2 Requisitos não funcionais

RNF-01	Os cálculos devem ser feitos utilizando-se o método numérico de diferenças finitas BTCS.
---------------	--

RNF-02	O programa deverá ser multi-plataforma, podendo ser executado em <i>Windows</i> , <i>GNU/Linux</i> ou <i>Mac</i> .
---------------	--

2.3 Casos de uso

Tabela 2.1: Exemplo de caso de uso

Nome do caso de uso:	Cálculo da temperatura
Resumo/descrição:	Cálculo da distribuição de temperatura em determinadas condições.
Etapas:	<ol style="list-style-type: none"> 1. Escolha da temperatura e do material 2. Desenhar o objeto desenhado 3. Escolher um ponto de estudo 4. Rodar a simulação e analisar resultados 5. Salvar objeto e resultados em pdf
Cenários alternativos:	Um cenário alternativo envolve uma entrada de propriedades de um metal obtidas em laboratório, escolher se essas propriedades vão ser calculadas por correlação ou interpolação.

2.3.1 Diagrama de caso de uso geral

O diagrama de caso de uso geral da Figura 2.1 mostra o usuário desenhando um objeto com material padrão do simulador, escolhendo um ponto de estudo, rodando a simulação, analisando os resultados e salvando o objeto e resultados em pdf.

2.3.2 Diagrama de caso de uso específico

O caso de uso específico na Figura 2.2 mostra um cenário onde o usuário quer utilizar os valores da condutividade térmica obtidos em laboratório. Ele deve montar um arquivo .txt com esses valores (a forma de criar esse arquivo é descrito no Apêndice B), e carregar no simulador.

O usuário terá a liberdade de comparar seu material com outros padrões do simulador, e escolhe-lo para o desenho do objeto.

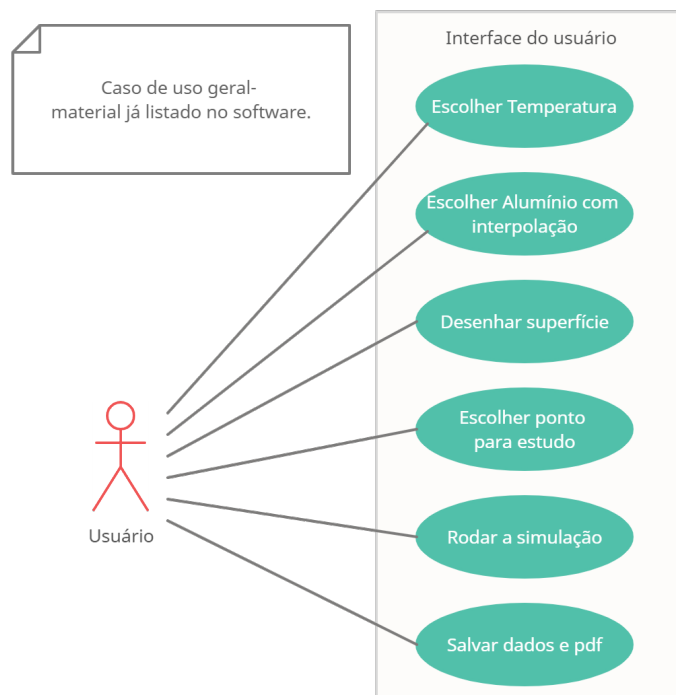


Figura 2.1: Diagrama de caso de uso – Caso de uso geral

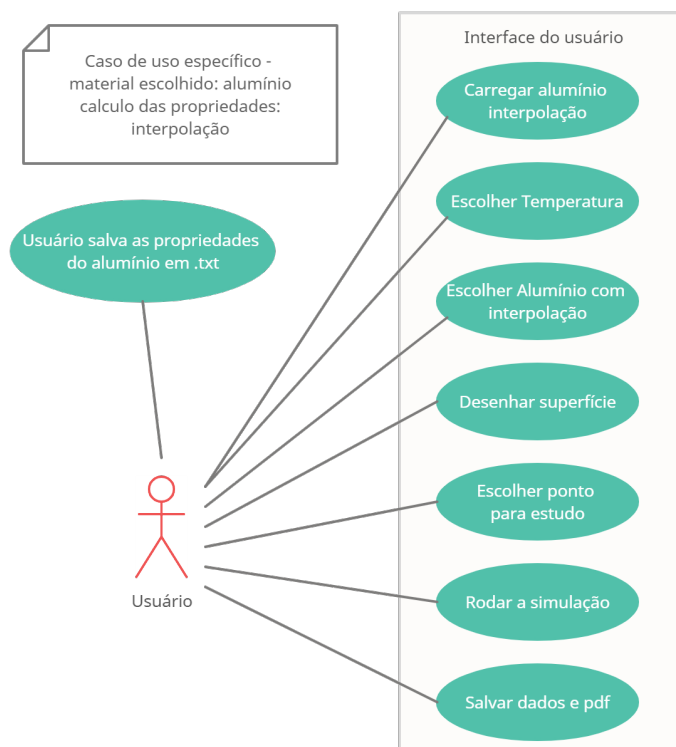


Figura 2.2: Diagrama de caso de uso específico

Capítulo 3

Elaboração

Depois da definição dos objetivos, da especificação do software e da montagem dos primeiros diagramas de caso de uso, a equipe de desenvolvimento do projeto de engenharia passa por um processo de elaboração que envolve o estudo de conceitos relacionados ao sistema a ser desenvolvido, a análise de domínio e a identificação de pacotes.

Na elaboração fazemos uma análise dos requisitos, ajustando os requisitos iniciais de forma a desenvolver um sistema útil, que atenda às necessidades do usuário e, na medida do possível, permita seu reuso e futura extensão.

3.1 Análise de domínio

Após estudo dos requisitos/especificações do sistema, algumas entrevistas, estudos na biblioteca e disciplinas do curso foi possível identificar nosso domínio de trabalho:

- Fenômeno dos transportes: área principal no qual o software foi desenvolvido. Utilizando equação do balanço de temperatura, propriedades termofísicas de materiais e condutividade térmica.
- Engenharia de petróleo: tópico principal para as simulações do software, especialmente a simulação de injeção térmica em reservatórios.
- Modelagem numérica computacional: desenvolvimento das equações diferenciais do balanço de temperatura, para que seja possível simular os mais diversos casos.
- Programação: utilização da linguagem C++ e paradigma orientado ao objeto, além de paralelismos para utilizar o máximo do poder de processamento e acelerar o software.
- Pacote de malhas: organiza o objeto desenhado em vetores.
- Pacote de simulação: resolve a equação da temperatura por métodos numéricos.

- Pacote de interpolação: utilizado para realizar interpolação com propriedades termofísicas dos materiais.
- Pacote de correlação: utilizado para realizar correlações com propriedades termofísicas dos materiais.
- Pacote de interface ao usuário: utilização da biblioteca Qt, para criar interface gráfica amigável.
- Pacote de gráficos: utilização da biblioteca qcustomplot, para montar os melhores gráficos para o problema.

3.2 Formulação teórica

A equação da difusão de calor (Cap. 2 Incropera####) pode ser estruturada a partir da Lei de Fourier. A equação geral da difusão de calor em meios tridimensionais cartesianos está na equação 3.1:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} = \frac{\rho c_p}{k} \frac{\partial T}{\partial t} \quad (3.1)$$

Onde ρ é a massa específica, c_p é a capacidade térmica, k é a condutividade térmica.

A modelagem pode ser feita por diferenças finitas atrasadas BTCS, onde cada derivada é representada abaixo:

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1,j,k}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i-1,j,k}^{n+1}}{\Delta x^2} \quad (3.2)$$

$$\frac{\partial^2 T}{\partial y^2} = \frac{T_{i,j+1,k}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j-1,k}^{n+1}}{\Delta y^2} \quad (3.3)$$

$$\frac{\partial^2 T}{\partial z^2} = \frac{T_{i,j,k+1}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j,k-1}^{n+1}}{\Delta z^2} \quad (3.4)$$

$$\frac{\partial T}{\partial t} = \frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta t} \quad (3.5)$$

Substituindo as diferenças finitas na equação geral:

$$\frac{T_{i+1,j,k}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i-1,j,k}^{n+1}}{\Delta x^2} + \frac{T_{i,j+1,k}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j-1,k}^{n+1}}{\Delta y^2} + \frac{T_{i,j,k+1}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j,k-1}^{n+1}}{\Delta z^2} = \frac{\rho c_p}{k} \frac{T_{i,j,k}^{n+1} - T_{i,j,k}^n}{\Delta t} \quad (3.6)$$

Onde a malha é homogênea na superfície, mas não entre os perfis, ou seja, $\Delta x = \Delta y \neq \Delta z$. Substituindo:

$$\begin{aligned} & \frac{T_{i+1,j,k}^{n+1} + T_{i,j+1,k}^{n+1} - 4T_{i,j,k}^{n+1} + T_{i-1,j,k}^{n+1} + T_{i,j-1,k}^{n+1}}{\Delta x} + \frac{T_{i,j,k+1}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j,k-1}^{n+1}}{\Delta z} \\ &= \frac{\rho c_p}{k} \frac{T_{i,j,k}^{n+1} - T_{i,j,k}^n}{\Delta t} \end{aligned} \quad (3.7)$$

Multiplicando pelo múltiplo comum:

$$\begin{aligned} & \frac{\Delta z (T_{i+1,j,k}^{n+1} + T_{i,j+1,k}^{n+1} - 4T_{i,j,k}^{n+1} + T_{i-1,j,k}^{n+1} + T_{i,j-1,k}^{n+1}) + \Delta x (T_{i,j,k+1}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j,k-1}^{n+1})}{\Delta x \Delta z} \\ &= \frac{\rho c_p}{k} \frac{T_{i,j,k}^{n+1} - T_{i,j,k}^n}{\Delta t} \end{aligned} \quad (3.8)$$

$$\begin{aligned} & \Delta z (T_{i+1,j,k}^{n+1} + T_{i,j+1,k}^{n+1} - 4T_{i,j,k}^{n+1} + T_{i-1,j,k}^{n+1} + T_{i,j-1,k}^{n+1}) + \Delta x (T_{i,j,k+1}^{n+1} - 2T_{i,j,k}^{n+1} + T_{i,j,k-1}^{n+1}) \\ &= \frac{\rho c_p \Delta x \Delta z}{k \Delta t} (T_{i,j,k}^{n+1} - T_{i,j,k}^n) \end{aligned} \quad (3.9)$$

$$\begin{aligned} & \Delta z T_{i+1,j,k}^{n+1} + \Delta z T_{i,j+1,k}^{n+1} - 4\Delta z T_{i,j,k}^{n+1} + \Delta z T_{i-1,j,k}^{n+1} + \Delta z T_{i,j-1,k}^{n+1} + \\ & \Delta x T_{i,j,k+1}^{n+1} - 2\Delta x T_{i,j,k}^{n+1} + \Delta x T_{i,j,k-1}^{n+1} = \frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^{n+1} - \frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^n \end{aligned} \quad (3.10)$$

Encontrando a seguinte equação:

$$\begin{aligned} & \Delta z T_{i+1,j,k}^{n+1} + \Delta z T_{i,j+1,k}^{n+1} + \Delta x T_{i,j,k+1}^{n+1} \\ & + \Delta z T_{i-1,j,k}^{n+1} + \Delta z T_{i,j-1,k}^{n+1} + \Delta x T_{i,j,k-1}^{n+1} \\ & - \left(4\Delta z + 2\Delta x + \frac{\rho c_p \Delta x \Delta z}{k \Delta t} \right) T_{i,j,k}^{n+1} \\ & = - \frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^n \end{aligned} \quad (3.11)$$

A Equação 3.11 é a geral da difusão de calor discretizada por diferenças finitas. Para implementar no software, é necessário modelar as fronteiras e, como é buscado uma generalização da equação para um objeto com superfície qualquer, será necessário entender alguns pontos.

Começamos entendendo a equação 3.11: na primeira linha, são concentrados pontos de temperaturas localizadas posteriormente ao estudado. Na segunda, são pontos anteriores ao estudado. Já o termo em parênteses na terceira linha, é o coeficiente para o termo estudado. Por fim, a última linha a direita da igualdade, é a temperatura no ponto estudado, mas no tempo anterior.

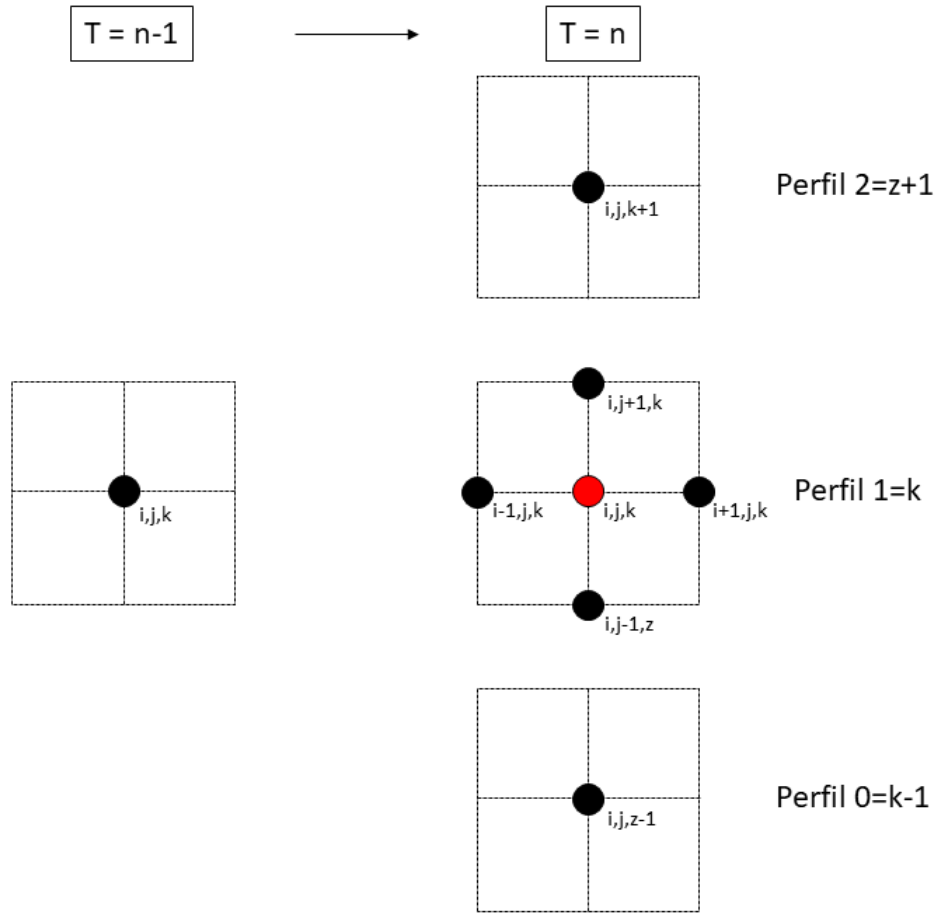


Figura 3.1: Malha utilizada para calcular um ponto de temperatura.

A seguir, serão realizadas duas etapas para finalizar a discretização. Primeiro a modelagem das fronteiras e, em seguida, a generalização da superfície de fronteira.

Primeira Parte

É discretizada a condição de contorno de Neumann, onde não há trocas com o meio externo, considerando que não há trocas com o ponto anterior no eixo x:

$$\frac{\partial T}{\partial x} = \frac{T_{i,j,k}^{n+1} - T_{i-1,j,k}^{n+1}}{\Delta x} = 0 \quad (3.12)$$

logo,

$$T_{i-1,j,k}^{n+1} = T_{i,j,k}^{n+1} \quad (3.13)$$

Todas as seis fronteiras possuem esse comportamento, então:

$$\begin{aligned}
T_{i-1,j,k}^{n+1} &= T_{i,j,k}^{n+1} \\
T_{i+1,j,k}^{n+1} &= T_{i,j,k}^{n+1} \\
T_{i,j-1,k}^{n+1} &= T_{i,j,k}^{n+1} \\
T_{i,j+1,k}^{n+1} &= T_{i,j,k}^{n+1} \\
T_{i,j,k-1}^{n+1} &= T_{i,j,k}^{n+1} \\
T_{i,j,k+1}^{n+1} &= T_{i,j,k}^{n+1}
\end{aligned} \tag{3.14}$$

Segunda Parte

Voltamos agora para a equação 3.11, faremos um caso onde há fronteira do lado esquerdo no eixo x (caso da primeira linha da equação 3.14):

$$\begin{aligned}
&\Delta z T_{i+1,j,k}^{n+1} + \Delta z T_{i,j+1,k}^{n+1} + \Delta x T_{i,j,k+1}^{n+1} \\
&+ \Delta z T_{i,j,k}^{n+1} + \Delta z T_{i,j-1,k}^{n+1} + \Delta x T_{i,j,k-1}^{n+1} \\
&- \left(4\Delta z + 2\Delta x + \frac{\rho c_p \Delta x \Delta z}{k \Delta t} \right) T_{i,j,k}^{n+1} \\
&= - \frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^n
\end{aligned}$$

Arrumando a equação:

$$\begin{aligned}
&\Delta z T_{i+1,j,k}^{n+1} + \Delta z T_{i,j+1,k}^{n+1} + \Delta x T_{i,j,k+1}^{n+1} \\
&+ \Delta z T_{i,j-1,k}^{n+1} + \Delta x T_{i,j,k-1}^{n+1} \\
&- \left(3\Delta z + 2\Delta x + \frac{\rho c_p \Delta x \Delta z}{k \Delta t} \right) T_{i,j,k}^{n+1} \\
&= - \frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^n
\end{aligned} \tag{3.15}$$

Podemos perceber que o termo $i-1,j,k$ sumiu da equação, e diminuiu o número 4 dentro do parênteses para 3, indicando que o número 4 é diretamente relacionado ao número de fronteiras da superfície xy, e o número 2, do eixo z.

Isso quer dizer que, caso exista uma fronteira na dimensão x ou y, esse termo deve ser anulado (condição de fronteira), e retirado 1 do total das 4 fronteiras e, caso exista uma fronteira no sentido de z, deve ser retirado a quantidade de fronteiras do total de 2. Portanto, é definido duas novas variáveis para o problema, nx e nz , onde $nx \in [0; 4]$ e $nz \in [0; 2]$

Pode-se ir além, e provar o caso onde há fronteiras em todos os sentidos:

$$\begin{aligned}
&\Delta z T_{i1,j,k}^{n+1} + \Delta z T_{i,j1,k}^{n+1} + \Delta x T_{i,j,k1}^{n+1} \\
&+ \Delta z T_{i,j,k}^{n+1} + \Delta z T_{i,j1,k}^{n+1} + \Delta x T_{i,j,k1}^{n+1} \\
&- \left(4\Delta z + 2\Delta x + \frac{\rho c_p \Delta x \Delta z}{k \Delta t} \right) T_{i,j,k}^{n+1} \\
&= - \frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^n
\end{aligned} \tag{3.16}$$

resultando em

$$T_{i,j,k}^{n+1} = \frac{\frac{\rho c_p \Delta x \Delta z}{k \Delta t}}{\frac{\rho c_p \Delta x \Delta z}{k \Delta t}} T_{i,j,k}^n \quad (3.17)$$

$$T_{i,j,k}^{n+1} = T_{i,j,k}^n \quad (3.18)$$

Ou seja, um ponto isolado no espaço não tem variação de temperatura.

Portanto, para ser possível implementar a equação discretizada 3.11 em C++, será utilizado:

$$T_{i,j,k}^{n+1} = \left(nx \Delta z + nz \Delta x + \frac{\rho c_p \Delta x \Delta z}{k \Delta t} \right)^{-1} \left[\frac{\rho c_p \Delta x \Delta z}{k \Delta t} T_{i,j,k}^n + \Delta z T_{i+1,j,k}^{n+1} + \Delta z T_{i,j+1,k}^{n+1} + \Delta x T_{i,j,k+1}^{n+1} + \Delta z T_{i-1,j,k}^{n+1} + \Delta z T_{i,j-1,k}^{n+1} + \Delta x T_{i,j,k-1}^{n+1} \right] \quad (3.19)$$

3.3 Identificação de pacotes – assuntos

- Pacote de malhas: organiza o objeto desenhado em vetores, facilita o acesso do simulador às propriedades de cada célula.
- Pacote de simulação: nela está presente o coração do simulador: o solver da equação da temperatura, discretizada por métodos numéricos, e resolvida por método iterativo.
- Pacote de interpolação: utilizado para realizar interpolação com propriedades termodinâmicas dos materiais, é acessado pelo simulador, e retorna as propriedades do material.
- Pacote de correlação: mesma função da linha acima, mas para método de correlação.
- Pacote de interface ao usuário: utilização da biblioteca Qt, para criar interface gráfica amigável. Fornece um ambiente onde o usuário pode enviar comandos para o simulador de maneira fácil, e apresenta os resultados.
- Pacote de gráficos: utilização da biblioteca qcustomplot, para montar os melhores gráficos para o problema. É solicitado ao pacote de malhas os resultados da temperatura. Está presente junto com o pacote de interface

3.4 Diagrama de pacotes – assuntos

Abaixo é apresentado o diagrama de pacotes (Figura 3.2).

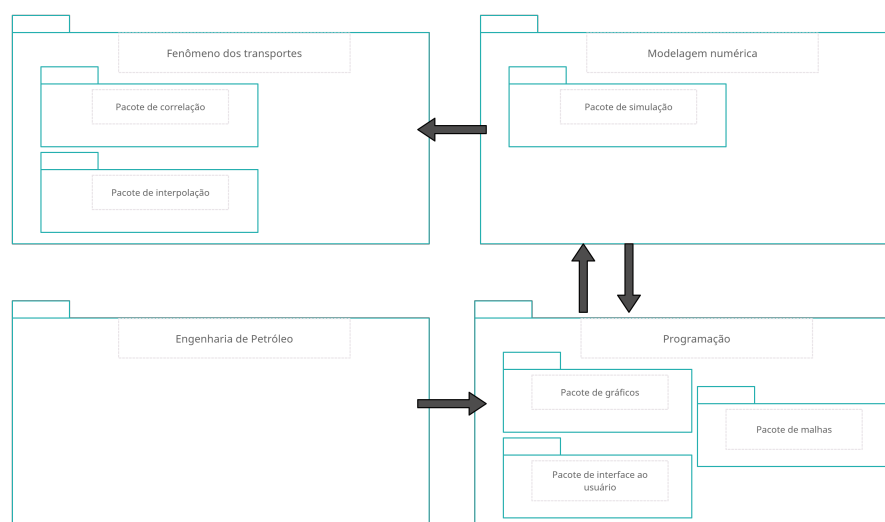


Figura 3.2: Diagrama de Pacotes

Referências Bibliográficas

Índice Remissivo

C

Casos de uso, 5

Concepção, 3

E

Elaboração, 7

especificação, 3