



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CENTRO DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA HIDRÁULICA E AMBIENTAL**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL**

**GUILHERME COSTA RODRIGUES NETO**

**MÉTODO LIVRE DE MALHA USANDO SOLUÇÃO FUNDAMENTAL APLICADO  
NA SIMULAÇÃO DO FLUXO DE ÁGUA SUBTERRÂNEO**

**FORTALEZA**

**2020**

GUILHERME COSTA RODRIGUES NETO

MÉTODO LIVRE DE MALHA USANDO SOLUÇÃO FUNDAMENTAL APLICADO NA  
SIMULAÇÃO DO FLUXO DE ÁGUA SUBTERRÂNEO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Civil – Recursos Hídricos da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Civil. Área de concentração: Recursos Hídricos.

Orientador: Prof. Dr. Marco Aurélio Holanda de Castro.

FORTALEZA

2020

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

R613m Rodrigues Neto, Guilherme Costa.  
Método livre de malha usando Solução Fundamental aplicado na Simulação do Fluxo de Água Subterrâneo / Guilherme Costa Rodrigues Neto. – 2020.  
79 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Civil: Recursos Hídricos, Fortaleza, 2020.  
Orientação: Prof. Dr. Marco Aurélio Holanda de Castro.

1. Método livre de malhas. 2. Método das Soluções Fundamentais. 3. Fluxo de água subterrâneo. 4. Simulação numérica. 5. Modelagem computacional. I. Título.

CDD 627

---

GUILHERME COSTA RODRIGUES NETO

MÉTODO LIVRE DE MALHA USANDO SOLUÇÃO FUNDAMENTAL APLICADO NA  
SIMULAÇÃO DO FLUXO DE ÁGUA SUBTERRÂNEO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Civil – Recursos Hídricos da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia Civil. Área de concentração: Recursos Hídricos.

Aprovada em: 07/02/2020.

BANCA EXAMINADORA

---

Prof. Dr. Marco Aurélio Holanda de Castro (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Paulo Roberto Lacerda Tavares  
Universidade Federal do Cariri (UFCA)

---

Prof. Lindemberg Lima Fernandes  
Universidade Federal do Pará (UFPA)

A Deus.

## AGRADECIMENTOS

À Universidade Federal do Ceará que através da infraestrutura de ensino e pesquisa do campus do Pici e do Departamento de Engenharia Hidráulica e Ambiental (DEHA) proporcionou a realização desta pesquisa.

À Fundação Cearense de Apoio à Pesquisa (FUNCAP) através da bolsa de mestrado 2018-2020.

Ao Corpo Docente do departamento, em especial ao meu orientador, por ter incentivado e oferecido auxílio através das aulas e do esclarecimento de questões relativas a este trabalho.

Aos pesquisadores do Laboratório de Hidráulica Computacional (LAHC) por compartilharem o ambiente de estudo, pelo auxílio na utilização das ferramentas computacionais e pela amizade construída ao longo do programa de Pós-Graduação.

À minha família por ter dado suporte ao meu crescimento pessoal e profissional e aos meus amigos, que sempre estiveram presentes não apenas em momentos de diversão, mas também em momentos de reflexão e de superação de dificuldades.

À meu pai, José Francisco, minha mãe, Maria do Socorro, minhas tias, Francisca Regina, Maria Zenilda e Djanira Paulo, minha irmã, Gabriella Costa, meus avôs, Egídio Francisco (em memória) e Antônio Bernardo e minhas avós, Itelvina Rodrigues (em memória) e Cecília Paulo por me conduzirem por um caminho de responsabilidade, seriedade, compromisso e respeito, que me permitiu chegar até aqui e permitirá, certamente, conquistar muito mais.

À Deus pela existência e pela natureza, Sua mais bela obra.

## RESUMO

O Método das Soluções Fundamentais (MSF) é um método numérico livre de malha usado para estimar valores funcionais de sistemas regidos por Equações Diferenciais Parciais em um domínio a partir de um conjunto de condições de contorno de um problema bem definido. Segundo essa metodologia, a Solução Fundamental do operador diferencial é aplicada a pontos do contorno relacionados a pontos de uma fronteira fictícia. Além de não precisar da construção de uma malha de nós ou elementos, o MSF apresenta convergência e estabilidade com relação à variação de uma fronteira fictícia que contém geometricamente o domínio analisado e que compõe uma das suas principais características. A escolha da localização dessa fronteira fictícia insere uma incerteza na aplicação do método. Portanto, neste trabalho, investigou-se a influência dessa localização na aplicação do MSF para determinar o nível piezométrico de pontos de um aquífero sedimentar sob a adoção das hipóteses de Dupuit-Forchheimer, incompressibilidade da água, homogeneidade do aquífero e fluxo em regime permanente, uma vez que essas hipóteses permitem aplicar o operador Laplaciano para descrever o fluxo de água subterrâneo. O MSF foi implementado em linguagem de programação Java e SciLAB, o modelo obtido foi aplicado em um aquífero real e comparado com o MODFLOW, que implementa o Método das Diferenças Finitas. Foram utilizados os dados de poços de observação da região da Área de Proteção Ambiental do córrego do Guariroba, que possui uma área de aproximadamente 360 km<sup>2</sup>, localizada no município de Campo Grande no Estado do Mato Grosso do Sul. Esses dados foram obtidos no trabalho de Cavazzana, Lastoria e Gabas (2019) e correspondem aos dados de um único dia de amostragem. Ao aplicar valores afastados em até 200 km e em até 10.000 km para a fronteira fictícia, em etapas de análise distintas, observou-se que os erros relativos às medidas de campo apresentaram estabilidade. Os resultados mostraram a convergência das equipotenciais de carga hidráulica obtidas através dos dois métodos e que a localização da fronteira fictícia que proporcionou os menores erros entre os valores numéricos e os de campo estava associada a uma distância de 1,7 vezes o raio da circunferência de mesma área da região analisada. Esse resultado aponta que o MSF é eficiente para aproximar níveis piezométricos em aquíferos sedimentares adotadas as hipóteses anteriores.

**Palavra-chave:** Método livre de malhas. Método das Soluções Fundamentais. Fluxo de água subterrâneo. Simulação numérica. Modelagem computacional.

## ABSTRACT

The Method of Fundamental Solutions (MFS) is a mesh-free numerical method used to estimate functional values of systems governed by Partial Differential Equations in a domain from a set of boundary conditions of a well-defined problem. According to this methodology, the Fundamental Solution of the differential operator is applied to contour points related to points on a fictitious border. In addition to not needing to build a mesh of nodes or elements, the MSF has convergence and stability with respect to the variation of a fictitious boundary that geometrically contains the analyzed domain and that makes up one of its main characteristics. The choice of the location of this fictitious border inserts an uncertainty in the application of the method. Therefore, in this work, the influence of this location on the application of MSF was investigated to determine the piezometric level of points in a sedimentary aquifer under the Dupuit-Forchheimer hypothesis, water incompressibility, aquifer isotropy and steady flow, since these hypotheses allow the Laplacian operator to be applied to describe the flow of groundwater. The MSF was implemented in Java and SciLAB programming language, the model obtained was applied in a real aquifer and compared with MODFLOW. Data from observation wells in the region of the Environmental Protection Area of the Guariroba stream, which has an area of approximately 360 km<sup>2</sup>, located in the municipality of Campo Grande in the state of Mato Grosso do Sul, were used. These data were obtained in the work of Cavazzana, Lastoria and Gabas (2019) and correspond to data from a single day of sampling. When applying values spaced up to 200 km and up to 10,000 km to the fictitious border, in different stages of analysis, it was observed that the errors related to field measurements showed stability. The results showed the convergence of the hydraulic load equipotentials obtained through the two methods and that the location of the fictitious border that provided the smallest errors between the numerical and field values was associated with a distance of 1.7 times the radius of the circumference of same area of the analyzed region. This result points out that the MSF is efficient to approximate piezometric levels in sedimentary aquifers adopted the previous hypotheses.

**Keywords:** Meshless Method. Method of Fundamental Solutions. Groundwater Flow. Numerical Simulation. Computational Modelling.



## LISTA DE FIGURAS

Figura 1	– Instante $t_0$ do deslocamento unidimensional do ponto material representado pelo círculo preto ao longo do eixo $s$ .....	20
Figura 2	– Domínio unidimensional de aplicação do truncamento da Série de Taylor para aproximação da primeira derivada de uma função por diferença progressiva, regressiva e central .....	27
Figura 3	– Placa metálica discretizada em pontos internos segundo uma malha quadrangular e com condições de contorno de Dirichlet aplicadas as suas fronteiras .....	28
Figura 4	– Resultados obtidos com o MDF para o problema enunciado na figura 3...	29
Figura 5	– Exemplo de discretização e de um domínio de análise bidimensional segundo uma geometria triangular dos elementos .....	30
Figura 6	– Esquemática de modelagem para MEF (esquerda) e para o MSF (direita) .....	32
Figura 7	– (A) Região do $\mathcal{R}^2$ delimitada por um contorno $\Gamma = \Gamma_1 \cup \Gamma_2$ e definida por um domínio $\Omega$ . (B) Fronteira fictícia que permite a aplicação do método e extingue a indeterminação da solução fundamental .....	35
Figura 8	– Esquemática do experimento realizado por Darcy .....	38
Figura 9	– Exemplificação de aquífero livre (vista lateral) onde as dimensões horizontais superam consideravelmente as dimensões verticais .....	39
Figura 10	– Elemento representativo do aquífero bidimensional .....	40
Figura 11	– Esquemática de um aquífero com uma região de fluxo livre e outra de fluxo confinado .....	43
Figura 12	– Fluxograma de geração de código por uma linguagem de programação genérica .....	45
Figura 13	– Algoritmo de aplicação do MSF .....	50
Figura 14	– Determinação do vetor normal do vértice de um polígono em função dos vértices anterior e posterior .....	51
Figura 15	– Janela principal do programa desenvolvido em Java para a aplicação do Método das Soluções Fundamentais .....	52
Figura 16	– Janela dos resultados da aplicação do MSF aos pontos de interesse .....	53
Figura 17	– APA do córrego Guariroba e poços de observação utilizados .....	55

Figura 18	–	Distribuição de precipitação e temperatura médias na região da APA do córrego do Guariroba entre os anos de 1996 e 2012 .....	56
Figura 19	–	Esquema da formação do Sistema Aquífero Baurú .....	57
Figura 20	–	Caracterização geológica e morfológica da APA do córrego do Guariroba .....	57
Figura 21	–	Fluxograma das etapas de obtenção dos resultados .....	59
Figura 22	–	Malha de pontos utilizada para calcular numericamente o valor da carga hidráulica em pontos internos ao domínio (Representações sem escala) ..	60
Figura 23	–	Gráfico do erro relativo aos valores de campo para valores de <i>off-set</i> variáveis .....	62
Figura 24	–	Curvas equipotenciais obtidas pelo MODFLOW (verde) e pelo MSF (vermelho) para o mesmo conjunto de dados iniciais .....	63
Figura 25	–	Erros relativos às leituras de campo para valores de off-set variando entre 100 e 10.000 km em passos de 100 km para cada um dos poços localizados no interior do domínio .....	65

## LISTA DE TABELAS

Tabela 1	– Classificação das EDPs de segunda ordem em função da relação $B2 - 4AC$ .....	22
Tabela 2	– Definição das condições de contorno complementares, onde $\phi$ corresponde à variável dependente, $a$ a uma constante, tal que $a \in \mathcal{R}$ , e $\Gamma$ significa a fronteira de um domínio de análise .....	23
Tabela 3	– Intervalo de valores de porosidade para alguns tipos de aglomerados de sedimentos e para alguns tipos de rochas sedimentares .....	41
Tabela 4	– Dados dos poços de observação utilizados na aplicação do MSF .....	54
Tabela 5	– Dados complementares. Tabela com a especificação dos valores adotados na simulação do problema no MODFLOW .....	59
Tabela 6	– Resumo dos resultados obtidos na primeira etapa da análise .....	63

## LISTA DE SÍMBOLOS

$\mathcal{R}$	Conjunto dos números reais
$\mathcal{R}^n$	Espaço euclidiano n-dimensional
$x_i$	Variável independente de uma função
$f(x_1, \dots, x_i, \dots, x_n)$	Valor de uma função definida por n variáveis
$A$	Coefficiente dos termos de uma EDP
$B$	Coefficiente dos termos de uma EDP
$C$	Coefficiente dos termos de uma EDP
$D$	Coefficiente dos termos de uma EDP
$E$	Coefficiente dos termos de uma EDP
$F$	Coefficiente dos termos de uma EDP
$G$	Coefficiente dos termos de uma EDP
$u$	Solução analítica de uma Equação Diferencial
$y^{[i]}, \frac{\partial^{[i]}y}{\partial x^{[i]}}$	Derivada de ordem i com relação à variável $x$
$\nabla^2 u$	Operador Laplaciano aplicado à função $u$
$t$	Variável tempo
$\phi$	Carga hidráulica
$\Gamma$	Fronteira de um domínio de análise
$\tilde{u}$	Solução aproximada/numérica
$\lambda$	Constantes do Método das Soluções Fundamentais
$A_r$	Área
$n$	Porosidade
$e$	Índice de vazios
$b$	Espessura da camada
$S_Y$	Produção específica
$S_s$	Armazenamento específico
$\mathbf{v}$	Vetor $v$

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	12
<b>1.1</b>	<b>Problemática e Contextualização</b> .....	12
<b>1.2</b>	<b>Justificativa</b> .....	14
<b>1.3</b>	<b>Objetivos</b> .....	14
<i>1.3.1</i>	<i>Objetivo Geral</i> .....	15
<i>1.3.2</i>	<i>Objetivos Específicos</i> .....	15
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b> .....	16
<b>2.1</b>	<b>Equações Diferenciais</b> .....	16
<i>2.1.1</i>	<i>Equações Diferenciais Ordinárias (EDO)</i> .....	19
<i>2.1.2</i>	<i>Equações Diferenciais Parciais (EDP)</i> .....	21
<i>2.1.3</i>	<i>Solução Fundamental de um Operador Diferencial</i> .....	24
<b>2.2</b>	<b>Métodos Numéricos</b> .....	25
<i>2.2.1</i>	<i>Métodos com malha</i> .....	26
<i>2.2.1.1</i>	<i>Método das Diferenças Finitas (MDF)</i> .....	26
<i>2.2.1.2</i>	<i>Método dos Elementos Finitos</i> .....	30
<i>2.2.2</i>	<i>Métodos livres de malha (Meshless)</i> .....	31
<i>2.2.2.1</i>	<i>Método das Soluções Fundamentais</i> .....	33
<b>2.3</b>	<b>Hidrogeologia</b> .....	36
<i>2.3.1</i>	<i>Conceitos Básicos</i> .....	37
<i>2.3.2</i>	<i>Aquíferos Sedimentares</i> .....	41
<b>2.4</b>	<b>Linguagens de programação</b> .....	44
<i>2.4.1</i>	<i>Java</i> .....	45
<i>2.4.2</i>	<i>SciLAB</i> .....	48
<b>3</b>	<b>METODOLOGIA</b> .....	49
<b>3.1</b>	<b>Desenvolvimento do algoritmo para aplicação do MSF</b> .....	49
<b>3.2</b>	<b>Implementação do algoritmo em linguagem Java e Scilab</b> .....	51
<b>3.3</b>	<b>Delimitação e caracterização da área de estudo</b> .....	54
<b>3.4</b>	<b>Avaliação da influência da localização da fronteira fictícia</b> .....	58
<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b> .....	61
<b>5</b>	<b>CONCLUSÕES</b> .....	66
	<b>REFERÊNCIAS</b> .....	67

<b>APÊNDICE A – CLASSE <i>MFSROOT</i> .....</b>	<b>72</b>
<b>APÊNDICE B – IMPLEMENTAÇÃO DO MSF EM SCILAB .....</b>	<b>74</b>

# 1 INTRODUÇÃO

Neste capítulo, serão apresentadas a problemática, a contextualização e a justificativa que inspiraram o desenvolvimento da pesquisa e serão expostos os objetivos geral e específicos.

## 1.1 Problemática e Contextualização

A modelagem de fluxo de água subterrânea é fundamental para a realização de intervenções relacionadas à gestão de recursos hídricos. Dispersão de poluentes, impactos associados à extração de água para consumo humano e para a agricultura e repartição de direitos de exploração de reservas de água subterrânea são alguns dos aspectos que evidenciam a importância da adoção de um modelo de fluxo robusto e preciso nos processos de tomada de decisão dos setores administrativos. (MIDDLEMIS; WALKER; PEETERS; RICHARDSON; HAYES; MOORE, 2019)

Os modelos de fluxo organizam os dados dos problemas analisados para se adequarem aos métodos numéricos e, após a aplicação, avaliam a qualidade dos resultados obtidos. Os métodos matemáticos contribuem com as ferramentas para a solução de sistemas de equações que estimam os valores numéricos das soluções analíticas que regem o fluxo de água subterrâneo. Assim, um mesmo problema pode ser resolvido por modelos diferentes, que podem ou não adotar métodos numéricos semelhantes, e, em geral, apresentam resultados convergentes. (ZIJJL; DE SMEDT; EL-RAWY; BATELAAN, 2018)

A partir do desenvolvimento do Método dos Elementos de Contorno (*Boundary Elements Method* – BEM), uma nova classe de métodos numéricos emergiu e tem mostrado maior flexibilidade na resolução de problemas relacionados ao fluxo de água subterrâneo e a outros fenômenos físicos. Esses métodos, chamados *meshless* ou *meshfree* são métodos livres de malha, o que significa que durante sua implementação não é necessário discretizar o domínio analisado em uma malha de elementos tal como nos métodos das diferenças finitas ou no método dos elementos finitos. Essa característica permite economizar tempo de processamento e recursos computacionais, além de proporcionar alternativas para a modelagem de problemas onde os métodos tradicionais não apresentam boa aplicabilidade. A relação entre as condições de contorno e as equações diferenciais parciais permite calcular numericamente os valores de pontos dentro do domínio, desde que sejam respeitados requisitos mínimos para que o problema seja considerado bem determinado (*well posed problem*). (LIU; GU, 2005)

Dentre os métodos livres de malha, o Método das Soluções Fundamentais (MSF) merece destaque porque é facilmente aplicado aos fenômenos físicos regidos pela equação de Laplace e de Poisson, pois a solução fundamental desses operadores diferenciais é conhecida. A aplicação desse método a outros operadores diferenciais, entretanto, encontra dificuldade durante a determinação da solução fundamental, o que nem sempre é uma tarefa simples.

Uma vez que o fluxo de água subterrâneo em aquíferos sedimentares pode ser descrito pela equação de Laplace em duas dimensões, adotadas as simplificações de Dupuit-Forchheimer, uma condição de fluxo não transiente (*steady-state flow*), assumindo-se a isotropia da formação geológica e a incompressibilidade da água (HAITJEMA, 1995), o MSF se torna uma ferramenta apropriada para tratar o problema da determinação das direções de fluxo e das equipotenciais de carga hidráulica. (WANG; ZHENG, 2015). Contudo, são encontradas na literatura poucas aplicações desse método numérico na resolução de problemas de fluxo de água subterrâneo. A maior parte das publicações se restringe à análise matemática e à proposição de alternativas para modificar algumas de suas propriedades básicas ou para a resolução de outros problemas físicos onde a equação de Laplace também se aplica.

A proposta da aplicação do MSF na resolução de problemas de fluxo de água subterrâneo preenche essa lacuna na comunidade científica. Para tanto, esse método precisa ser analisado na intenção de identificar as limitações que ele proporciona na simulação desse tipo de problema, e em especial, o caso de um aquífero sedimentar confinado ou livre.

Muitos trabalhos investigaram as propriedades do MSF, tais como a influência da escolha dos pontos de fonte e do formato da fronteira fictícia na precisão dos resultados obtidos (BOGOMOLNY, 1985, GOLBERG; CHEN, 1998) e o condicionamento das matrizes geradas, bem como métodos para evitar o mal condicionamento proporcionado por fronteiras fictícias muito afastadas do domínio (LIU, 2012, YOUNG; TSAI; CHEN; FAN, 2006, CHEN; GOLBERG; CHO, 2006, GOLBERG; CHEN, 1998). Alguns pesquisadores propuseram alternativas para contornar a necessidade de uma fronteira fictícia como Chen e Wang (2010), que sugeriram a substituição da solução fundamental nos elementos diagonais de (12) por uma técnica de interpolação inversa chamando o novo método de *Singular Boundary Method (SBM)* e Liu e Sarler (2013), que contornaram o problema da singularidade distribuindo pontos ao longo de um disco ao redor do ponto de indeterminação no que foi denominado *Non-Singular Method of Fundamental Solutions (NMFS)*. (BARRERO-GIL, 2012). Gu, Fan e Xu (2019) observaram que o MSF pode ser aplicado a sub-regiões de um domínio, o que o torna eficiente para a implementação de problemas que requerem soluções localizadas ou adjacentes e extensas do ponto de vista da geração de matrizes; esse método foi chamado de *Localized Method of*



*Fundamental Solutions (LMFS)*. Outros trabalhos verificaram a possibilidade de resolver problemas não homogêneos e dependentes do tempo (GOLBERG; CHEN; MULESHKOV, 1999 e ALVES; CHEN, 2005). Esses últimos proporcionam alternativas promissoras para a análise transiente do fluxo subterrâneo através do MSF. Wang e Zheng (2015) evidenciaram a possibilidade de sobrepor as soluções do MSF para simular um poço de bombeamento, encarado como uma inomogeneidade na aplicação do operador Laplaciano e obteve resultados mais precisos do que os obtidos com outros métodos livres de malha.

## **1.2 Justificativa**

A aplicação do MSF na análise do fluxo de água subterrâneo permitirá incorporar as vantagens de um método livre de malha na determinação das linhas de fluxo e equipotenciais de carga hidráulica, que, segundo Liu e Gu (2005), são maior precisão e adaptabilidade em comparação aos métodos tradicionais. Todos esses resultados publicados fornecem um embasamento teórico diverso que pode ser utilizado para inspirar a confecção futura de um modelo de fluxo de água subterrâneo que utilize o Método das Soluções Fundamentais como ferramenta numérica. Contudo, até que um modelo possa ser proposto, é necessário investigar o comportamento do MSF em exemplos piloto, principalmente a influência do posicionamento da fronteira fictícia, para que os ajustes, caso sejam necessários, sejam documentados e preparados a priori a fim de obter resultados úteis e convergentes com as observações de campo e com outros modelos de fluxo de água de subterrâneo. Para tanto, essa pesquisa buscou aplicar o MSF em um exemplo real a fim de simular o fluxo de água subterrâneo em uma região onde foi possível levantar dados suficientes para a montagem de um problema bem determinado; e em seguida comparou os resultados obtidos com os resultados proporcionados por outros métodos e com os valores de campo em pontos de validação.

## **1.3 Objetivos**

A conclusão desta pesquisa esteve associada à realização do seguinte objetivo geral e dos seguintes objetivos específicos.

### ***1.3.1 Objetivo Geral***

- Verificar a influência da localização da fronteira fictícia na aplicação do MSF na simulação do fluxo de água subterrâneo em uma área de estudo real.

### ***1.3.2 Objetivos Específicos***

- Elaborar o algoritmo da aplicação do MSF para dados de poços de observação;
- Implementar o algoritmo em uma linguagem de programação (Java e Scilab);
- Determinar e caracterizar uma área de estudo onde o modelo será aplicado;
- Aplicar o MSF segundo a modelagem proposta;
- Verificar a convergência do resultado com o resultado proporcionado pelo Método das Diferenças Finitas através do MODFLOW.

## 2 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão apresentados os conceitos e o conjunto das informações que fundamentam o desenvolvimento desta pesquisa. Foram consultados livros e artigos publicados nos principais periódicos nacionais e internacionais com a intenção de apresentar o conteúdo de maneira sistemática e concisa. Esta revisão aborda os seguintes temas: Equações Diferenciais, Métodos Numéricos, com destaque para o Método das Soluções Fundamentais, os fundamentos de hidrogeologia e uma introdução às linguagens de programação orientadas a objeto.

### 2.1 Equações Diferenciais

As equações diferenciais são uma ferramenta matemática que permite representar fenômenos físicos através das suas principais propriedades. Associar um sistema natural a uma equação diferencial significa estabelecer os critérios que descrevem seu funcionamento e quantificam os valores das suas variáveis, o que também é conhecido como modelagem matemática. Esse tipo de equação trabalha com o conceito de derivadas. As derivadas equivalem à taxa de variação de uma determinada variável dependente com relação a uma ou mais variáveis independentes dentro de uma relação funcional. (LEITHOLD, 1994). Em uma função de uma única variável definida na equação (1), sua derivada com relação a essa variável é igual ao limite expresso na equação (2), se ela for contínua nesse ponto.

$$f: \mathcal{R} \rightarrow \mathcal{R} \mid y = f(x) \mid x, y \in \mathcal{R} \quad (1)$$

$$\frac{\partial f(x)}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} \quad (2)$$

No caso de uma função de “n” variáveis (equação 3), sua derivada com relação a uma dessas variáveis  $x_i$  pode ser expressa como o limite do valor funcional para um  $\Delta x_i$  que tende a zero, mantendo-se constantes os valores das outras variáveis, o que dá origem ao conceito de derivada parcial (equação 4).

$$f: \mathcal{R}^n \rightarrow \mathcal{R} \mid y = f(x_1, \dots, x_i, \dots, x_n) \mid x_i, y \in \mathcal{R} \quad (3)$$

$$\frac{\partial f(x_1, \dots, x_i, \dots, x_n)}{\partial x_i} = \lim_{\Delta x_i \rightarrow 0} \frac{f(x_1, \dots, x_i + \Delta x_i, \dots, x_n) - f(x_1, \dots, x_i, \dots, x_n)}{\Delta x_i} \quad (4)$$

As equações diferenciais apresentam pelo menos uma derivada em sua composição. Exemplos de equações diferenciais podem ser obtidos através da análise de problemas simples como o caso de um corpo em queda livre na atmosfera, esvaziamento de um recipiente, movimento de um pêndulo simples, fluxo de água em meios porosos, dentre outros.

É possível classificar as equações diferenciais em função do seu tipo, da sua ordem, da sua linearidade e da sua homogeneidade.

a) Quanto ao tipo:

Quanto ao tipo, as equações diferenciais podem ser *ordinárias* ou *parciais*. As equações diferenciais ordinárias (EDO) são equações diferenciais onde as derivadas existentes na equação são as derivadas de uma única variável dependente com relação a uma variável independente. (LEITHOLD, 1994). As equações (5) e (6) são exemplos de EDO's.

$$\frac{\partial y(x)}{\partial x} + x = 0 \quad (5)$$

$$\frac{\partial^2 y(x)}{\partial x^2} + 5 \frac{\partial y(x)}{\partial x} + 3y = f(x) \quad (6)$$

As equações diferenciais parciais (EDP) são equações diferenciais onde as derivadas parciais de uma variável dependente aparecem na equação, isso é, as derivadas dessa variável dependente com relação a mais de uma variável independente, equações (7) e (8).

$$\alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2} = \frac{\partial u(x, t)}{\partial t} \quad (7)$$

$$\frac{\partial f(x, y, z)}{\partial x} + \frac{\partial f(x, y, z)}{\partial y} + \frac{\partial f(x, y, z)}{\partial z} = 0 \quad (8)$$

As equações diferenciais também podem ser agrupadas em sistemas de equações diferenciais, onde a solução envolve a determinação de mais de uma função resposta. (BOYCE; DIPRIMA, 2010).

b) Quanto a ordem:

As equações diferenciais podem ser classificadas quanto a derivada de maior ordem que aparece na equação. A ordem de uma derivada é o índice de derivação que está indicado na sua representação escrita. (LEITHOLD, 1994). Na sequência da equação (9), a ordem das derivadas cresce de 0 a “n”. A derivada de ordem “zero” é a própria função.

$$y(x), \frac{\partial y}{\partial x}, \frac{\partial^2 y}{\partial x^2}, \dots, \frac{\partial^i y}{\partial x^i}, \dots, \frac{\partial^n y}{\partial x^n} \quad (9)$$

Nesse sentido, é possível classificar as equações (5) e (8) como equações diferenciais de primeira ordem e as equações (6) e (7) como equações diferenciais de segunda ordem.

c) Quanto a linearidade:

Uma equação diferencial é dita linear se ela for uma combinação linear das derivadas que a compõem. No caso de uma EDO, ela pode ser classificada como linear se puder ser reescrita no seguinte formato: equação (10), onde os índices  $n$  equivalem à maior ordem encontrada na equação e  $g(t)$  é uma função qualquer.

$$a_0(t)y^{[n]} + a_1(t)y^{[n-1]} + \dots + a_n(t)y = g(t) \quad (10)$$

Uma EDP de segunda ordem, por sua vez, pode ser classificada como linear se puder ser reescrita conforme a equação (11), onde as letras  $A, B, C, D, E, F$  e  $G$  são constantes ou funções independentes de  $u$  e de suas variáveis.

$$A \frac{\partial^2 u}{\partial x^2} + B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + D \frac{\partial u}{\partial x} + E \frac{\partial u}{\partial y} + Fu = G \quad (11)$$

Uma equação diferencial é linear se, em cada um dos seus termos, a variável dependente ou suas derivadas ocorrerem apenas uma vez, elevadas ao expoente 1 e não compuserem

uma função composta. Seguindo essa definição é possível determinar que as equações (5), (6), (7) e (8) são lineares, enquanto as equações (12) e (13) são não-lineares.

$$u(x) \frac{\partial^2 u(x)}{\partial x^2} = 0 \quad (12)$$

$$\frac{\partial f}{\partial x} + \cos f = g(t) \quad (13)$$

d) Quanto a homogeneidade:

A classificação quanto a homogeneidade se refere ao termo que não está relacionado às derivadas, o termo independente da equação. (LEITHOLD, 1994). Se esse termo for igual a zero, como nas equações (5), (7), (8) e (12), a equação diferencial pode ser classificada como homogênea. Caso contrário, como nas equações (6) e (13), ela é dita não-homogênea.

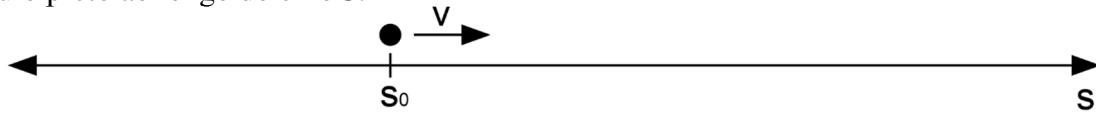
Solucionar uma equação diferencial significa encontrar a função  $y(x)$  que satisfaça a identidade (14). Encontrar essa função resposta nem sempre é possível através de meios analíticos e, dependendo do problema, podem existir infinitas ou nenhuma solução. Uma das alternativas utilizadas para aproximar os valores funcionais em determinados pontos são os métodos numéricos.

$$F(x, y, y^{[1]}, y^{[2]}, \dots, y^{[n]}) = 0 \quad (14)$$

### **2.1.1 Equações Diferenciais Ordinárias (EDO)**

Através da resolução de EDOs é possível analisar fenômenos simples e unidimensionais e determinar algumas das principais fórmulas sobre as quais a física clássica apoia seu estudo. Um dos exemplos clássicos é o estudo do deslocamento de um ponto material ao longo de um eixo retilíneo. De acordo com a figura 1, um ponto material ocupa uma posição  $s_0$  no instante  $t_0$  localizado sobre seu eixo de deslocamento.

**Figura 1** – Instante  $t_0$  do deslocamento unidimensional do ponto material representado pelo círculo preto ao longo do eixo  $s$ .



Fonte: O Autor.

Sabendo que a taxa de variação da sua posição no tempo é igual a sua velocidade, é possível escrever a equação (15), que descreve o fenômeno físico e pode ser usada para prever a posição  $s$  do ponto material para qualquer instante desejado.

$$\frac{\partial s}{\partial t} = v \quad (15)$$

Resolvendo analiticamente a equação (15), obtém-se a equação (16), que representa um conjunto infinito de funções que satisfazem as condições até então estabelecidas, onde  $s_i$  representa uma constante de integração arbitrária associada à posição do ponto material.

$$s(t) = s_i + vt \quad (16)$$

Para que o problema possa ser considerado bem determinado (*well-posed problem*), isso é, para que admita apenas uma única solução, é necessário o fornecimento de informações complementares, apenas assim será possível determinar o valor de  $s_i$  e distinguir, dentre as infinitas soluções possíveis, aquela que descreve com exatidão o deslocamento do ponto material analisado. Em função do tipo de informação complementar disponibilizado, a resolução da EDO pode ser associada a dois tipos de problemas: *problemas de valor inicial* ou *problemas de valores de contorno*.

Nos problemas de valor inicial, a informação complementar está associada a um único ponto ou a um conjunto de pontos a partir do qual é possível aplicar a solução genérica e determinar a solução particular, já nos problemas de valores de contorno, a informação complementar está associada aos limites físicos da área analisada. Em problemas unidimensionais, como no caso do deslocamento de um ponto material ao longo de um eixo, o conhecimento de uma posição inicial e de uma velocidade inicial (para o caso de um movimento uniformemente variado) em um determinado instante é suficiente para encontrar a solução particular. Nos outros tipos de problemas, é necessário um conjunto de informações sobre o contorno, isso é, sobre as fronteiras da região que se está analisando, tal como uma posição de

partida e outra de chegada, na situação unidimensional ilustrada anteriormente, ou as características de uma fronteira de análise, em uma situação multidimensional associada principalmente aos espaços 2D ou 3D.

### 2.1.2 Equações Diferenciais Parciais (EDP)

Analogamente às EDOs, as equações diferenciais parciais (EDPs) apresentam em sua composição as derivadas parciais de uma função qualquer com relação as suas variáveis independentes. Problemas multidimensionais, portanto, mais complexos, podem ser descritos por esse tipo de equação. As EDPs de segunda ordem são um caso específico de EDPs que podem ser generalizadas, em sua forma linear, no formato da equação (11). As EDPs de segunda ordem encontram aplicações em diversas áreas de estudo e podem descrever o funcionamento de diversos fenômenos físicos. Exemplos de equações diferenciais parciais de segunda ordem são a equação de Laplace (17), que descreve o potencial de uma variável desconhecida, neste exemplo, em um domínio bidimensional, a equação da condução do calor em uma dimensão (18) e a equação unidimensional da onda (19), onde  $a$  é uma constante e  $t$  é a variável tempo.

$$\nabla^2 u(x, y) = \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 0 \quad (17)$$

$$\frac{\partial u}{\partial t} = a^2 \frac{\partial^2 u}{\partial x^2} \quad (18)$$

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2} \quad (19)$$

As EDPs podem ser classificadas em homogêneas ou não-homogêneas em função do valor de  $G$  na equação (11). Se  $G$  for igual a zero, então a equação é chamada de homogênea, caso contrário, não-homogênea. Uma das principais propriedades de uma equação diferencial linear homogênea é a de que duas soluções conhecidas podem ser combinadas linearmente para satisfazer a mesma equação diferencial. Esse é o chamado princípio da superposição. Em equações não-homogêneas, uma solução para o problema pode ser encontrada ao adicionar uma solução particular, que representa a inhomogeneidade, a uma solução geral da parte homogênea da equação diferencial. (CHOW, 2003).



As equações diferenciais parciais lineares podem ser classificadas quanto à relação existente entre seus coeficientes. A relação  $B^2 - 4AC$  dos coeficientes da equação (11) permitem classificá-las segundo a tabela 1. De acordo com Chapra e Canale (2008), essa classificação, fundamentada na aplicação do método das características, agrupa as diferentes equações diferenciais parciais em contextos semelhantes que sugerem metodologias de resolução similares. Segundo essa classificação, a equação de Laplace (17) é do tipo elíptica, a equação da condução do calor (18) é parabólica e a equação da onda (19) é hiperbólica.

**Tabela 1** – Classificação das EDPs de segunda ordem em função da relação  $B^2 - 4AC$ .

$B^2 - 4AC$	Categoria
$< 0$	Elíptica
$= 0$	Parabólica
$> 0$	Hiperbólica

Fonte: Adaptado de Chapra e Canale (2008).

Apesar da classificação anterior sugerir estratégias de resolução para EDPs agrupadas segundo a relação existente entre seus coeficientes, elas ainda estão sujeitas a condições de contorno ou a valores iniciais que permitam a determinação de uma única solução para cada problema modelado, isso é, desde que essas condições complementares sejam suficientes para caracterizá-lo como um problema bem determinado.

Essas condições complementares para o caso de problemas de valores de contorno podem ser de alguns tipos básicos (JAZAYERI; WERNER, 2019):

- Condições de Dirichlet;
- Condições de Neumann;
- Condições de Robin;
- Condições de Cauchy e
- Condições Mistas.

As condições de Dirichlet, também conhecidas como condições do tipo 1 ou condições essenciais, são condições de contorno associadas ao valor funcional da variável dependente da equação diferencial que se deseja solucionar (tabela 2). (BREBBIA, 1978). No caso da modelagem do fluxo de água subterrâneo, as condições de Dirichlet correspondem aos valores de carga hidráulica ou carga de pressão medidos em poços de observação ou a valores de carga hidráulica constantes como os atribuídos a corpos hídricos superficiais, rios e lagos, por exemplo. No caso da modelagem da difusão de calor, essa condição se refere à temperatura

medida em algum ponto ou em uma das faces do objeto onde esse fluxo está sendo estudado. Assim como os demais tipos de condições complementares, as condições de Dirichlet podem corresponder a valores iniciais ou a valores de contorno dependendo da natureza da análise. (JAZAYERI; WERNER, 2019).

As condições de Neumann também são conhecidas como condições do tipo 2 ou condições naturais e estão associadas aos valores da primeira derivada da variável dependente (tabela 2). (BREBBIA, 1978). No caso do fluxo de água subterrâneo, esse tipo de condição representa valores de fluxo, assim, uma região impermeável na fronteira de um domínio de análise é interpretada como uma condição de contorno de Neumann com valor igual a zero e pontos associados a poços de bombeamento ou de injeção, a condições de Neumann com valor igual à respectiva vazão. (JAZAYERI; WERNER, 2019).

**Tabela 2** – Definição das condições de contorno complementares, onde  $\phi$  corresponde à variável dependente,  $a$  a uma constante, tal que  $a \in \mathcal{R}$ , e  $\Gamma$  significa a fronteira de um domínio de análise.

Condição complementar	Definição
Condição de Dirichlet	$\phi = \begin{cases} \phi_i, & \text{carga constante} \\ f(x) x \in \Gamma, & \text{carga especificada} \end{cases}$
Condição de Neumann	$\frac{\partial \phi}{\partial n} = \begin{cases} q_i, & \text{vazão de bombeamento} \\ f(x) x \in \Gamma, & \text{fluxo especificado} \end{cases}$
Condição de Robin	$\frac{\partial \phi}{\partial n} + a\phi = f(x) x \in \Gamma$
Condição de Cauchy	$\begin{cases} \phi = f(x) \\ \frac{\partial \phi}{\partial n} = g(x) \end{cases}  x \in \Gamma$
Condição Mista	$\begin{cases} \phi = f(x) x \in \Gamma_1 \\ \frac{\partial \phi}{\partial n} = g(x) x \in \Gamma_2 \end{cases}; \text{onde } \Gamma_1 \cup \Gamma_2 = \Gamma$

Fonte: Adaptado de Jazayeri e Werner (2019).

As condições de Robin, ou condições do tipo 3, são uma combinação linear dos dois tipos de condições complementares anteriores, Dirichlet e Neumann (tabela 2). Na modelagem do fluxo de água subterrâneo, esse tipo de condição complementar representa o fluxo através de uma fronteira de análise (vazão associada a um rebaixamento) ou regiões próximas a rios e lagos onde o nível dos leitos é inferior à carga hidráulica adjacente. As condições de Cauchy (tabela 2) são condições que representam valores funcionais e da primeira derivada da variável dependente analisada simultaneamente. São casos de difícil interpretação e não representam significado direto na modelagem do fluxo de água de subterrâneo. As condições mistas são aquelas onde uma fronteira é determinada por regiões adjacentes e não-sobrepostas onde em

cada uma ocorre um tipo de condição complementar. (JAZAYERI; WERNER, 2019). Esse tipo de situação é comum na modelagem de problemas de valores de contorno aplicados à modelagem de fluxo de água subterrânea, onde um determinado domínio pode conter em seus limites uma região impermeável (condição de Neumann igual a zero) e em outro, uma carga hidráulica constante (condição de Dirichlet), por exemplo.

### 2.1.3 Solução Fundamental de um Operador Diferencial

A solução fundamental de um operador diferencial deriva do conceito de equação de Green. Segundo esse conceito, uma solução fundamental  $F$  (equação 20) é uma função que aplicada ao operador diferencial resulta na função Delta de Dirac, onde  $L$  representa o operador diferencial,  $F$ , a solução fundamental e  $\delta(t - t')$  a função Delta de Dirac, definida na equação (21), aplicada ao ponto  $t' \in \mathcal{R}^n$ . Observe que na equação (20) a subtração  $(t - t')$  significa a distância euclidiana entre esses dois pontos para problemas no  $\mathcal{R}^2$  (BOAS, 1966).

$$LF = \delta(t - t') \quad (20)$$

Uma interpretação física da definição anterior permite associar a solução fundamental a um estímulo unitário, um impulso, aplicado ao operador diferencial no ponto  $t'$ . São apresentados a função Delta de Dirac na equação (21) e uma propriedade na equação (22). Essa propriedade é consequência do fato de a integral da função Delta de Dirac ser igual a 1 ao longo da região em que a sua variável está definida, desde que o intervalo de integração contenha o ponto  $t'$ . (BOAS, 1966).

$$\delta(t - t') = \begin{cases} 0, & \text{se } t \neq t' \\ \infty, & \text{se } t = t' \end{cases} \quad (21)$$

$$f(t_0) = \int f(t)\delta(t - t_0)dt \quad (22)$$

Aplicando a definição (20) ao operador Laplaciano (17), obtêm-se (23).

$$\nabla^2 F = \delta(t - t'), \quad t \in \mathcal{R}^2 \quad (23)$$

Para duas dimensões, a solução fundamental do operador Laplaciano é igual a equação (24), onde  $r$  é o vetor distância entre os pontos  $t$  e  $t'$  da definição da equação (21). Essa função satisfaz o operador Laplaciano para qualquer ponto dentro do domínio e pode ser utilizada através da equação (20) para encontrar uma solução particular de uma situação não-homogênea aplicando a propriedade (22), onde  $f(t)$  representa a inhomogeneidade.

$$F(r) = -\frac{1}{2\pi} \ln r \quad (24)$$

## 2.2 Métodos Numéricos

Historicamente, a aplicação dos métodos numéricos ocorreu simultaneamente ao desenvolvimento dos primeiros computadores. Uma vez que esses métodos buscam aproximar a solução de equações diferenciais através da resolução de sistemas de equações numéricas, sua praticidade só pode ser usufruída a partir do surgimento das primeiras máquinas eletrônicas a partir das décadas de 60 e 70. Nessa época, dois métodos principais foram objeto da confecção de extenso material científico. Toda a euforia da novidade computacional em torno do Método das Diferenças Finitas (MDF) e do Método dos Elementos Finitos (MEF) tornou possível a aplicação da tecnologia do processamento digital em diversas áreas. Da indústria aeroespacial à construção civil, várias atividades de pesquisa e econômicas sofreram os impactos e experimentaram melhorias nas técnicas e no conhecimento dos seus principais objetos de estudo. Obter valores de variáveis dependentes a partir de relações algébricas baseadas nas equações diferenciais que regem os fenômenos físicos transformou-se em uma prática viável. No campo da simulação hidrogeológica, os resultados dessas pesquisas proporcionaram a construção de vários softwares de modelagem de fluxo de água subterrâneo, dentre os quais o MODFLOW, que atualmente continua sendo referência na modelagem de aquíferos através da aplicação do MDF. Outros métodos numéricos ainda em fase de desenvolvimento e de investigação, tais como os métodos apoiados na teoria dos Elementos de Contorno (*Boundary Element Methods* – BEM) e métodos livres de malha, em contraste ao MDF e ao MEF, foram pesquisados com ênfase teórica em detrimento de aplicações práticas. (BREBBIA; TELLES; WROBEL, 1984, MCDONALD; HARBAUGH, 2005).

### 2.2.1 Métodos com malha

Os métodos com malha são métodos numéricos que aplicam princípios de cálculo sobre a geometria do problema, pois fundamentam sua análise na propagação de certas condições de contorno à elementos discretizados dentro do domínio de análise. O MDF e o MEF utilizam estratégias diferentes para performar a aproximação dos valores de uma variável dependente através da continuidade geométrica que existe entre os subelementos (nós, vértices ou elementos). Tal característica permite a simulação de corpos sólidos e a observação de propriedades tais como a distribuição de tensões e deformações, a propagação de calor e de ondas, por exemplo. Contudo, para Liu e Gu (2005), a necessidade de uma malha que represente a continuidade do domínio limita sua aplicabilidade em situações de estresse geométrico, análises adaptativas e simulação de crescimento de falhas e estudos de ruptura.

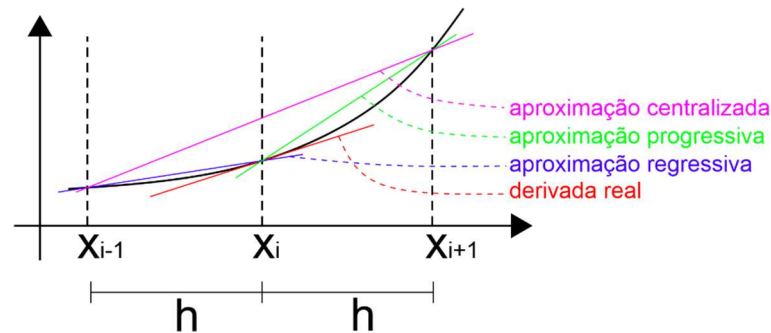
Apesar dessas limitações, os métodos com malha permitiram conquistas em diversas áreas do conhecimento. A construção de edifícios com formas curvilíneas, a economia de combustível em aeronaves mais eficientes, modelos de fluxo de água subterrânea em aquíferos sedimentares, modelos de difusão de poluentes, modelos de previsão de características atmosféricas, dentre outros, só foram possíveis a partir da extensiva investigação sobre a aplicabilidade desses métodos em sistemas físicos reais. A seguir são apresentados brevemente os princípios de funcionamento desses dois métodos que ilustram a classe dos métodos tradicionais com malha.

#### 2.2.1.1 Método das Diferenças Finitas (MDF)

O Método das Diferenças Finitas é um dos métodos numéricos mais conhecidos por ter sofrido intensa pesquisa e desenvolvimento a partir das décadas de 60 e 70 e por ser fácil de implementar e de interpretar fisicamente. Ele fundamenta-se na aproximação das derivadas por uma equação de diferenças algébricas aplicada a pontos discretos de um domínio de análise. (LEVEQUE, 2007). A figura 2 ilustra o exemplo em um domínio unidimensional.

Essa equação de diferenças algébricas é obtida através do truncamento da série de Taylor. São apresentadas a diferença ascendente ou progressiva (equação 25), calculada com  $f(x_{i+1})$ , e descendente ou regressiva (equação 26), calculada com  $f(x_{i-1})$ , onde  $O(h)$  é o termo que compõe o erro de truncamento.

**Figura 2** – Domínio unidimensional de aplicação do truncamento da série de Taylor para aproximação da primeira derivada de uma função por diferença progressiva, regressiva e central.



Fonte: O Autor.

Nas equações (25) e (26), a série de Taylor foi truncada depois da primeira derivada. Outras formulações permitem o truncamento em derivadas de maior índice, influenciando na redução do erro sob a penalidade de aumentar o tempo de processamento computacional devido o acréscimo de termos algébricos. Segundo Quadros e Bortoli (2009), o truncamento depois da segunda derivada é suficiente para atender os critérios de acurácia nos principais problemas de engenharia.

$$f(x_{i+1}) = f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2!}h^2 + \frac{f^{[3]}(x_i)}{3!}h^3 + \dots + \frac{f^{[n]}(x_i)}{n!}h^n + R_n$$

$$f'(x_i) = \frac{f(x_i + h) - f(x_i)}{h} + O(h) \quad (25)$$

$$f(x_{i-1}) = f(x_i) - f'(x_i)h - \frac{f''(x_i)}{2!}h^2 - \frac{f^{[3]}(x_i)}{3!}h^3 - \dots - \frac{f^{[n]}(x_i)}{n!}h^n - R_n$$

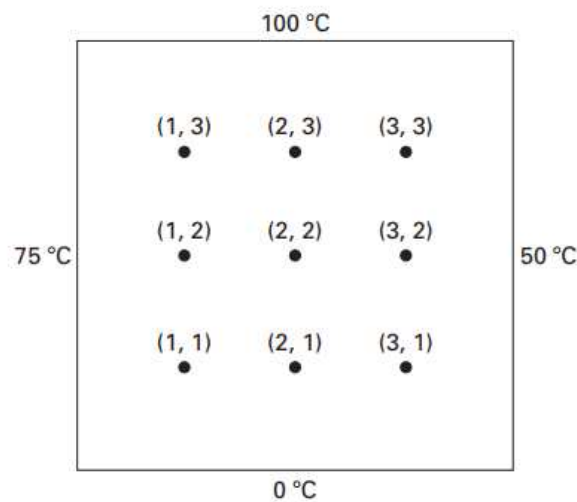
$$f'(x_i) = \frac{f(x_i - h) + f(x_i)}{h} - O(h) \quad (26)$$

Outra forma de aproximar o valor da primeira derivada em  $x_i$  é através da diferença centrada, que subtrai a série de Taylor progressiva da regressiva e permite escrever a expressão aproximada da primeira derivada como na equação (27), onde o termo  $O(h^2)$  assume ordem 2.

$$f'(x_i) = \frac{f(x_i + h) - f(x_i - h)}{2h} - O(h^2) \quad (27)$$

Para ilustrar a aplicação do MDF, tomemos a equação de Laplace que descreve a distribuição em equilíbrio da temperatura em uma placa metálica de pequena espessura onde as bordas possuem temperatura fixada e correspondem, portanto, às condições de contorno que caracterizam o problema como bem determinado (figura 3).

**Figura 3** – Placa metálica discretizada em pontos internos segundo uma malha quadricular e com condições de contorno de Dirichlet aplicadas às suas fronteiras.



Fonte: Chapra e Canale (2008).

A determinação das equações das diferenças centradas para as segundas derivadas parciais, analogamente ao que foi demonstrado para a primeira derivada anteriormente, de uma função  $T(x, y)$  permite escrever as equações (28) e (29). (CHAPRA; CANALE, 2008).

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} \quad (28)$$

$$\frac{\partial^2 T}{\partial y^2} = \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} \quad (29)$$

Substituindo as equações (28) e (29) no operador Laplaciano (equação 17) e fazendo  $\Delta x = \Delta y$ , pois é uma malha quadriculada, é possível obter a equação da diferença de Laplace (equação 30).

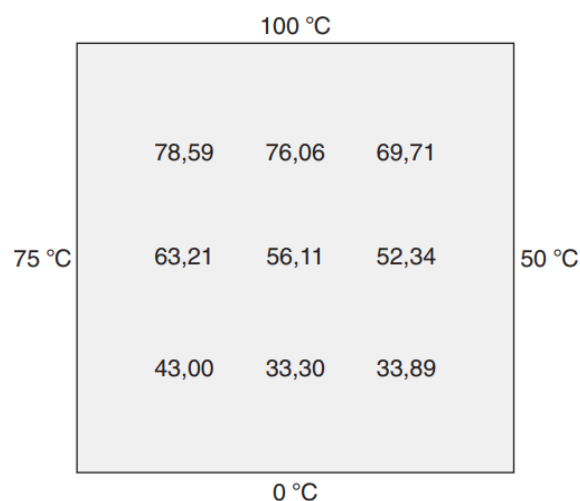
$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0 \quad (30)$$

Cada um dos nós da figura 3 possui sua própria versão da equação (30). Encontrar o valor aproximado segundo esse método para cada um dos pontos discretizados significa resolver o sistema de equações (31), que também pode ser tratado matricialmente. No sistema de equações (31), os índices 0 e 4 estão associados às condições de contorno.

$$\begin{aligned}
 T_{2,1} + T_{0,1} + T_{1,2} + T_{1,0} - 4T_{1,1} &= 0 \\
 T_{3,1} + T_{1,1} + T_{2,2} + T_{2,0} - 4T_{2,1} &= 0 \\
 T_{4,1} + T_{2,1} + T_{3,2} + T_{3,0} - 4T_{3,1} &= 0 \\
 T_{2,2} + T_{0,2} + T_{1,3} + T_{1,1} - 4T_{1,2} &= 0 \\
 T_{3,2} + T_{1,2} + T_{2,3} + T_{2,1} - 4T_{2,2} &= 0 \\
 T_{4,2} + T_{2,2} + T_{3,3} + T_{3,1} - 4T_{3,2} &= 0 \\
 T_{2,3} + T_{0,3} + T_{1,4} + T_{1,2} - 4T_{1,3} &= 0 \\
 T_{3,3} + T_{1,3} + T_{2,4} + T_{2,2} - 4T_{2,3} &= 0 \\
 T_{4,3} + T_{2,3} + T_{3,4} + T_{3,2} - 4T_{3,3} &= 0
 \end{aligned} \tag{31}$$

A aplicação do MDF nesse problema através da solução do sistema de equações (31) proporciona os resultados da figura 4. (CHAPRA; CANALE, 2008).

**Figura 4** – Resultados obtidos com o MDF para o problema enunciado na figura 3.



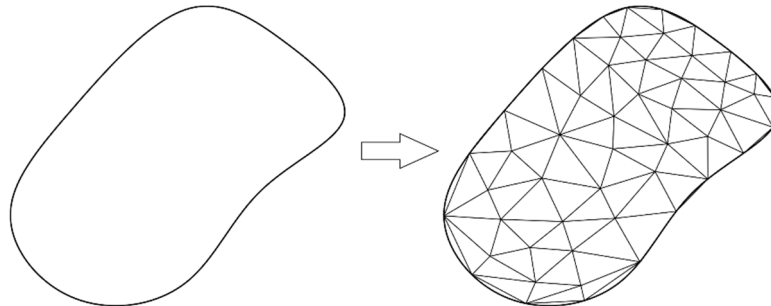
Fonte: Chapra e Canale (2008).



### 2.2.1.2 Método dos Elementos Finitos

O Método dos Elementos Finitos é outra aproximação numérica que encontra aplicação em diferentes áreas. Aplicações em modelos estruturais, análise de tensões e comportamento dinâmico de estruturas são algumas das principais atividades onde esse método encontra estudo e implementação. Trata-se de um método numérico que discretiza o domínio em elementos que conservam as propriedades do conjunto como um todo, transformando a análise global em uma série de análises adjacentes que permitem obter a aproximação final desejada. (LOTTI; MACHADO; MAZZIEIRO; LANDRÉ JÚNIOR, 2006). A utilização do MEF permite modelar o comportamento de meios contínuos a partir de elementos representativos onde é possível assumir que as equações diferenciais globais possam ser aproximadas por algum método variacional (figura 5). (REDDY, 2006).

**Figura 5** – Exemplo de discretização e de um domínio de análise bidimensional segundo uma geometria triangular dos elementos.



Fonte: O Autor.

Sergelind (1984) aponta que a principal diferença do Método dos Elementos Finitos com relação aos outros métodos numéricos está no fato de ele utilizar uma formulação integral para criar o sistema de equações algébricas que compõem a solução buscada e porque usa uma função contínua para realizar essa aproximação. Segundo esse mesmo autor, o seguinte conjunto de cinco etapas pode resumir a metodologia de aplicação do método.

- **Discretizar a região:**

Consiste em dividir o domínio de análise em subdomínios segundo uma geometria predefinida que facilite a representação do problema global. Uma maior densidade de elementos proporcionará resultados mais precisos.

- **Especificar as equações de interpolação:**

Determinar como a variável dependente se comportará dentro do elemento com relação aos seus nós e vértices. Isso é possível através da determinação das funções de forma, elas podem ser lineares, parabólicas nas principais aplicações.

- **Montar o sistema de equações:**

Através da aplicação de uma função de ponderação dos resíduos, determinar as equações algébricas que formará o sistema matricial ao solucionar a integral de ponderação dos resíduos associada à equação diferencial que rege o fenômeno, conforme a equação (32), onde  $i$  e  $j$  são os limites de distribuição dos elementos,  $W(x)$  é a função de ponderação dos resíduos e  $R(x)$  é a função resíduo, que pode ser obtida a partir da equação diferencial analisada.

$$\int_i^j W(x)R(x)dx = 0 \quad (32)$$

- **Resolver o sistema de equações e incorporar as condições de contorno.**

- **Calcular os valores que se deseja encontrar:**

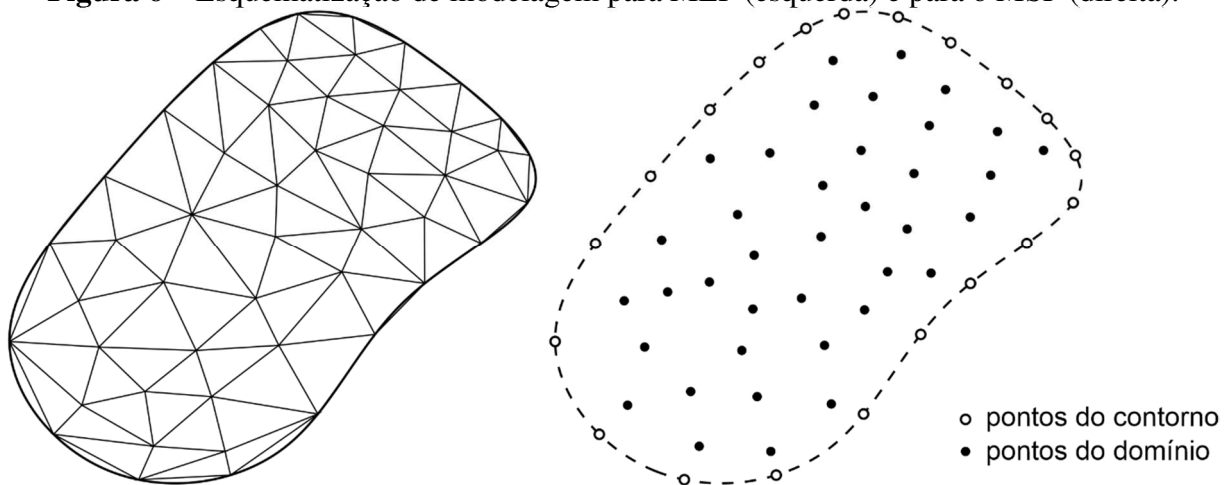
Pois nem sempre se tem interesse apenas nos valores da variável dependente. É comum buscar também os valores de suas derivadas. Para tanto, recorre-se às funções de forma que descrevem o comportamento da variável dentro do elemento.

### 2.2.2 Métodos livres de malha (*Meshless*)

Os métodos livres de malha, diferentemente do MDF e do MEF, aproximam a solução analítica através da discretização do domínio em pontos que não precisam conectar-se uns aos outros. Segundo Liu e Gu (2005), o requisito mínimo para que um método seja considerado livre de malha é o de que ele não necessite da construção de uma malha de elementos para a computação dos valores aproximados da variável dependente que está sendo investigada. Como condição ideal, os autores apontam que a malha seja dispensável em todas as etapas da modelagem, desde a definição da geometria até a formulação das equações algébricas de aproximação. A utilização de pontos em detrimento de elementos lineares ou de geometria predefinida para aproximar o comportamento de variáveis dependentes em domínio irregulares evita o surgimento de erros de aproximação geométricos, tais como os observados em

superfícies curvas simuladas como elementos planos triangulares ou curvas simuladas como segmentos de retas, assim como se observa no MEF. A figura 6 ilustra a diferença do tratamento do modelo para um problema encarado sob as principais características do MEF e de qualquer método *meshless* ideal. Nessa figura, no domínio da direita, é possível identificar os pontos brancos, que são os pontos onde estão definidas as condições de contorno iniciais, cujos valores serão utilizados para determinar os valores da variável dependente nos pontos pretos, contidos no domínio analisado.

**Figura 6** – Esquemática de modelagem para MEF (esquerda) e para o MSF (direita).



Fonte: O Autor.

A construção de uma malha que representa o domínio de análise de um modelo tradicional (com malha) compõe uma etapa que demanda esforço computacional e proporciona erros geométricos de aproximação em função da densidade da discretização adotada, o que influencia diretamente na precisão dos resultados obtidos. A aplicação de metodologias livres de malha apresenta como principais vantagens a eliminação do esforço de processamento associado à construção de uma malha, a facilidade de implementação em regiões geométricas irregulares, em modelagem de fraturas e crescimento de falhas em elementos contínuos e o controle da precisão de modo mais simplificado. (ARAÚJO, 2016, LIU; GU, 2005).

A formulação matemática dos métodos *meshless* recai em três categorias principais (LIU; GU, 2005):

- **Métodos fundamentados em formulação fraca (*Weak formulation*):**

Os métodos *meshless* baseados nessa formulação obtêm as equações algébricas a partir da solução de integrais associadas aos pontos que definem o contorno do problema analisado a partir de uma integral do tipo ponderação de resíduos

(equação 26). Exemplos desses métodos são o *Reproducing Kernel Particle Method (RKPM)* e o Método dos Elementos Difusos (*Diffuse Element Method – DEM*). (LIU; GU, 2005).

- **Métodos fundamentados em técnicas de colocação (*Strong-formulation*):**

Os métodos *meshfree*, ou *meshless*, baseados em técnicas de colocação utilizam as equações diferenciais que regem os fenômenos analisados para obter diretamente o sistema de equações algébricas em pontos de colocação que representam os pontos de interesse onde se deseja estimar o valor da variável dependente. (LIU; GU, 2005). Esse tipo de formulação é uma consequência da aplicação da formulação fraca para uma função de ponderação dos resíduos igual à função Delta de Dirac, que transforma a integral em um somatório sobre os pontos de colocação. (BREBBIA; DOMINGUEZ, 1992). Os métodos de Kansa com RBF e o Método das Soluções Fundamentais se encaixam nesta classificação.

- **Métodos mistos (formulação fraca e técnicas de colocação):**

Os métodos mistos utilizam as duas formulações anteriores para gerar um sistema de equações associado à formulação fraca para pontos no contorno que possuem condições do tipo 2 e técnicas de colocação para pontos com condição de contorno do tipo 1. Exemplo de método classificado como misto é o *Smooth Particle Hydrodynamics (SPH)* desenvolvido por Lucy (1977) e Gingold e Monaghan (1977). (ARAÚJO, 20116, LIU; GU, 2005).

### 2.2.2.1 Método das Soluções Fundamentais

O Método das Soluções Fundamentais é um método numérico verdadeiramente livre de malha que foi desenvolvido na década de 60 por V.D. Kupradze e M.A. Alexidze. (CHEN; KARAGEORGHIS; SMYLIRS, 2008) que pode ser classificado como um método de colocação, pois a função de ponderação dos resíduos é a função Delta de Dirac. (BREBBIA; DOMINGUEZ, 1992). Ele consiste na aproximação da solução analítica de uma Equação Diferencial Parcial (EDP) por uma expressão  $\tilde{u}$  na forma da equação (33), portanto uma combinação linear da solução fundamental do operador diferencial segundo as constantes  $\lambda$ .

$$u \cong \tilde{u} = \sum \lambda F(x, x_i) \mid x, x_i \in \mathcal{R}^n \quad (33)$$

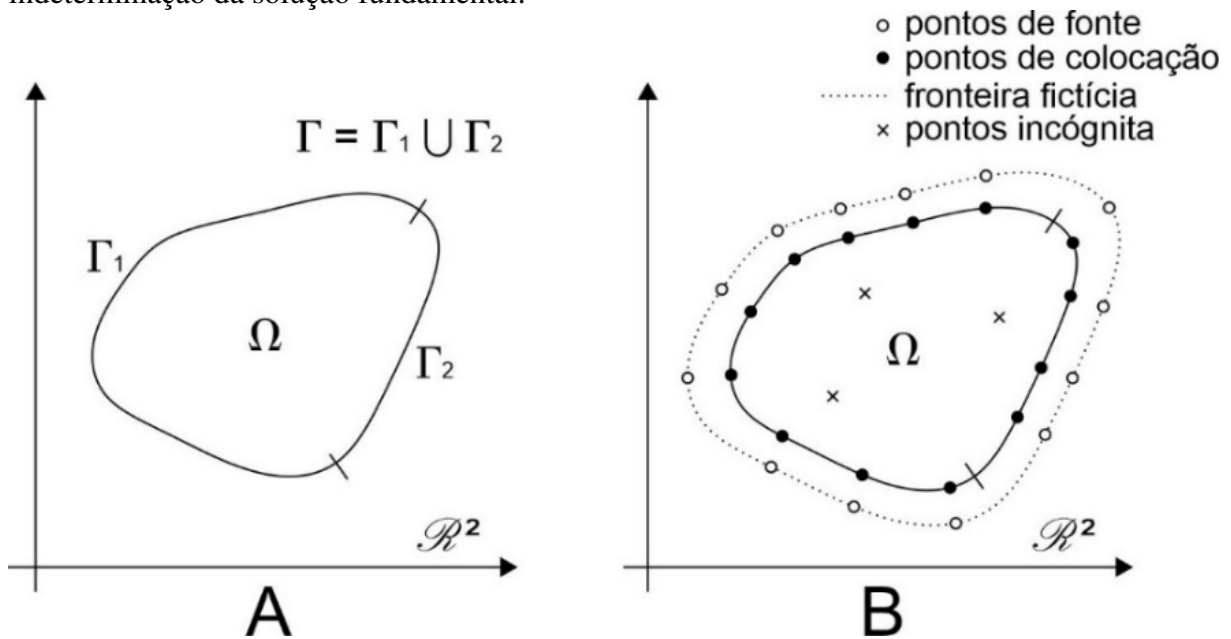
Em que  $u$  é a solução analítica,  $\lambda$  são constantes e  $F(x, x_i)$  é a solução fundamental do operador diferencial aplicado ao módulo de um vetor  $\mathbf{r}$  com início em  $x$  (pontos de fonte) e final em  $x_i$  (pontos de colocação), significando, portanto, a distância euclidiana entre os dois pontos no espaço  $\mathcal{R}^n$ . A solução aproximada  $\tilde{u}$  é a combinação linear de funções de forma iguais à solução fundamental do operador aplicado a cada ponto de colocação.

Em uma região do  $\mathcal{R}^2$  delimitada por um contorno  $\Gamma = \Gamma_1 \cup \Gamma_2$ , tal que  $\Gamma_1 \cap \Gamma_2 = \emptyset$ , e definida por um domínio  $\Omega$  onde  $u(x, y) \mid \nabla^2 u = 0$  (figura 7A), vale a seguinte relação (equação 34), pois os resíduos são iguais a zero nos pontos pertencentes ao contorno (pontos de colocação), em que  $n$  é o vetor unitário normal a esse contorno. De acordo com essa relação,  $\Gamma_1$  se refere a parte de  $\Gamma$  que possui condição de contorno de Dirichlet (tipo 1) enquanto  $\Gamma_2$  se refere à parte com condição de contorno de Neumann (tipo 2).

$$\left\{ \begin{array}{l} \tilde{u}(x_i) = u(x_i) \mid x_i \in \Gamma_1 \\ \frac{\partial \tilde{u}(x_j)}{\partial n} = \frac{\partial u(x_j)}{\partial n} \mid x_j \in \Gamma_2 \end{array} \right. \quad (34)$$

Uma vez que a solução fundamental do operador Laplaciano proporciona uma indeterminação matemática para pontos de fonte e de colocação coincidentes ( $\mathbf{r} = 0 \Rightarrow \ln|\mathbf{r}| = \text{indet.}$ ), é necessário criar uma fronteira fictícia  $\Gamma_{ficticia} \subset \Gamma$  para acomodar os pontos de fonte de tal forma que a coincidência seja evitada (figura 7B). Um domínio bidimensional como o representado genericamente pela figura 7B possui seu contorno composto por um conjunto de pontos de colocação. Cada um desses pontos está associado a um valor de condição de contorno (condição de Dirichlet ou de Neumann). Esse domínio está completamente contido dentro da fronteira fictícia (linha pontilhada) de formato paralelo ao formato da fronteira real (linha contínua) e afastada de uma distância  $d$ , o que define uma curva de *off-set*. Na fronteira fictícia são definidos pontos de fonte, tantos quanto forem os pontos de colocação existentes. Na região interna ao domínio existem pontos incógnita (representados como um “xis”), onde se deseja calcular o valor numérico da variável dependente. Essa interpretação guia o desenvolvimento das equações a seguir.

**Figura 7** - (A) Região do  $\mathcal{R}^2$  delimitada por um contorno  $\Gamma = \Gamma_1 \cup \Gamma_2$  e definida por um domínio  $\Omega$ . (B) Fronteira fictícia que permite a aplicação do método e extingue a indeterminação da solução fundamental.



Fonte: O Autor.

Aplicando a equação (33) para cada um dos pontos da fronteira fictícia em relação aos pontos do contorno e igualando à equação (34), pois os valores da função nesses pontos do contorno são conhecidos (condições de contorno do problema), é possível obter o seguinte sistema de equações (35) na forma matricial. Em que  $[f]$  corresponde ao bloco dos valores escalares das condições de contorno do tipo Dirichlet e  $[g]$  ao bloco dos valores escalares das condições de contorno do tipo Neumann. O vetor escalar  $[\lambda]$  pode ser determinado algebricamente (36) e pode ser utilizado para determinar os valores dos pontos incógnita  $s_i$  conforme a equação (37). As matrizes  $[\lambda]$  e  $\begin{bmatrix} f \\ g \end{bmatrix}$  são matrizes colunas e a matriz  $\begin{bmatrix} F(x, x_i) \\ \frac{\partial F(x, x_j)}{\partial n} \end{bmatrix}$  é uma matriz quadrada “n” por “n”, onde “n” é o número de pontos de colocação.

$$[\lambda] \begin{bmatrix} F(x, x_i) \\ \frac{\partial F(x, x_j)}{\partial n} \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \quad (35)$$

$$[\lambda] = \begin{bmatrix} F(x, x_i) \\ \frac{\partial F(x, x_j)}{\partial n} \end{bmatrix}^{-1} \begin{bmatrix} f \\ g \end{bmatrix} \quad (36)$$

$$\tilde{u}(s_i) = [\lambda][F(x_i, s_i)] = \sum \lambda_i F(x_i, s_i) \quad (37)$$

O Método das Soluções Fundamentais possui uma fonte de erros que está associada à determinação da melhor distância  $d$  do *off-set*. Contudo, Golberg e Chen (1998) mostraram que a precisão dos valores aproximados aumenta conforme essa distância tende ao infinito e que essa conclusão funciona para qualquer curva no espaço bidimensional que contenha o domínio de análise, seja ela uma circunferência ou um *off-set*. O fator limitante da aplicação dessa propriedade é o mal condicionamento das matrizes observado quando se utiliza valores de distância muito grandes relativos às dimensões do domínio. Os mesmos autores afirmam que o retorno recebido dos algoritmos de cálculo, que acusam matrizes não invertíveis, está relacionado ao limite imposto por índices de condicionamento e não à real impossibilidade de inversão.

### 2.3 Hidrogeologia

Hidrogeologia pode ser definida como a ciência que estuda os fenômenos que ocorrem entre as formações geológicas e a água nas camadas superiores da crosta terrestre. (FETTER, 2001). Assim, o estudo do escoamento superficial, da infiltração, do fluxo subsuperficial e subterrâneo em regiões sedimentares, cristalinas e urbanas são seus principais tópicos de estudo. A aplicação dos resultados obtidos na investigação dessa relação proporciona a criação de modelos de análise que permitem prever, a partir do monitoramento de algumas variáveis específicas, como a água se comporta, suas direções de fluxo, seu nível de ocorrência, dentre outros aspectos. Esses resultados permitem que os setores administrativos apliquem uma gestão racional e tecnológica dos recursos hídricos subterrâneos, assim é possível conduzir as atividades antrópicas de exploração através de decisões que produzam resultados eficientes e preservem a abundância e a ocorrência dessas fontes de recurso naturais. (MIDDLEMIS; WALKER; PEETERS; RICHARDSON; HAYES; MOORE, 2019).

Para o desenvolvimento desta pesquisa, foram utilizados os conceitos associados principalmente ao fluxo de água subterrâneo. Para tanto, foram revisitadas as equações e hipóteses que produziram as fórmulas clássicas que descrevem esse fenômeno e também foi realizada uma revisão acerca das formações geológicas sedimentares.

### 2.3.1 Conceitos Básicos

- **Lei de Darcy**

O fluxo de água subterrâneo em meios porosos pode ser descrito pela Lei de Darcy. A série de experimentos que ele realizou em 1874 utilizando o aparato esquematizado na figura 8, permitiu a obtenção da equação (38), que associa a vazão observada à carga hidráulica entre duas seções.  $Q$  é a vazão observada entre as duas seções onde estão instalados os piezômetros,  $k$  é a constante conhecida como condutividade hidráulica,  $Ar$  é a área da seção transversal do aparato e  $\phi_i$  e  $z_i$  são a carga hidráulica e a cota associadas à seção  $i$ , respectivamente.

$$Q = k \cdot Ar \cdot \frac{\phi_2 - \phi_1}{z_2 - z_1} \quad (38)$$

Normalizando a vazão com relação à área transversal através da utilização do conceito de vazão específica  $q_z$  na direção do eixo  $z$ , definida na equação (39), e assumindo que as diferenças expressas na equação (38) valem em nível infinitesimal, é possível escrever a equação (40), e generalizar a Lei de Darcy para  $n$  dimensões segundo a notação tensorial da equação (41), onde os índices  $i$  e  $j$  significam as componentes espaciais dos sistemas de coordenadas dos campos vetoriais da descarga específica e do gradiente, respectivamente, e o somatório ocorre sobre o índice  $j$ . O sinal negativo na equação (41) implica que a direção da vazão específica é no sentido oposto à direção do gradiente hidráulico.

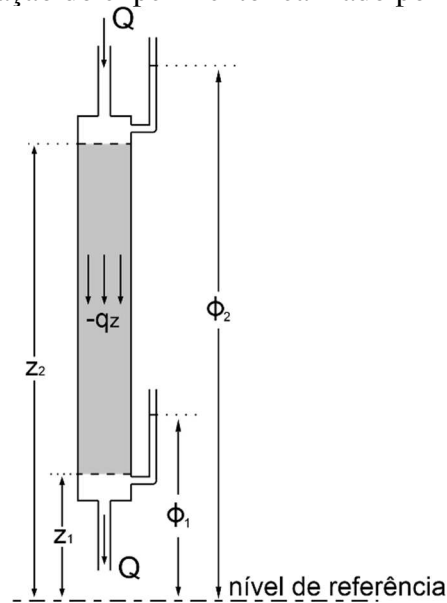
$$q_z = \frac{Q}{Ar} \quad (39)$$

$$\frac{\phi_2 - \phi_1}{z_2 - z_1} = \frac{d\phi}{dz} \cong \frac{\partial\phi}{\partial z} \quad (40)$$

$$q_i = -k_{ij} \cdot \partial_j \phi \quad (41)$$



**Figura 8** – Esquematização do experimento realizado por Darcy.



Fonte: Adaptado de Haitjema (1995).

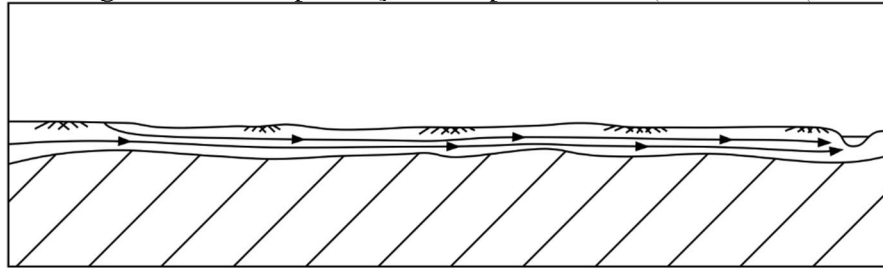
Na figura 8,  $Q$  é a vazão de entrada e de saída,  $q_z$  é a vazão normalizada segundo a área transversal do aparato,  $\Phi_i$  é a carga hidráulica associada aos dois piezômetros e  $z_i$  é a medida da cota segundo um plano horizontal de referência.

- **Simplificação de Dupuit-Forchheimer**

A hipótese de Dupuit-Forchheimer foi proposta por esses dois pesquisadores de maneira independente em 1863 e 1886 respectivamente. Segundo essa simplificação, para aquíferos sedimentares cuja extensão horizontal supere consideravelmente a profundidade (figura 9), onde as dimensões horizontais superam consideravelmente as dimensões verticais, é possível adotar que a vazão vertical  $q_z$  é igual a zero (equação 42).

$$q_z = 0 \quad (42)$$

Strack (1984) e Haitjema (1987) investigaram a influência dessa simplificação na obtenção de linhas de fluxo para simulações de aquíferos reais e verificaram a precisão e a viabilidade da utilização dessa hipótese para a modelagem hidrogeológica na maioria dos casos práticos. (HAITJEMA, 1995).

**Figura 9** – Exemplificação de aquífero livre (vista lateral).

Fonte: Adaptado de Haitjema (1995).

Ao adotar a hipótese de um fluxo bidimensional, a equação de Darcy (equação 41) pode ser reescrita como expresso na equação (43), onde os índices  $x$  e  $y$  estão associados aos eixos ortogonais de um plano de referência horizontal. Na hipótese de um aquífero isotrópico e na coincidência na orientação dos eixos dos campos tensorial de condutividade hidráulica e vetorial de gradiente hidráulico, a equação (43) pode ser simplificado como mostrado na equação (44). Essa simplificação relaciona o gradiente hidráulico diretamente às direções de fluxo da vazão específica e permite o cálculo dos valores do primeiro a partir do segundo, desde que se conheça um valor de condutividade hidráulica representativo.

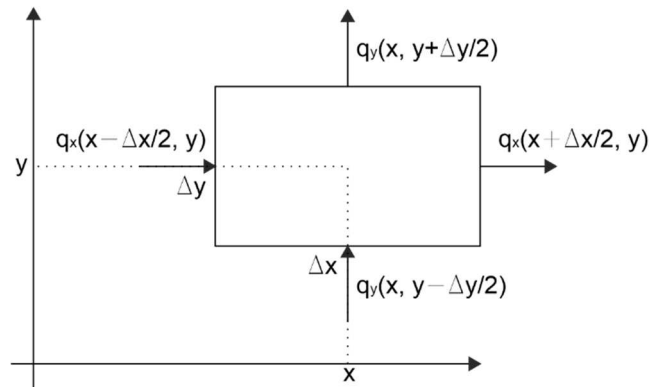
$$\begin{cases} q_x = -k_{xx} \cdot \partial_x \phi - k_{xy} \cdot \partial_y \phi \\ q_y = -k_{xy} \cdot \partial_x \phi - k_{yy} \cdot \partial_y \phi \end{cases} \quad (43)$$

$$\begin{cases} q_x = -k \cdot \partial_x \phi \\ q_y = -k \cdot \partial_y \phi \end{cases} \quad (44)$$

- **Equação da continuidade**

A equação da continuidade pode ser determinada a partir de um elemento representativo do aquífero (figura 10), cujo centro está localizado nas coordenadas  $(x, y)$  e possui laterais perpendiculares iguais a  $\Delta x$  e  $\Delta y$ . Assumindo a incompressibilidade da água e o fluxo bidimensional, é possível escrever que toda a água que entra no elemento sai sob a mesma orientação inicial. O somatório das diferenças de vazão específica entre as faces opostas e coaxiais está expresso na equação (45). Dividindo ambos os termos pelo produto  $\Delta x \Delta y$ , obtém-se a equação (46). Fazendo  $\Delta x, \Delta y \rightarrow 0$  na equação (46), é possível expressar as diferenças como os diferenciais da vazão específica em cada um dos eixos.

**Figura 10** - Elemento representativo do aquífero bidimensional.



Fonte: O Autor.

Em termos de aproximação, os diferenciais podem ser substituídos pelas derivadas e a equação (46) é reescrita como a equação (47), que é a equação diferencial da continuidade expressa em termos da vazão específica. Ao substituir a Lei de Darcy (equação 44) sob as hipóteses da isotropia e da equivalência de sistema de eixos entre os campos tensorial da condutividade hidráulica e vetorial do gradiente hidráulico, obtém-se a equação (48), que é a equação de Laplace bidimensional aplicada à carga hidráulica.

$$\left[ q_x \left( x - \frac{\Delta x}{2}, y \right) - q_x \left( x + \frac{\Delta x}{2}, y \right) \right] \Delta y + \left[ q_y \left( x, y - \frac{\Delta y}{2} \right) - q_y \left( x, y + \frac{\Delta y}{2} \right) \right] \Delta x = 0 \quad (45)$$

$$\frac{\left[ q_x \left( x - \frac{\Delta x}{2}, y \right) - q_x \left( x + \frac{\Delta x}{2}, y \right) \right]}{\Delta x} + \frac{\left[ q_y \left( x, y - \frac{\Delta y}{2} \right) - q_y \left( x, y + \frac{\Delta y}{2} \right) \right]}{\Delta y} = 0 \quad (46)$$

$$\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = 0 \quad (47)$$

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (48)$$

Encontrar as linhas equipotenciais de carga hidráulica em um problema de fluxo de água subterrâneo cuja modelagem assume as hipóteses anteriores está associado a encontrar uma solução particular do operador Laplaciano para determinados valores iniciais. Como nem sempre uma solução analítica é viável, métodos numéricos são utilizados para aproximar os resultados em pontos ou elementos em um domínio de análise.

### 2.3.2 Aquíferos Sedimentares

Os aquíferos sedimentares são formados por rochas sedimentares cuja porosidade permite o armazenamento e o fluxo de água em quantidades suficientes para a exploração econômica desse recurso em atividades antrópicas. A porosidade de uma rocha pode ser calculada segundo a equação (49), onde  $n$  é a porosidade,  $V_v$  é o volume de vazios e  $V$  é o volume total de uma amostra representativa. O índice de vazios  $e$  é outra grandeza associada à porosidade de um material e pode ser relacionado com a porosidade segundo a equação (50). A tabela 3 mostra o intervalo de valores de porosidade para alguns tipos de materiais.

$$n = \frac{V_v}{V} \quad (49)$$

$$e = \frac{n}{1 - n} \quad (50)$$

**Tabela 3** – Intervalo de valores de porosidade para alguns tipos de aglomerados de sedimentos e para alguns tipos de rochas sedimentares.

<b>Material</b>	<b>Porosidade</b>
<b>SEDIMENTOS</b>	
Seixo, grosso	24-36
Seixo, fino	25-38
Areia, grossa	31-46
Areia, fina	26-53
Silte	34-61
Argila	34-60
<b>ROCHAS</b>	
Arenito	5-30
Siltito	21-41
Calcário, dolomítico	0-40
Xisto	0-10

Fonte: Davis (1969) e Johnson e Morris (1962) apud Domenico e Schwartz (1997).

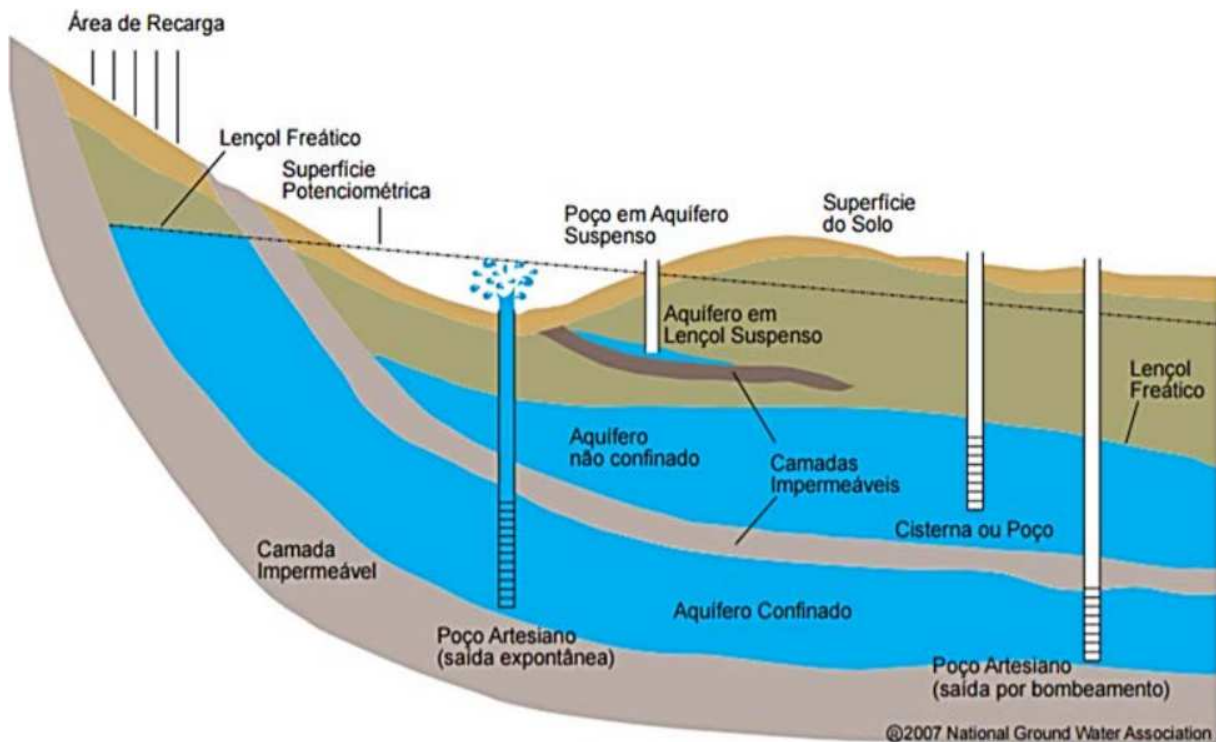
A capacidade de uma formação rochosa conduzir água a partir da existência de um gradiente hidráulico motriz está associado a sua porosidade efetiva. Diferentemente da porosidade definida na equação (49), a porosidade efetiva é espaço conectado entre os poros. Esse espaço limita o fluxo de água e seu valor é sempre menor que a porosidade definida na equação (49). A porosidade efetiva é uma propriedade da formação geológica e depende dos seus processos de formação e consolidação. Entretanto, essa grandeza fornece apenas uma ideia

inicial do potencial de condução de um aquífero. Em termos práticos é introduzida outra grandeza, a condutividade hidráulica, associada a Lei de Darcy, que é aferida experimentalmente ou estimada estatisticamente e representa a capacidade efetiva de transporte de uma formação rochosa. Gonçalves e Libardi (2013) explicam que na situação de solo saturado a condutividade é máxima, pois todos os poros contribuem para o escoamento. Por outro lado, o solo não saturado possui ar ocupando os espaços vazios, o que reduz drasticamente a condutividade hidráulica. Assim, esse parâmetro é uma propriedade dependente das condições de saturação, enquanto a porosidade é uma propriedade fixa da formação sedimentar.

As formações geológicas sedimentares são formadas pelos processos de erosão, transporte e depósito de sedimentos ao longo das eras geológicas às quais a superfície do planeta esteve exposta. De acordo com Domenico e Schwartz (1997) a ocorrência de um depósito sedimentar continental e a sua correspondente formação geológica está associada a seis categorias distintas: depósitos fluviais, vales aluviais, bacias aluviais, depósitos eólicos, depósitos lacustres e depósitos glaciais. Em cada uma dessas categorias, os agentes erosivos depositam sedimentos em regiões ao longo do tempo que se consolidam e formam rochas sedimentares. Essas formações recebem e acumulam água de fontes diversas, sendo a infiltração da água das chuvas a fonte mais comum.

Durante os processos de deposição de sedimentos, diferentes agentes erosivos podem contribuir para a chegada de partículas com diferentes propriedades químicas e granulométricas, produzindo ao longo das etapas de consolidação camadas de rochas sedimentares com valores de permeabilidade distintos. Assim, essas camadas podem ser classificadas como aquíferos, quando suas propriedades granulométricas permitem a ocorrência de fluxo suficiente para a exploração de água em alguma atividade antrópica ou aquíclode ou aquíclardio, em caso contrário (figura 11). Essas últimas também são chamadas de camadas impermeáveis ou confinantes, no entanto, elas ainda possuem alguma permeabilidade, cujo valor é apenas muito pequeno comparado com as primeiras. A capacidade de conduzir uma pequena quantidade de água é observada até mesmo em rochas matrizes fraturadas. É comum encontrar nas regiões sedimentares camadas intercaladas com diferentes permeabilidades. A figura 11 mostra a existência de regiões em aquíferos onde a carga hidráulica coincide com o potencial piezométrico, quando isso ocorre, o aquífero é classificado como livre. Quando o potencial piezométrico se desloca do nível da água por conta da imposição de camadas impermeáveis, o aquífero é classificado como confinado.

**Figura 11** – Esquemática da vista lateral de uma formação geológica que contém aquíferos com regiões de fluxo livre e fluxo confinado.



Fonte: Pinto-Coelho; Havens (2016)

O armazenamento de uma formação sedimentar pode ser medido em termos dos parâmetros produção específica (*specific yield* -  $S_y$ ) e armazenamento específico (*specific storage* -  $S_s$ ). A produção específica é definida como a porcentagem de água liberada de um elemento de volume unitário representativo do aquífero sob o efeito exclusivo da gravidade, é possível deduzir que o valor de produção específica de um elemento de aquífero é sempre inferior ao seu valor de porosidade. O armazenamento específico é o volume de água que um elemento representativo libera quando um valor unitário de carga hidráulica é “aliviado” da sua fronteira, sendo que o elemento continua saturado durante o processo, assim, são levadas em consideração os fenômenos de compressibilidade da água e da própria formação rochosa. (FETTER, 2001). Esses dois parâmetros são relacionados através da equação (51) para definir o conceito de armazenamento (*Storativity* -  $S$ ), onde  $b$  é a profundidade do aquífero. Esse conceito resume que a capacidade de fornecimento de água é a soma entre o potencial de dessaturação e o de descompressão da água contida nos poros da rocha.

$$S = b.S_y + S_s \quad (51)$$

O processo de fornecimento/extração de água por parte de um lençol freático pode ocorrer de duas maneiras, conforme ele seja classificado como livre ou confinado. Um aquífero livre libera água através principalmente do fenômeno de dessaturação dos seus poros, enquanto um aquífero confinado, em função das grandes pressões às quais está submetido, libera água por descompressão. Segundo Fetter (2001), a inclusão desses dois fenômenos na descrição do fluxo de água subterrâneo acrescenta termos de inhomogeneidade ao operador Laplaciano, produzindo as seguintes equações (52) e (53) em duas dimensões. A equação (52) se aplica a aquíferos livres sob a hipótese de que o volume de água extraído não afeta significativamente o nível freático e a equação (53) a aquíferos confinados. Nessas equações,  $\phi$  é a carga hidráulica,  $K$  é a condutividade hidráulica,  $T$  é a transmissividade, definida como  $T = Kb$ , sendo  $b$  a espessura do aquífero,  $t$  é o tempo e  $l$  é o fluxo de água que ocorre para as camadas confinantes (menos permeáveis). O desenvolvimento desta pesquisa não incorpora elementos de inhomogeneidades. Portanto, as equações (52) e (53) são trabalhadas em sua forma homogênea, que se resumem à equação de Laplace para ambas as situações, aquíferos livre ou confinados.

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = \frac{S_y}{T} \frac{\partial \phi}{\partial t} \quad (52)$$

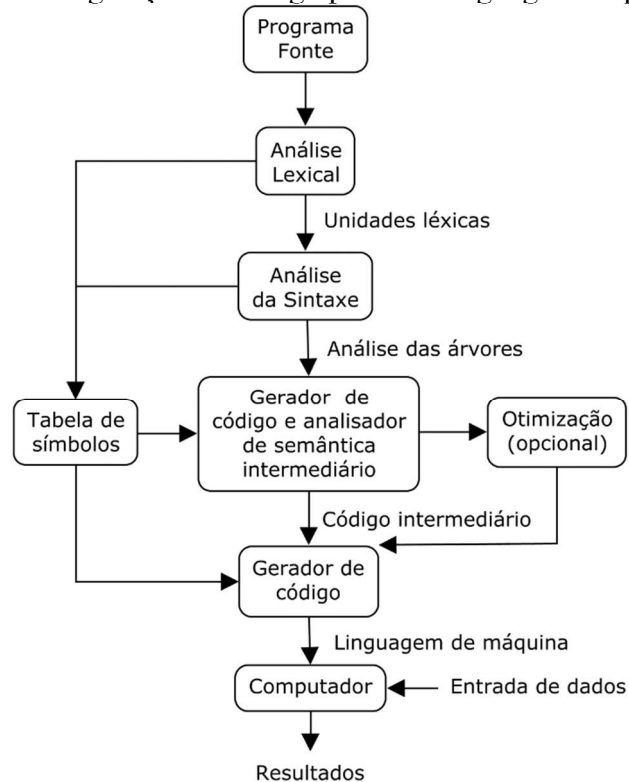
$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{l}{T} = \frac{S}{T} \frac{\partial \phi}{\partial t} \quad (53)$$

## 2.4 Linguagens de programação

As linguagens de programação realizam a comunicação entre programador e máquina e permitem a implementação de algoritmos em sistemas operados binariamente pelos circuitos eletrônicos que formam o núcleo de processamento da unidade computacional. Sebesta (2009) classifica as linguagens de programação em quatro grupos: imperativas, funcionais, lógicas e orientadas a objeto. As imperativas e funcionais são caracterizadas pelo extenso detalhamento do algoritmo e a especificação das regras (funções) em uma ordem não necessariamente sequencial. As lógicas requerem a especificação das regras e algoritmo em ordem sequencial e progressiva. As orientadas a objeto derivam das linguagens imperativas e funcionais e normalmente são classificadas também como linguagens de alto nível, porque necessitam de várias camadas de código que dão suporte à criação do programa desde as etapas de confecção

e escrita até propriamente o ambiente de execução. Esse mesmo autor apresenta o seguinte fluxograma (figura 12) que ilustra o processo genérico de criação de código em uma linguagem de programação.

**Figura 12** – Fluxograma de geração de código por uma linguagem de programação genérica.



Fonte: Sebesta (2009).

O analisador léxico identifica as palavras especiais reservadas da linguagem e separa os blocos de programas em unidades léxicas. O analisador de sintaxe identifica a organização da árvore de blocos em que o programa está dividido, o que permite o gerador de código e analisador de semântica intermediário gerar um programa em uma linguagem intermediária para que o compilador, o gerador de código, crie o código em linguagem de máquina. Esse conjunto de instruções binárias é “entendido” pelo processador e executa o algoritmo implementado a partir dos dados de entrada. (SEBESTA, 2009).

### 2.4.1 Java

Segundo a Oracle (2019, tradução nossa), “Java é uma linguagem de alto nível e de plataforma de software único. Ela roda em mais de 50 milhões de computadores pessoais e em



bilhões de dispositivos ao redor do mundo. 9 milhões de desenvolvedores criam aplicações em Java em todos os ramos da indústria”. A mesma aplicação pode rodar em múltiplas plataformas graças a implementação do *Java Virtual Machine (JVM)*, que permite essa flexibilidade. Junto com a API (*Application Program Interface*), o JVM compõe a plataforma Java, que pode ser instalada e manipulada para a construção de novos programas em qualquer computador pessoal e potencialmente em qualquer dispositivo que ofereça os recursos mínimos de processamento.

Java é uma linguagem orientada a objetos. Isso significa que a capacidade de representação dos problemas reais é expandida através dessa abstração. Antes desse tipo de linguagem se tornar efetiva no ambiente de programação, os algoritmos, que correspondem à sequência de instruções lógicas que conduzem à resolução de um problema, eram pensados para se adequar às limitações da máquina. Assim, problemas complexos eram modelados em algoritmos muito mais complexos. A perspectiva de se trabalhar com objetos facilitou a representação dos sistemas e também reduziu a complexidade e a mão-de-obra associada ao desenvolvimento de *softwares*. Eckel (2006) afirma que qualquer linguagem orientada a objetos parte dos seguintes princípios básicos:

- **Tudo é um objeto.** Um objeto pode ser entendido como uma variável abrangente que pode armazenar dados, possuir parâmetros, solicitar instruções e/ou realizar operações. Nesse sentido, qualquer fenômeno pode ser implementado como um objeto. Um mesmo objeto pode ser desmembrado em outros objetos. A estrutura de objetos dependerá da estratégia de modelagem adotada pelo programador.
- **Um programa é um conjunto de objetos que interagem entre si através de instruções.**
- **Cada objeto possui sua própria memória dedicada, de tal forma que pode ser manipulado independentemente de outros objetos do mesmo tipo.**
- **Todo objeto tem um tipo.** No âmbito da programação orientada a objeto, o termo “instanciar uma classe” significa criar um objeto de um determinado tipo que é descrito por uma determinada classe. Um exemplo simples seria a criação e uma classe chamada “automóvel”. Essa classe genérica e abstrata enumera as principais propriedades de um automóvel. Assim, segundo essa classe, qualquer objeto que seja do tipo automóvel terá algumas propriedades como cor, potência, consumo, peso, marca, ano de fabricação, dentre outros. Se alguém deseja instanciar um objeto da classe automóvel, esse alguém necessitará especificar

esses parâmetros. Da mesma forma, podem ser criados tantos automóveis quantos sejam necessários, contudo, todos eles compartilharão a mesma estrutura abstrata e genérica da classe que fazem parte. Isso é, cada um dos automóveis terá uma cor, uma potência, um consumo, um peso, uma marca, um ano de fabricação, dentre outros. Essa criação de objetos através de classes gera o próximo princípio.

- **Todos os objetos do mesmo tipo respondem às mesmas mensagens.** Em outras palavras, utilizando o exemplo anterior, se alguém sabe que determinado objeto é do tipo automóvel, então é possível solicitar a esse objeto a informação sobre o qualquer um dos seus parâmetros. Qualquer objeto automóvel será passivo de responder qual a sua cor, sua potência, dentre outras propriedades.

A orientação a objetos permite a modelagem de problemas complexos através da interação entre diferentes tipos de objetos, que realizam ações específicas e coordenadas. Pensar em um problema segundo essa metodologia é diferente de trabalhar com o conceito imperativo das linguagens de baixo-nível. O Java também proporciona uma série de ferramentas que facilitam o processo de concepção e desenvolvimento de programas. O tratamento de exceções, o suporte a aplicativos em ambiente Web, a existência de *plug-ins*, de programas acessórios e de uma extensa variedade de bibliotecas dos mais variados tipos distribuídos por empresas particulares e por projetos *Open-Source*, tais como bibliotecas gráficas, de análise matemática, de edição de textos, dentre outras, transformam a linguagem Java em uma das melhores ferramentas para o desenvolvimento de *softwares* das últimas décadas. Apesar dessa flexibilidade, o Java tem perdido público para novas linguagens dedicadas ao ambiente web, como o Python.

Os programas de escrita de *softwares* em Java mais conhecidos e utilizados pelos grupos de desenvolvimento são o Eclipse e o NetBeans. O primeiro é apreciado pelos entusiastas e pelos pequenos grupos, pois além de ser o resultado de um projeto *Open Source* possui suporte indireto em diversos fóruns *on-line* e uma comunidade que desenvolve extensões que incorporam as novidades das novas edições da linguagem Java em velocidade competitiva e eficiente para a maior parte das aplicações. O segundo é mais aplicado em grandes empresas de *software* que requerem uma equipe de profissionais trabalhando simultaneamente nos mesmos blocos de código e precisam de uma plataforma que permita organizar o fluxo corporativo de dados.

### 2.4.2 SciLAB

O Scilab é um *software* gratuito e *open source* para engenheiros e cientistas, com uma longa história e uma comunidade crescente. Sua primeira versão data de 1994 e possui uma linguagem de programação orientada ao número e às operações matemáticas. (SCILAB, 2019). É uma alternativa ao MATLAB e possui uma comunidade *on-line* que fornece suporte indireto. A linguagem de alto nível permite a manipulação de operações extensas e o monitoramento de variáveis em diversas aplicações e simulações numéricas. Atualmente possui uma versão *on-line* que pode ser executada na nuvem, dinamizando o acesso e proporcionando poder de processamento aos usuários. Sua interface simplificada e intuitiva requer pouco tempo de familiarização. A facilidade na implementação de operações algébricas transforma o Scilab numa ferramenta viável para o teste de hipóteses associadas a métodos numéricos e ao teste de algoritmos.

### 3 METODOLOGIA

A metodologia adotada nesta pesquisa seguiu as seguintes etapas:

- **Desenvolvimento do algoritmo para aplicação do MSF;**
- **Implementação do algoritmo em linguagens Java e Scilab;**
- **Delimitação e caracterização da área de estudo;**
- **Avaliação da influência da localização da fronteira fictícia.**

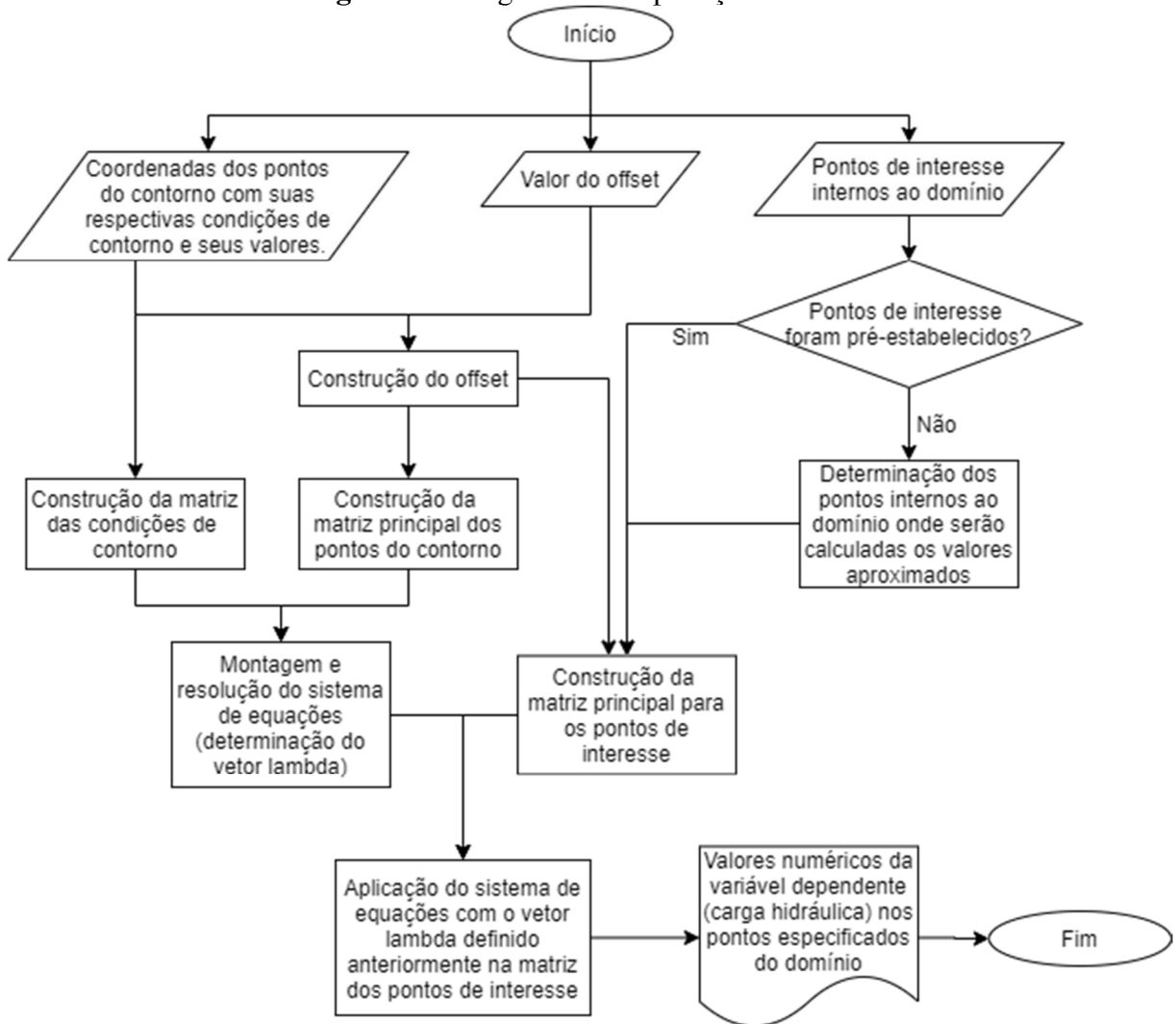
#### 3.1 Desenvolvimento do algoritmo para aplicação do MSF

A aplicação do Método das Soluções Fundamentais utiliza as condições de contorno dos pontos de colocação localizados na fronteira do domínio analisado num sistema de equações. Esses pontos de colocação podem representar condições de contorno de Dirichlet ou de Neumann, principalmente. A construção da matriz principal da equação (29) pode ser feita linha por linha, o que permite o encadeamento sequencial dos pontos e a inserção da relação de interdependência que eles possuem na combinação linear que compõe a solução final através da solução do sistema de equações. Essa solução significa determinar os coeficientes  $\lambda$  do vetor escalar da equação (29). Assim, o algoritmo global foi pensado conforme o fluxograma da figura 13, adotando a notação proposta por Ascênsio e Campos (2007). Segundo essa notação, as elipses indicam o início e o fim do algoritmo, os paralelogramos indicam entrada de dados, os losangos representam condicionais, os retângulos representam ações e operações genéricas e o símbolo de nota de papel representa a saída de dados.

De acordo com esse algoritmo são necessários três grupos de dados de entrada que representam a modelagem da região de estudo, são eles: coordenadas dos pontos do contorno associadas aos seus valores de contorno, bem como um marcador do seu tipo (tipo 1 ou tipo 2), valor do *offset* (adotaremos que a expressão “valor do *off-set*” se refere à distância  $d$  que separa a fronteira real da fronteira fictícia, como esquematizado na figura 7) e coordenadas dos pontos internos ao domínio onde se deseja conhecer a solução aproximada. Caso os pontos de interesse não sejam fornecidos, o algoritmo propõe um conjunto de pontos distribuídos ao longo do domínio que pode ser utilizado para desenhar as curvas equipotenciais e as correspondentes linhas de fluxo. O *off-set* determinado a partir do valor do *off-set* e do conjunto de pontos do contorno são utilizados para construir a matriz principal que será invertida durante a resolução do sistema de equações. A matriz das condições de contorno é construída diretamente a partir

do primeiro grupo de dados de entrada. Após a resolução das equações algébricas, o conjunto de informações que relaciona os pontos de interesse e seus valores aproximados é disponibilizado em um arquivo de texto que pode ser exportado diretamente para algum *software* SIG e tratado vetorialmente como um *shapefile*.

**Figura 13** – Algoritmo de aplicação do MSF.



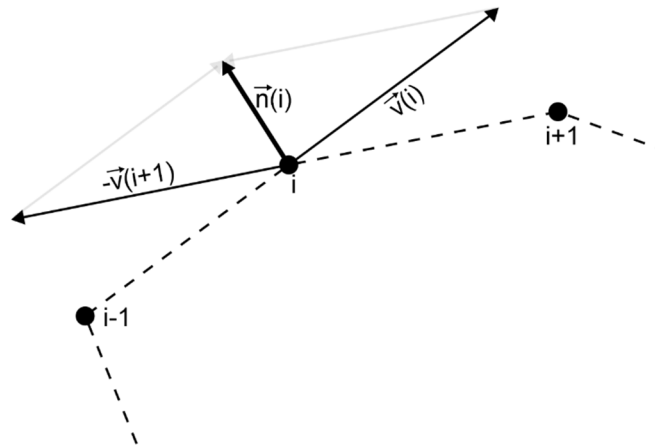
Fonte: O Autor.

A construção do *off-set* partiu do vetor normal aproximado dos pontos que compõem os vértices do polígono que delimitam a área de estudo. De acordo com a figura 14, três pontos consecutivos podem ser utilizados para determinar o vetor normal aproximado do ponto intermediário através da equação vetorial (equação 54), onde as letras em negrito representam os vetores associados aos pontos com índice  $i$ , sendo  $\mathbf{n}$  o vetor normal e  $\mathbf{v}$  os vetores que orientam os lados do polígono. Com esse vetor normal aproximado podem ser criadas imagens

paralelas do contorno do polígono a qualquer distância desejada. Essa distância corresponde ao valor do *off-set*, que é o parâmetro cuja influência foi investigada nesta pesquisa.

$$\mathbf{n}(i) = \mathbf{v}(i) - \mathbf{v}(i - 1) \quad (54)$$

**Figura 14** – Determinação do vetor normal do vértice de um polígono em função dos vértices anterior e posterior.



Fonte: O Autor.

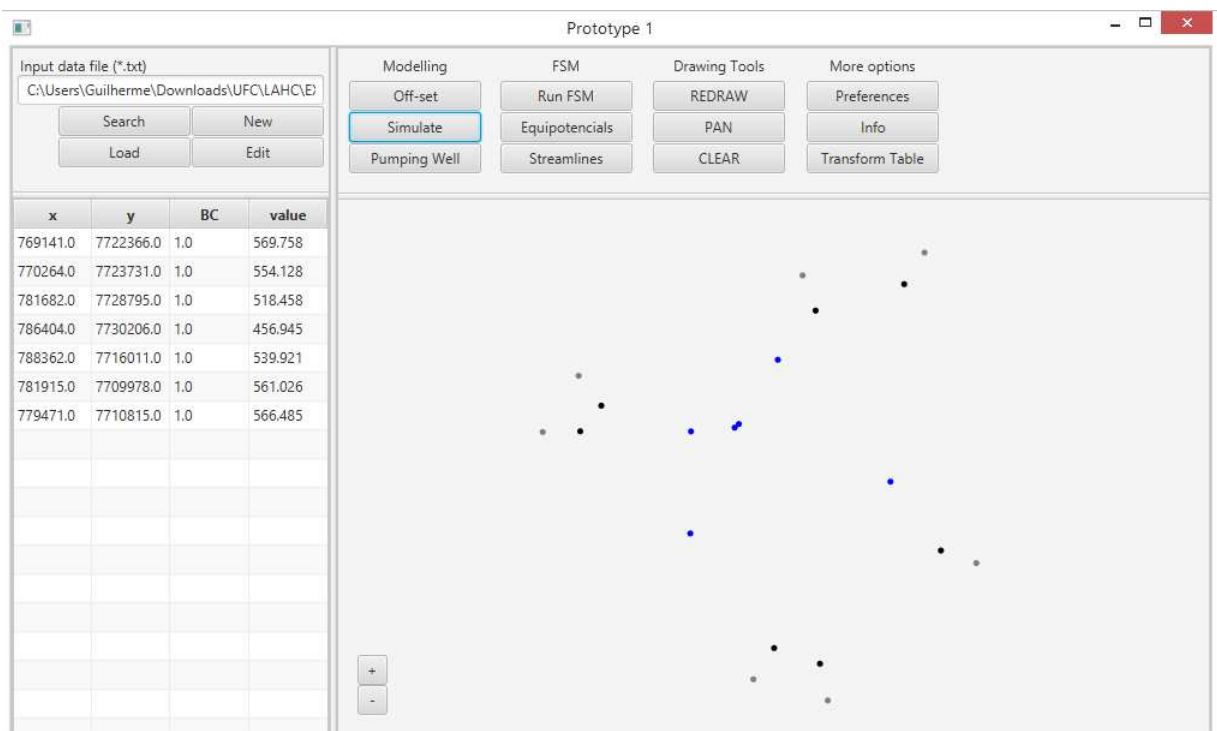
A construção da matriz principal foi realizada através da utilização de um algoritmo específico. As linhas da matriz principal relacionam um único ponto de fonte a todos os pontos de colocação do contorno, ou seja, elas podem ser construídas individualmente e posteriormente agregadas para compor a matriz completa. Essa estratégia foi aplicada tanto na fase de determinação do vetor escalar de constantes  $\lambda$  quanto no cálculo dos valores aproximados de carga hidráulica nos pontos de interesse.

### 3.2 Implementação do algoritmo em linguagem Java e Scilab

A implementação do algoritmo em Java utilizou a biblioteca gráfica *JavaFX* (<https://openjfx.io/>) e a biblioteca de cálculo matricial *Apache Commons Math* (APACHE COMMONS, 2016). Foi construída uma interface básica que permite a inserção de arquivos de texto que contêm as coordenadas dos pontos do contorno e os seus valores, as coordenadas dos pontos de interesse e uma opção de configurar o valor do *off-set* requerido pelo método respeitando o algoritmo definido anteriormente. A interface gráfica permite a visualização simplificada da modelagem que está sendo realizada. O programa implementa a estratégia *Model-View-Controller* (MVC), que separa a parte gráfica (*view*) da interface de cálculo (*model*) e as conecta através de um conjunto de instruções (*controller*). Essa estratégia possui

vantagens como as facilidades na migração entre bibliotecas gráficas e linguagens de programação distintas, na manutenção de código, na implementação de novas ferramentas e na capacidade de assimilação por parte de outros programadores, na oportunidade em que podem ajudar no desenvolvimento do software, ao mesmo tempo permite o tratamento matemático e o monitoramento preciso e funcional das operações algébricas envolvidas na aplicação do método. (SCIORE, 2019). A figura 15 ilustra o aspecto da janela principal durante a etapa que antecede a aplicação do Método das Soluções Fundamentais. Nessa interface não estão disponíveis informações de escala ou de projeção cartográfica. Ela foi criada apenas para uma visualização rápida da configuração geométrica dos dados inseridos, pois pressupõe-se a correspondência com dados aferidos em softwares SIG. Nela, os pontos pretos representam os poços de observação que delimitam o domínio de análise. Os pontos de cor cinza representam os pontos da fronteira fictícia, o *off-set*. Os pontos azuis representam os pontos de interesse, onde se deseja calcular os valores estimados da carga hidráulica. A tabela a esquerda mostra os valores das coordenadas dos pontos da fronteira bem como o tipo de condição de contorno e o valor.

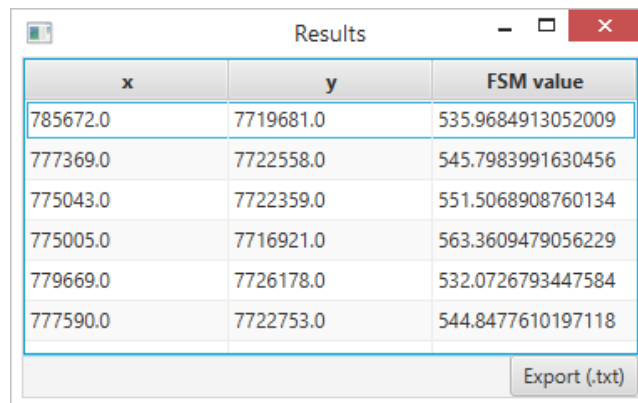
**Figura 15** – Janela principal do programa desenvolvido em Java para a aplicação do Método das Soluções Fundamentais.



Fonte: O Autor.

O resultado da aplicação do Método pode ser exportado para um arquivo de texto em formato de modelo digital de elevação (MDE), facilmente reproduzido em softwares de informação geográfica, como o ArcGIS e o QGIS, por exemplo (figura 16). Alguns botões não possuem funções implementadas. Apenas as funções *Search*, *Load*, *Off-set*, *Simulate*, *Run MFS*, *Redraw*, *Clear*, *Zoom In* e *Zoom Out*, que desempenham as atividades mínimas necessárias para a aplicação linear do método a um conjunto de dados iniciais, estão operantes. As curvas equipotenciais são obtidas através da utilização de softwares SIG a partir da interpolação dos valores calculados nos pontos internos em uma malha de pontos com densidade suficiente para representar a superfície piezométrica obtida numericamente.

**Figura 16** – Janela dos resultados da aplicação do MSF aos pontos de interesse.



x	y	FSM value
785672.0	7719681.0	535.9684913052009
777369.0	7722558.0	545.7983991630456
775043.0	7722359.0	551.5068908760134
775005.0	7716921.0	563.3609479056229
779669.0	7726178.0	532.0726793447584
777590.0	7722753.0	544.8477610197118

Fonte: O Autor.

No Apêndice A está disponibilizada a classe *MFSRoot* referente às operações principais do Método das Soluções Fundamentais implementado em Java. Aproveitando a estratégia MVC, essa classe pode ser utilizada em qualquer futura expansão da capacidade gráfica do programa ou inserida em outros projetos como uma alternativa de cálculo numérico, desde que os dados sejam preparados e inseridos adequadamente.

A implementação do algoritmo em Scilab seguiu os mesmos princípios de implementação da classe *MFSRoot* em Java. Apesar da linguagem Scilab não ser orientada a objetos, ela absorve os conceitos de operação de dados em matrizes e cálculos algébricos de maneira natural e isso permitiu a implementação de um código semelhante. A principal diferença nessa plataforma é a inviabilidade de aplicar a estratégia MVC, uma vez que, em Scilab, as etapas de entrada, processamento e saída de dados são integradas. Por outro lado, ela é uma excelente ferramenta para implementar cálculos iterativos e sistemas de equações



extensos, poupando tempo de programação e otimizando o teste de hipóteses. O Apêndice B disponibiliza as linhas de código desse programa.

### 3.3 Delimitação e caracterização da área de estudo

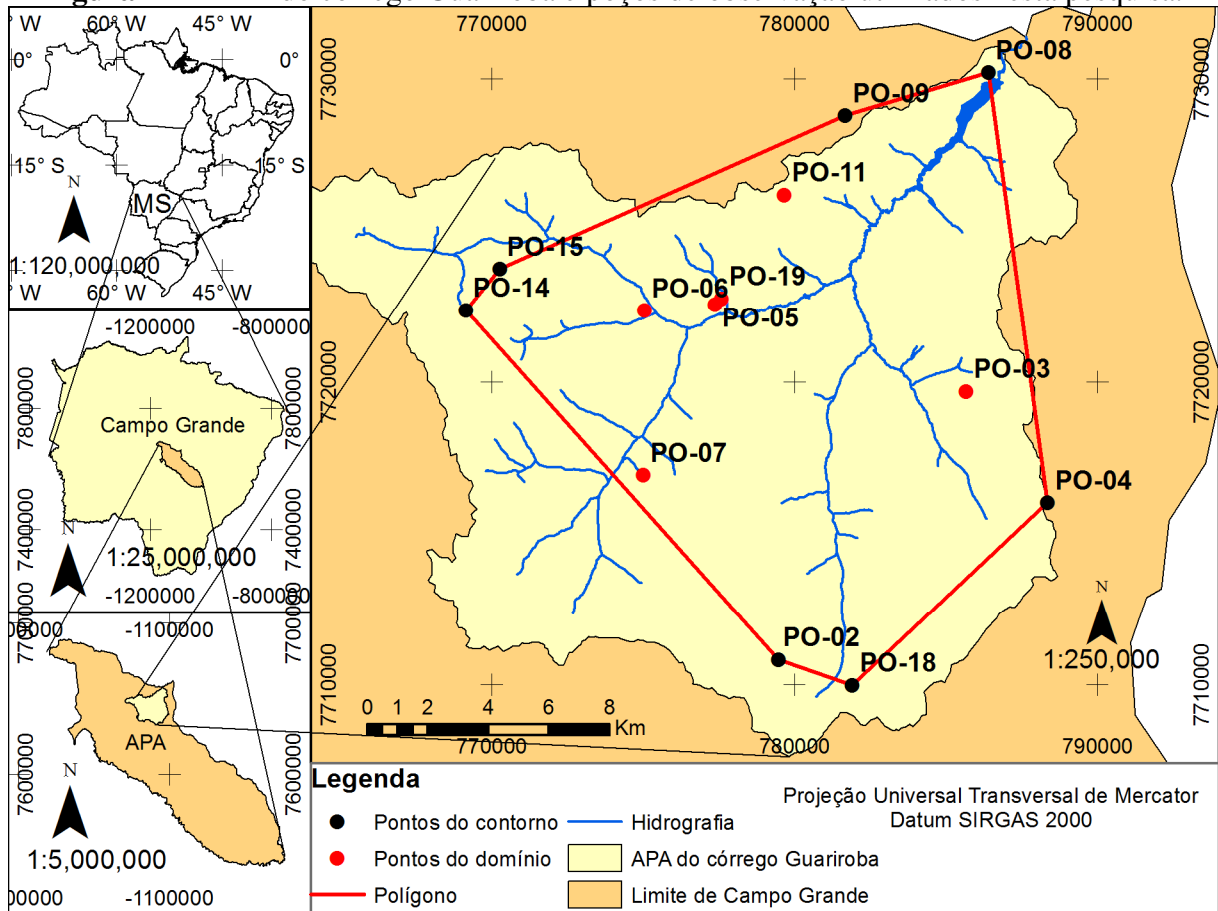
A área de estudo escolhida foi a Área de Proteção do córrego Guariroba na cidade de Campo Grande, no Estado de Mato Grosso do Sul, Brasil, utilizada por Cavazzana, Lastoria e Gabas (2019) no estudo sobre a relação entre o escoamento superficial e subterrâneo da mesma região. Nesse trabalho, os pesquisadores coletaram os valores do nível estático de 13 poços no intervalo de março de 2015 a fevereiro de 2017. Nos dois dias anteriores a cada expedição de coleta de dados, os proprietários dos poços foram orientados a não realizarem bombeamento. Esse cuidado permite a obtenção de representatividade nos valores que representam um domínio onde se assume não existir inhomogeneidade associada à extração ou injeção de água. A figura 17 mostra a disposição espacial dos poços que delimitam o polígono de análise e os poços internos ao domínio, cujos valores de carga hidráulica foram utilizados na validação dos resultados. Os poços de observação, suas coordenadas e seus valores de carga hidráulica estão disponibilizados na tabela 4. Os dados referentes a apenas um dia de amostragem foram utilizados. Nessa tabela, as coordenadas projetadas estão em unidades de metro e associadas ao Datum Horizontal SIRGAS 2000, zona 21S. Esses dados são referentes a um único dia de amostragem (2 de março de 2015) e foram fornecidos por Cavazzana, Lastoria e Gabas (2019).

**Tabela 4** – Dados dos poços de observação utilizados na aplicação do MSF.

	<b>ID</b>	<b>Longitude</b>	<b>Latitude</b>	<b>Cota piezométrica (m)</b>
<b>Poços da fronteira</b>	PO-14	769141	7722366	569.758
	PO-15	770264	7723731	554.128
	PO-09	781682	7728795	518.458
	PO-08	786404	7730206	456.945
	PO-04	788362	7716011	539.921
	PO-18	781915	7709978	561.026
	PO-02	779471	7710815	566.485
<b>Poços internos</b>	PO-03	785672	7719681	510.236
	PO-05	777369	7722558	496.435
	PO-06	775043	7722359	494.108
	PO-07	775005	7716921	508.856
	PO-11	779669	7726178	537.589
	PO-19	777590	7722753	493.908

Fonte: Cavazzana, Lastoria e Gabas (2019)

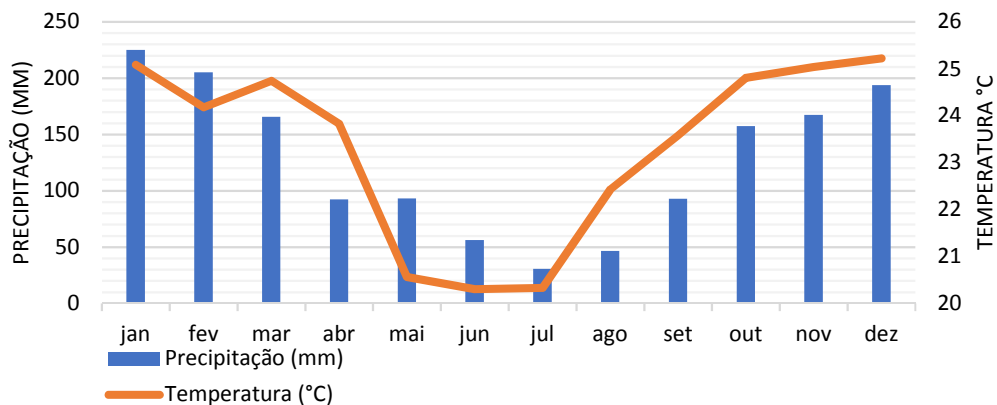
**Figura 17 – APA do córrego Guarairoba e poços de observação utilizados nesta pesquisa.**



Fonte: Cavazzana, Lastoria e Gabas (2019), Campo Grande (2008).

A área de proteção ambiental do córrego do Guarairoba foi criada em 1995 pelo Decreto Municipal nº 7.183/1995 e corresponde a sua bacia hidrográfica, uma área de 360 km<sup>2</sup>. O córrego desagua no rio Botas e ambos estão localizados na bacia continental do rio Paraná. Um reservatório superficial que armazena a vazão do córrego Guarairoba é responsável pelo abastecimento de cerca de 30% da demanda da cidade de Campo Grande. O clima da região é, de acordo com a classificação de Köppen, *Aw*, o que significa que o verão é a estação úmida e o inverno, a estação seca. A figura 18 mostra a distribuição de temperatura e precipitação médias entre 1996 e 2012. (CAMPO GRANDE, 1995, 2008, 2013, CAVAZZANA; LASTORIA; GABAS, 2019).

**Figura 18** – Distribuição de precipitação e temperatura médias na região da APA do córrego do Guariroba entre os anos de 1996 e 2012.



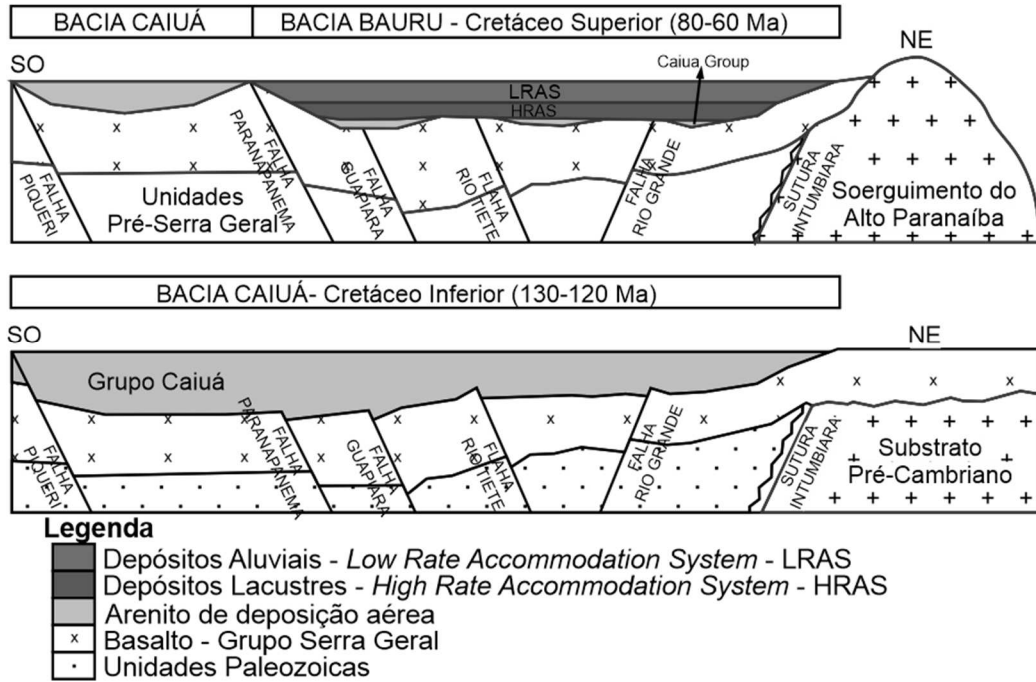
Fonte: Adaptado de Campo Grande (1995, 2008, 2013) e Cavazzana; Lastoria e Gabas (2019).

A caracterização geológica da região é marcada pelos Grupos Caiuá e Bauru de origem sedimentar. O primeiro se formou no Cretáceo Inferior (133-120 milhões de anos – Ma) após processos de deposição aérea sobre as unidades basálticas do Grupo Serra Grande em um fenômeno conhecido como *sand sea* (mar de areia), enquanto o segundo, no Cretáceo Superior (80-60 Ma), sobrepôs-se ao primeiro em algumas regiões a partir de outros processos de deposição (figura 19). Atualmente compõem o Sistema Aquífero Baurú, um extenso aquífero livre exposto à superfície que ocupa cerca de 37% do território do Estado do Mato Grosso do Sul. A análise sedimentológica das unidades do Grupo Caiuá, que predomina na área de estudo (figura 20), mostrou a ocorrência de arenitos de granulometria fina com sedimentos bem selecionados e profundidades de até 280 metros. O Grupo Serra Geral, localizado abaixo dos grupos sedimentares é composto por unidades de basalto fraturado. Na APA do córrego do Guariroba, a erosão causada pelo fluxo superficial de água através da rede de drenagem escavou as formações sedimentares e alcançou as camadas de basalto apenas nos leitos mais profundos. (FERNANDES; COIMBRA, 1994, BATEZELLI, 2010, GOVERNO ESTADUAL DE MATO GROSSO DO SUL, 2010, UECHI; GABAS; LASTORIA, 2017, CAMPO GRANDE, 2018, CAVAZZANA; LASTORIA; GABAS, 2019).

A superfície morfológica da APA do córrego do Guariroba apresenta planícies fluviais separadas por morros suaves com a altitude variando entre 665 e 455 metros. Ocorrem latossolos vermelhos, vermelho-amarelados e neossolos nas regiões mais altas e neossolos fluviais e hidromórficos nos vales. A vegetação natural é característica do bioma Cerrado, espécies arbóreas e arbustivas espalhadas em regiões densas e esparsas adjacentes e a principal

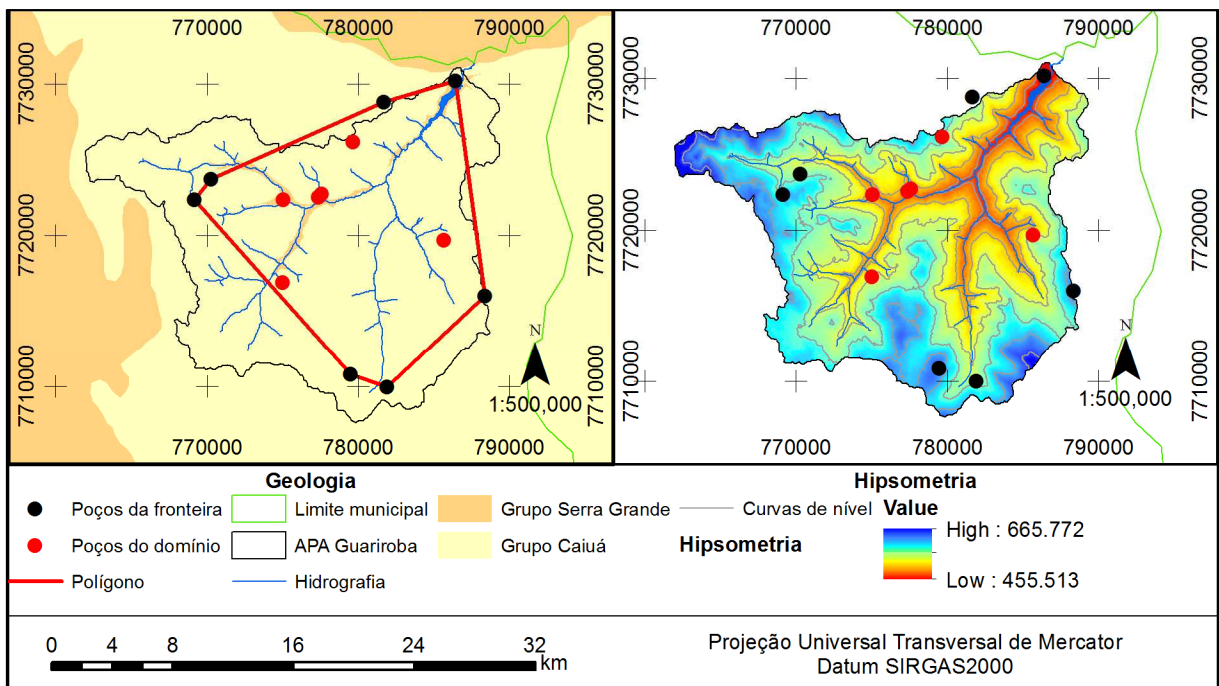
atividade de uso e ocupação do solo é a criação extensiva de gado. (CAMPO GRANDE, 2008, OLIVEIRA et al., 2017, CAPOANE, 2019, CAVAZZANA; LASTTORIA; GABAS, 2019).

**Figura 19** – Esquema da formação do Sistema Aquífero Baurú.



Fonte: Adaptado de Batezelli (2010)

**Figura 20** - Caracterização geológica e morfológica da APA do córrego do Guariroba.



Fonte: Adaptado de TOPODATA (2008), Lacerda Filho; Et Al. (2004), Campo Grande (2008), Cavazzana; Lastoria e Gabas (2019).

### 3.4 Avaliação da influência da localização da fronteira fictícia

A investigação da influência do posicionamento da fronteira fictícia nos resultados da aproximação numérica da carga hidráulica em um exemplo de fluxo de água subterrâneo ocorreu em duas etapas.

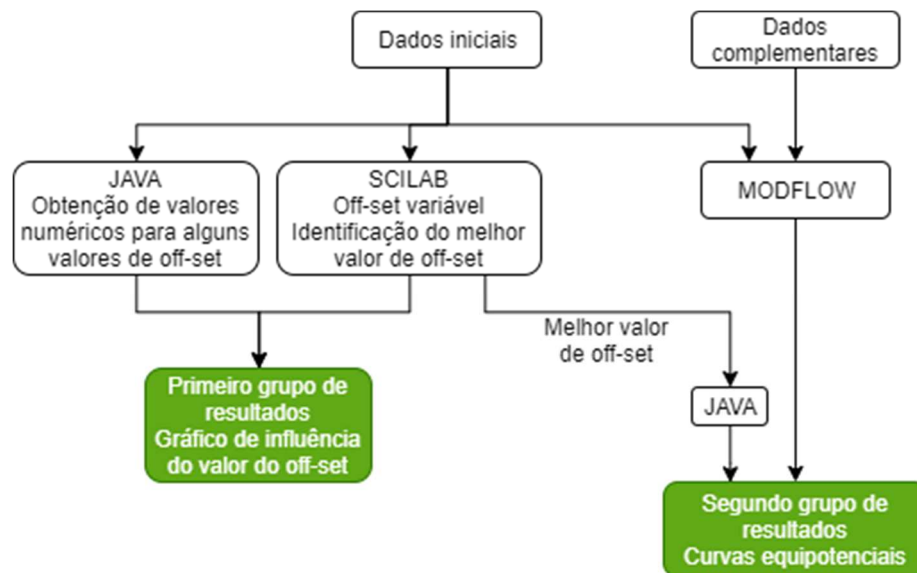
1. **Comparação entre os valores numéricos e os valores de campo para um *off-set* variável;**
2. **Comparação entre as curvas equipotenciais e os valores numéricos obtidos pelo MSF e pelo MODFLOW para o melhor *off-set* determinado na etapa 1.**

Na primeira etapa, o problema definido pelos dados da tabela 4 e pela área de estudo escolhida foi inserido no programa desenvolvido em Scilab (Apêndice B). Nesse programa, o MSF é aplicado várias vezes ao mesmo conjunto de dados iniciais para valores de *off-set* variáveis. A cada iteração, esse valor foi incrementado em 100 unidades, portanto, 100 metros. O valor de partida foi 100 e o final, 200.000 (200 km). Esse intervalo representa a fronteira próxima da região. O valor de 200 km foi escolhido como limite porque, a partir desse valor de *off-set*, os resultados não divergiram significamente. Em cada aplicação do MSF, os valores numéricos calculados foram comparados com os valores de campo para cada poço de observação dentro do domínio. Os erros relativos a cada poço foram armazenados em uma variável a cada passo de aplicação da rotina do método numérico. Ao final dos 2.000 passos, os erros relativos às medidas de campo foram plotados em gráfico. A partir da interpretação desse gráfico foi possível escolher o valor de *off-set* que proporcionou os melhores resultados. Esse valor foi utilizado na próxima etapa. O programa desenvolvido em Java foi utilizado para extrair valores calculados numericamente para determinados valores de *off-set* e para ilustrar a modelagem da aplicação do MSF (figura 15).

Na segunda etapa, o mesmo conjunto de dados foi utilizado na construção de uma malha interna ao domínio em pontos distribuídos em uma malha com densidade de 200x200 metros para gerar curvas equipotenciais de cota piezométrica (carga hidráulica). Esse valor foi escolhido pois a partir dessa densidade foi possível desenhar as curvas que apresentassem aspecto suave e contínuo. O MSF foi aplicado, segundo o melhor valor de *off-set*, aos pontos dessa malha no programa desenvolvido em Java e os valores calculados foram utilizados para gerar um arquivo no formato *raster* e, em seguida, as curvas equipotenciais, através de um Software de Informações Geográficas (SIG). A mesma malha foi simulada no MODFLOW, através da interface UFC-FLOW, para os mesmos dados de condições de contorno. Assim, foi

possível obter um arquivo no formato *raster* e gerar as curvas equipotenciais. O mapa da sobreposição visual dos dois resultados é o produto dessa etapa. A figura 21 mostra a esquematização das etapas.

**Figura 21** – Fluxograma das etapas de obtenção dos resultados aplicadas nesta pesquisa.



Fonte: O Autor.

Para a aplicação do MODFLOW, além dos dados requeridos pelo MSF é necessário especificar dados complementares. Para uma simulação em estado permanente, o MODFLOW requer ainda dados de porosidade efetiva, de condutividade hidráulica, de armazenamento específico e de espessura da camada. A tabela 5 resume os valores adotados. Nessa tabela, o valor da espessura do aquífero foi obtido através da diferença entre o maior e o menor valor de altitude do mapa hipsométrico da região. Os demais valores foram escolhidos segundo a caracterização sedimentar das unidades geológicas do Grupo Caiuá, arenito com sedimentos de pequena granulometria e bem selecionados.

**Tabela 5** – Dados complementares. Tabela com a especificação dos valores adotados na simulação do problema no MODFLOW.

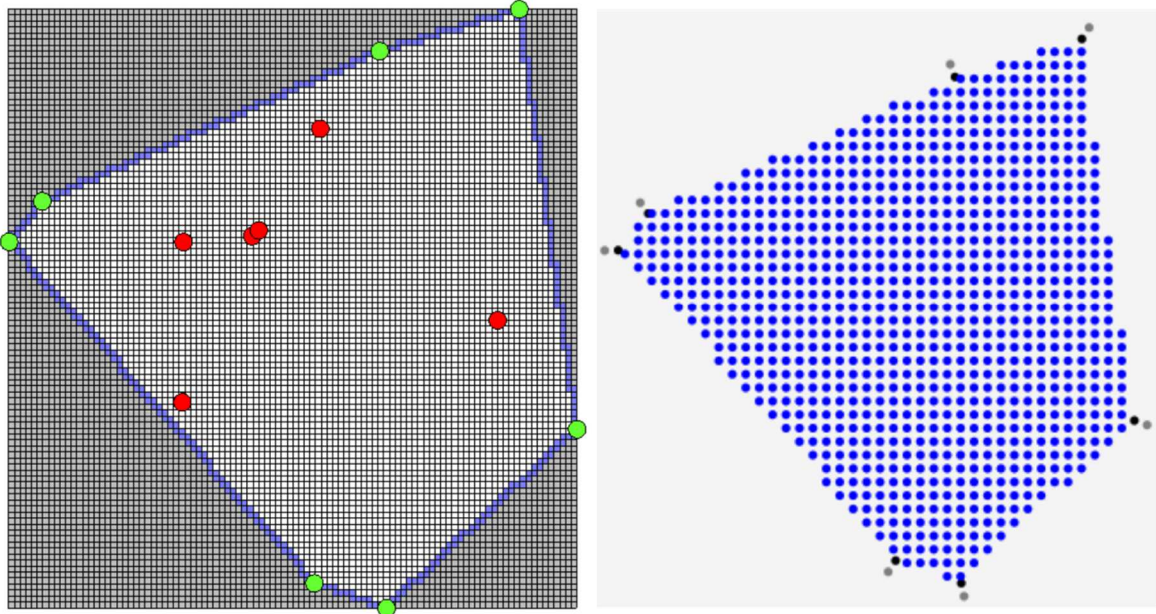
Parâmetro	Unidade	Valor	Referência
Espessura	m	210	Figura 20
$k_x, k_y$	m/s	$6 \times 10^{-7}$	Domenico e Schwartz (1997)
$S_s$	( $m^3/m^3$ )	21%	Johnson (1967)
$n$	( $m^3/m^3$ )	25%	Domenico e Schwartz (1997)

Fonte: Domenico e Schwartz (1997) e Jhonson (1967).



A figura 22 mostra os detalhes da malha utilizada para gerar os *rasters*. Nessa figura, a esquerda, a malha discretizada para aplicação do Método das Diferenças Finitas no MODFLOW. Os pontos verdes representam os poços de fronteira, enquanto os vermelhos, os poços do domínio. Os elementos mais claros são as células ativas, os mais escuros, as inativas e as azuis, as de contorno. A direita, na aplicação do MSF, os pontos pretos representam os poços do contorno, os pontos cinza representam o *off-set* e os pontos azuis, os pontos internos, onde os valores numéricos foram computados. Em todas as simulações foram consideradas as hipóteses que permitem a aplicação da equação de Laplace como operador diferencial do fluxo de água subterrâneo. O MSF não requer a criação de uma malha para ser implementado. Essa malha só foi gerada na figura da direita para que fosse possível desenhar as curvas equipotenciais de carga hidráulica para os resultados obtidos pelo método. O MSF pode ser aplicado a um único ponto ou a um conjunto de pontos mesmo que esses não estejam distribuídos em uma malha regular.

**Figura 22** – Malha de pontos utilizada para calcular numericamente o valor da carga hidráulica em pontos internos ao domínio (Representações sem escala).



Fonte: O Autor.

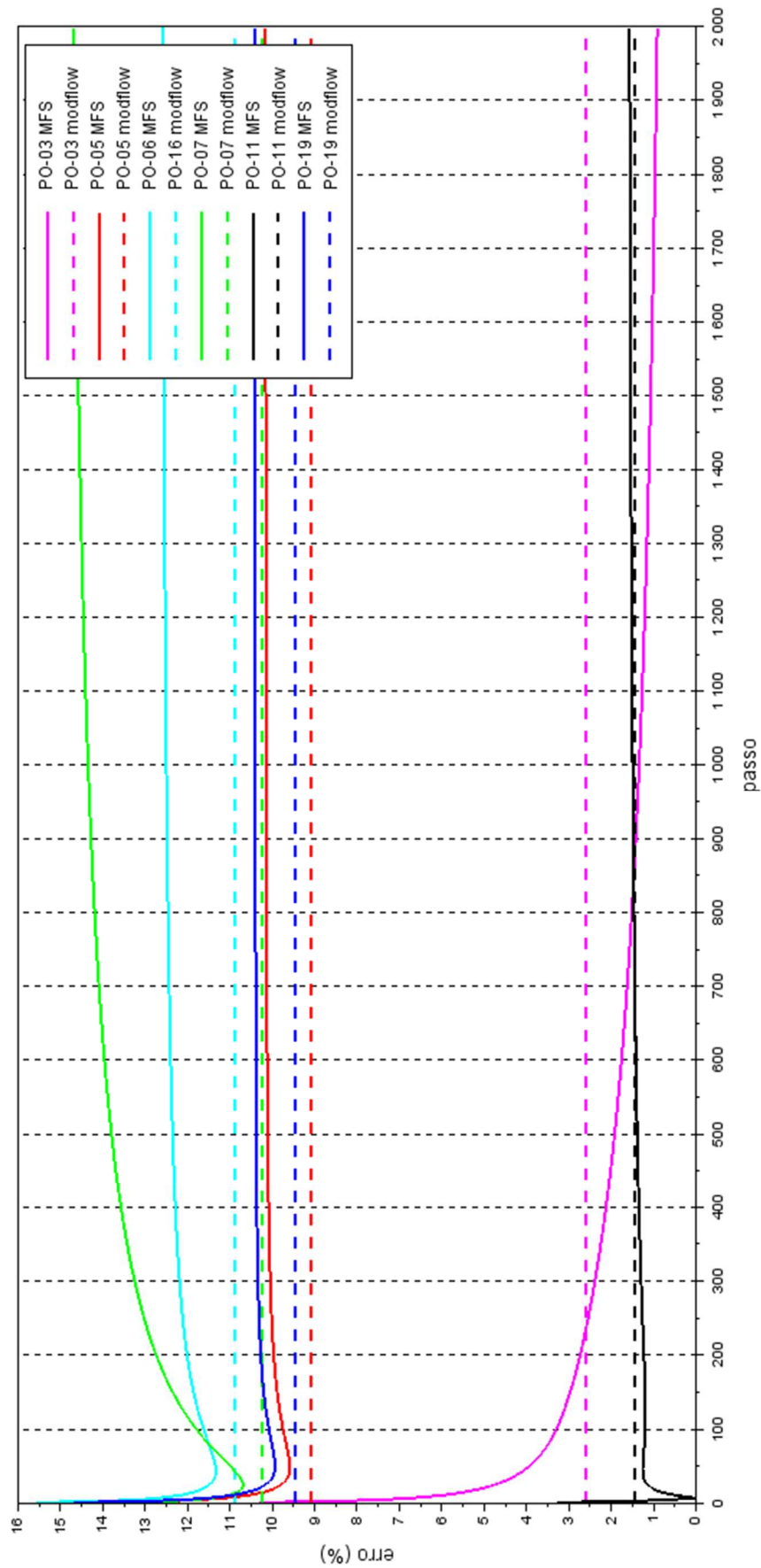
## 4 RESULTADOS E DISCUSSÃO

O gráfico da primeira etapa de análise da influência do valor do *off-set* nos resultados da aplicação do MSF foi disponibilizado detalhadamente na figura 23. De acordo com esse gráfico, os valores de erro relativo para todos os poços de observação decresceram bruscamente no intervalo antes do passo de número 50, o que representa um *off-set* afastado de 5.000 metros dos limites do polígono que delimita a fronteira analisada. O polígono analisado possui área igual a 229,74 km<sup>2</sup> e perímetro igual a 60,43 km. Uma circunferência com essa mesma área possui raio igual a 8,55 km e perímetro igual a 53,73 km. O coeficiente de compacidade do polígono é igual a 0,89. Tomando a circunferência como aproximação da área de estudo é possível afirmar que o valor ótimo de *off-set* pode ser determinado em distâncias menores que 1,7 do raio dessa circunferência. Para esta região de estudo, o valor de 5.000 metros foi escolhido como o valor de *off-set* ótimo.

A partir do passo nº50 os erros aumentaram progressivamente até alcançarem um equilíbrio. Apenas o poço PO-03 apresentou redução progressiva conforme os valores de *off-set* cresciam, corroborando os comentários de Golberg e Chen (1998), segundo os quais o aumento da distância da fronteira fictícia implica em aumento da precisão. Esses pesquisadores, entretanto, avaliaram o MSF em exemplos hipotéticos montados a partir de soluções analíticas. Por outro lado, não foi realizada nenhuma quantificação das incertezas associadas aos dados de entrada. Middlemis, Walker, Peeters, Richardson, Hayes e Moore (2019) apontam que os modelos de fluxo de água subterrâneo incorporam, dentre outros, incertezas associadas aos processos de medida de dados de campo e a sua representatividade. Incertezas nos valores de contorno podem facilmente conduzir os métodos numéricos para resultados que não representam significativamente o sistema real. Isso pode ser facilmente observado ao comparar os resultados do MSF com os resultados do MODFLOW. A tabela 6 condensa esses resultados. De acordo com essa tabela, os valores calculados pelo MODFLOW e para o MSF (segundo o melhor valor global de offset, 5000m) para os poços de observação não diferem consideravelmente, indicando a convergência dos resultados obtidos e endossando a possibilidade da existência de incertezas associadas aos dados de entrada. Os resultados do MSF estão agrupados por valores de *off-set* distintos (O500m, O1000m, O5000m, O10000m, O20000m). Abaixo de cada valor foi calculado o erro relativo (%) à correspondente leitura de campo.



**Figura 23** - Gráfico do erro relativo aos valores de campo para valores de *off-set* variáveis.



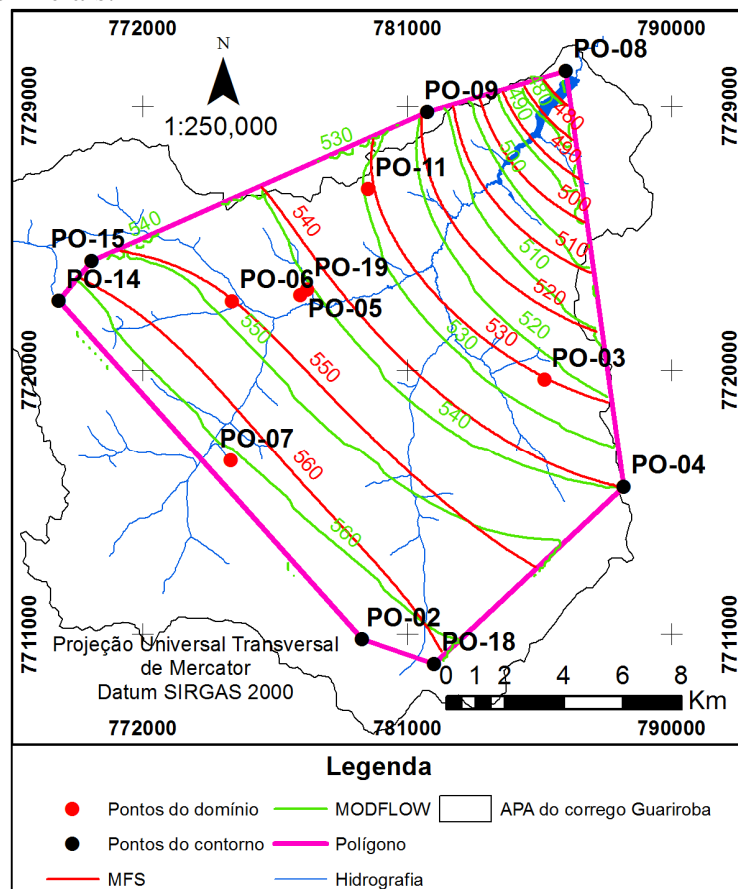
Fonte: O Autor.

**Tabela 6** – Resumo dos resultados obtidos na primeira etapa da análise.

ID	Leitura de campo (m)	MODFLOW	MSF				
			O500m	O1000m	O5000m	O10000m	O20000m
PO-03	<b>510.236</b>	<b>523.573</b>	547.3829	541.3277	<b>530.3037</b>	527.0559	524.1885
		<b>2.61</b>	7.28	6.09	<b>3.93</b>	3.29	2.73
PO-05	<b>496.435</b>	<b>541.574</b>	553.9196	549.2675	<b>544.0272</b>	544.8186	545.9157
		<b>9.09</b>	11.58	10.64	<b>9.59</b>	9.75	9.97
PO-06	<b>494.108</b>	<b>547.898</b>	558.8258	554.6207	<b>550.1547</b>	551.5379	553.4636
		<b>10.88</b>	13.10	12.25	<b>11.343</b>	11.62	12.01
PO-07	<b>508.856</b>	<b>560.956</b>	568.6278	565.2049	<b>564.8224</b>	568.86	573.6855
		<b>10.23</b>	11.75	11.07	<b>11.00</b>	11.79	12.74
PO-11	<b>537.589</b>	<b>529.771</b>	540.3556	535.3639	<b>530.8397</b>	531.1501	530.983
		<b>1.45</b>	0.51	0.41	<b>1.26</b>	1.20	1.23
PO-19	<b>493.908</b>	<b>540.629</b>	553.0966	548.3772	<b>542.9946</b>	543.6865	544.6471
		<b>9.46</b>	11.98	11.03	<b>9.94</b>	10.08	10.27

Fonte: O Autor.

**Figura 24** – Curvas equipotenciais obtidas pelo MODFLOW (verde) e pelo MSF para o mesmo conjunto de dados iniciais.



Fonte: O Autor.

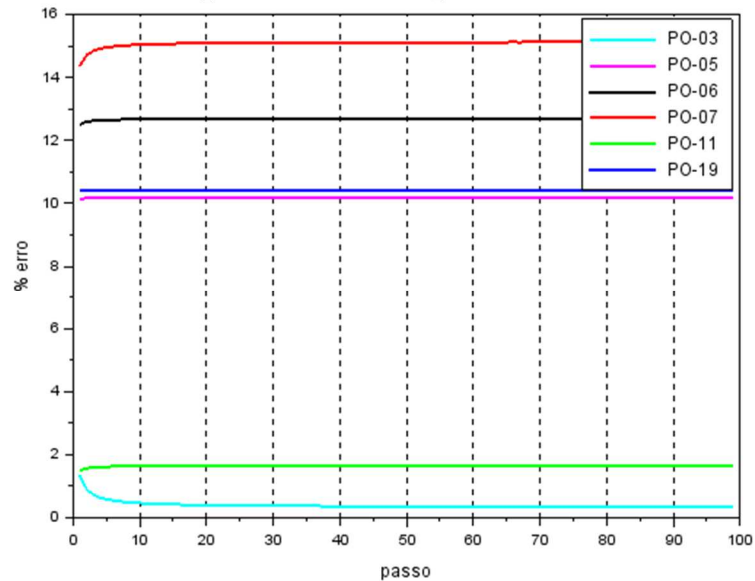
Outro resultado que aponta a convergência dos modelos é o mapa das curvas equipotenciais de carga hidráulica obtidos. A figura 24 mostra a sobreposição entre as curvas equipotenciais geradas pelo MODFLOW e pelo MSF. De acordo com a figura, as diferenças entre as equipotenciais estão limitadas a no máximo 1 km. Apesar disso, as curvas obtidas segundo o MSF são mais suaves e não apresentaram distorções nas proximidades da fronteira.

Cada poço possui dois conjuntos de dados representados no gráfico do Apêndice C. A linha tracejada representa o erro obtido pelo MODFLOW, enquanto a linha contínua representa o erro obtido pelo MSF. A linha contínua varia conforme o passo aumenta, pois esse valor influencia a distância do *off-set* considerado na execução das operações algébricas e na construção das matrizes. É possível observar que ambos os erros se aproximam nas proximidades do *off-set* ótimo. Para os poços PO-11 e PO-03 os resultados do MSF são inclusive mais precisos que os correspondentes obtidos pelo MODFLOW.

A tentativa de calcular os erros do gráfico do Apêndice C para valores de *off-set* maiores que 400 km levou à construção de matrizes mal condicionadas. Isso converge com o comportamento do MSF relatado por Golberg e Chen (1998) e Liu (2012). A determinação de aproximações numéricas para valores de *off-set* superiores foi limitada pela capacidade do programa resolver os sistemas lineares representados por esse tipo de matriz. Apesar disso, foi investigada a aplicação de valores extremos. A figura 25 mostra o resultado do mesmo teste realizado na confecção do gráfico do Apêndice C com a diferença que os valores de *off-set* variaram entre 100 e 10.000 km em passos de 100 km.

Valores extremos conduziram a valores de erro convergentes. Os melhores resultados são obtidos no intervalo apontado anteriormente, para valores de *off-set* menores que 1,7 vezes o raio da circunferência de mesma área do polígono analisado. O único poço que não apresenta esse comportamento é o poço PO-03, para o qual o erro converge para o valor de 0,5%. Contudo, o erro global, aqui definido como a soma dos erros relativos de todos os poços para o mesmo valor de *off-set*, é ótimo para 5.000m de distância. A partir de 10.000 km o computador aborta as operações, pois entende que a partir desse valor a matriz é não invertível, em função de números de condicionamento elevados.

**Figura 25** – Erros relativos às leituras de campo para valores de *off-set* variando entre 100 e 10.000 km em passos de 100 km para cada um dos poços localizados no interior do domínio.



Fonte: O Autor.

A utilização do Java e do Scilab para implementar o Método das Soluções Fundamentais aproveitou as principais características de cada uma das linguagens de programação. Ambas as plataformas forneceram ferramentas de tratamento de operações algébricas suficientes para a realização da tarefa. O Scilab mostrou ser mais simplificado na implementação das etapas puramente matemáticas, enquanto o Java promoveu mais ferramentas para a criação de uma interface gráfica. A sequência natural de concepção de um algoritmo robusto apontou para a implementação em Scilab, onde as rotinas foram testadas e os eventuais erros foram identificados e corrigidos. Posteriormente, o mesmo algoritmo foi implementado em Java, onde foi possível incorporar elementos gráficos e a estratégia Model-View-Controller, proporcionando uma interface intuitiva e um ponto de partida para a construção de um programa que incorpore mais funcionalidades, principalmente aquelas relacionadas à representação visual e à interação com o usuário.

## 5 CONCLUSÕES

- As linguagens de programação Java e Scilab apresentaram as ferramentas necessárias para a implementação do Método das Soluções Fundamentais. Esse método numérico possui simplicidade na montagem do sistema de equações e na preparação dos dados iniciais.
- O posicionamento da fronteira fictícia influenciou os resultados das aproximações numéricas obtidos através do Método das Soluções Fundamentais. O valor de *off-set* que proporcionou o menor erro global foi igual a 1,7 vezes o raio da circunferência de mesma área da região analisada. Para valores de *off-set* muito maiores que as dimensões da área de estudo, os erros convergiram para valores constantes, evidenciando a estabilidade do método.
- Os resultados do MSF convergiram com os resultados do MODFLOW para uma malha com mesma densidade de nós. As divergências observadas entre os valores numéricos e os valores de campo sugerem a necessidade da realização de um estudo dos erros associados às medidas, para que a confiança e o risco associado possam ser considerados.

## REFERÊNCIAS

- ALVES, C. J. S.; CHEN, C. S. A new method of fundamental solutions applied to nonhomogeneous elliptic problems. **Advances in computational mathematics**, v. 23, p. 125-142, 2005.
- APACHE COMMONS. **Apache Commons Math**. Apache Commons, 2016. Disponível em: “<https://commons.apache.org/proper/commons-math/>”. Acesso em: dezembro de 2019.
- ARAÚJO, R. S. A. **Metodologias baseadas no mínimo resíduo para a escolha do parâmetro de forma utilizado em RBF’s aplicadas no método assimétrico livre de malhas (MESHLESS)**. 2016. 208 f. Tese (Doutorado em Engenharia Civil: Recursos Hídricos) – Centro de Tecnologia, Universidade Federal do Ceará, Fortaleza, 2016.
- ASCÊNCIO, A. F. G.; CAMPOS, E. A. V. **Fundamentos da programação de computadores**. 2. ed., São Paulo: Pearson Prentice Hall, 2007.
- BARRERO-GIL, A. The method of fundamental solutions without fictitious boundary for solving Stokes problems. **Computers & Fluids**, v. 62, p. 86-90, 2012. (ISSN 0045-7930)
- BATEZELLI, A. Arcabouço tectono-estratigráfico e evolução das Bacias Caiuá e Bauru no Sudeste brasileiro. **Revista Brasileira de Geociências**, v. 40, n. 2, p. 265-285, jun, 2010.
- BOAS, M. L. **Mathematical Methods in the Physical Sciences**. 2 ed. Joh Willey & sons, 1966.
- BOGOMOLNY, A. Fundamental Solutions Method for Elliptic Boundary Value Problems. **SIAM Journal on Numerical Analysis**, v. 22, n. 4, p. 644-669, 1985.
- BOYCE, W. E.; DIPRIMA, R. C. **Equações diferenciais elementares e problemas de valores de contorno**. Tradução e revisão: Valéria de Magalhães Iório. 9 ed. Rio de Janeiro: LTC, 2010. (ISBN 978-5-216-1756-3)
- BREBBIA, C. A. **The boundary element method for engineers**. London: Pentech Press, 1978
- BREBBIA, C. A.; DOMINGUEZ, J. **Boundary Elements: An introductory course**. Boston, Southampton: WITPress Computational Mechanics Publications, 1992.
- BREBBIA, C. A.; TELLES, J. C. F.; WROBEL, L. C. **Boundary Element Techniques**. Springer-Verlag: Berlin, Heidelberg, New York, Tokyo, 1984.
- CAMPO GRANDE. Prefeitura. **Decreto nº 7.183, de 21 de setembro de 1995**. Campo Grande, 1995.
- CAMPO GRANDE. Prefeitura. **Plano de Manejo da Área de Proteção Ambiental dos Mananciais do Córrego Guariroba – APA do Guariroba**. Campo Grande, 2008.
- CAMPO GRANDE. Prefeitura. **Plano Municipal de Saneamento Básico de Campo Grande**. Campo Grande, 2013.

CAPOANE, V. Susceptibility to erosion in the Corrego Guariroba watershed, Campo Grande, Mato Grosso do Sul. **Acta Brasiliensis**, v. 3, n. 2. p. 49-55, 2019.

CAVAZZANA, G. H.; LASTORIA, G.; GABAS, S. G. Surface-groundwater interaction in unconfined sedimentary aquifer system in the Brazil's tropical wet region. **RBRH**, Porto Alegre, v. 24, n. 8, 2019.

CHAPRA, S. C.; CANALE, R. P. **Métodos numéricos para engenharia**. Tradução Técnica: Helena Castro. 5 ed. McGraw-Hill, 2008. (ISBN 978-85-86804-87-8)

CHEN, C. S.; CHO, H.; GOLBERG, M. A. Some comments on the ill-conditioning of the method of fundamental solutions. **Engineering Analysis with Boundary Elements**, n. 30, p. 405-410, 2006.

CHEN, C. S.; KARAGEORGHIS, A.; SMYLIRS, Y. S. **The Method of Fundamental Solutions – A Meshless Method**. Dynamic Publishers, 2008.

CHEN, W.; WANG, F. Z. A method of fundamental solutions without fictitious boundary. **Engineering Analysis with Boundary Elements**, v. 34, n. 5, p. 530-532, 2010.

CHOW, T. L. **Mathematical methods for physicists: a concise introduction**. Cambridge University Press, 2003

DAVIS, S. N. Porosity and permeability of natural materials in flow through porous materials. *In*: **R.J.M. DeWiest (ed.)** Academic Press, New York, p. 54-89, 1969.

DOMENICO, P. A.; SCHWARTZ, W. **Physical and chemical hydrogeology**. 2 ed. John Wiley & Sons, Inc, 1997.

ECKEL, B. **Thinking in Java**. 4 ed. Prentice Hall, 2006. (ISBN 0-13-187248-6)

FERNADES, L. A.; COIMBRA, A. M. O Grupo Caiuá (Ks): Revisão estratigráfica e contexto deposicional. **Revista Brasileira de Geociências**, v. 24, n. 3, p. 164-176, setembro, 1994.

FETTER, C. W. **Applied hydrogeology**. 4 ed. Prentice-Hall, 2001.

GOLBERG, M. A.; CHEN, C. S. The method of fundamental solutions for potential, Helmholtz and diffusion problems. **Boundary integral methods: numerical and mathematical aspects 1**, p. 103-176, 1998.

GOLBERG, M. A.; CHEN, C. S.; MULESHKOV, A. S. The method of fundamental solutions for time-dependent problems. **Transactions on Modelling and Simulation**, v. 22, 1999. (ISSN 1743-355X)

GONÇALVES, A. D. M. A.; LIBRADI, P. L. Análise da determinação da condutividade hidráulica do solo pelo método do perfil instantâneo. **R. Bras. Ci. Solo**, v. 37, p. 1174-1184, 2013.

GOVERNO DO ESTADO DE MATO GROSSO DO SUL. Secretaria de Estado de Meio Ambiente, do Planejamento, da Ciência e Tecnologia. **Plano estadual de recursos hídricos de Mato Grosso do Sul**. Campo Grande: Editora UEMS, 2010.

GU, Y.; FAN, C. M.; XU, R. P. Localized method of fundamental solutions for large-scale modeling of two-dimensional elasticity problems. **Applied Mathematics Letters**, v. 93, p. 8-14, 2019. (ISSN 0893-9659)

GINGOLD, R. A.; MORAGHAN, J. J. Smooth particle hydrodynamics: theory and applications to non-spherical stars. **Man. Not. Roy. Astron. Soc.**, n. 181, p.375-389, 1977.

HAITJEMA, H. M. **Analytic element modeling of groundwater flow**. London, UK: Academic Press Limited, 1995. (ISBN 0-12-316550-4)

HAITJEMA, H. M. Comparing a three-dimensional and a Dupuit-Forchheimer solution for a circular recharge area in a confined aquifer. **J. Hydrology**, n.91, p.83-101, 1987.

JAZAYERI, A.; WERNER, A. D. Boundary condition nomenclature confusion in groundwater flow modeling. **Groundwater**, v. 57, n. 5, 2019.

JOHNSON, A. I. Specific yield-compilation of specific yields for various materials. **U.S. Geol. Survey Water Supply Paper**, 1662-D, 1967. (74 p.)

JOHNSON, A. I.; MORRIS, D. A. Physical and hydrologic properties of water bearing deposits from core holes in the Las Banos-Kettleman City area, California. **U.S. Geological Survey Open. File Rept.** Denver, Colo., 1962.

LACERDA FILHO, J. V.; VALENTE, C. R.; LOPES, R. C., OLIVEIRA, I. W. B.; SACHS, L. L. B.; SILVA, V. A.; BATISTA, I. H., 2004. Folha SF.21 – Campo Grande. *In*: SCHOBENHAUS, C.; GONÇALVES, J. H.; SANTOS, J. O. S.; ABRAM, M. B.; LEÃO NETO, R.; MATOS, G. M. M.; VIDOTTI, R. M.; RAMOS, M. A. B.; JESUS, J. D. A. (eds). **Carta Geológica do Brasil ao Milionésimo**, Sistema de Informações Geográficas. Programa Geologia Do Brasil. CPRM, Brasília. CD-ROM.

LEVEQUE, R. J. **Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems**. Society for Industrial and Applied Mathematics (SIAM): Philadelphia, 2007. (ISBN 978-898716-29-0)

LIU, C. S. An equilibrated method of fundamental solutions to choose the best source points for the Laplace equation. **Engineering Analysis with Boundary Elements**, v. 36, n. 8, p. 1235-1245, 2012. (ISSN 0955-7997)

LIU, G. R.; GU, Y. T. **An introduction to meshfree methods and their programming**. Springer, 2005.

LIU, Q. G.; SARLER, B. Non-Singular Method of Fundamental Solutions for Two-Dimensional Isotropic Elasticity Problems. **CMES - Computer Modeling in Engineering and Sciences**, v. 91, n. 4, p. 235-266, 2013.



- LOTTI, R. S.; MACHADO, A. W.; MAZZIEIRO, E. T.; LANDRE JÚNIOR, J. Aplicabilidade científica do método dos elementos finitos. **Revista Dental Press Ortodon Ortop Facial**. v. 11, n. 2, p. 35-43, mar/abr., 2006.
- LUCY, L. A numerical approach to testing the fission hypothesis. **Astron. J.**, n. 82, p. 1013-1024, 1977.
- MCDONALD, M. G.; HARBAUGH, A. W. The History of MODFLOW. **Groundwater**, v. 41, n. 2, march-april 2003.
- MIDDLEMIS, H.; WALKER, G.; PEETERS, L.; RICHARDSON, S.; HAYES, P; MOORE, C. **Groundwater modelling uncertainty** – implications for decision making. Summary report of the national groundwater modelling uncertainty workshop, 10 July 2017, Sydney, Australia. Flinders University, National Centre for Groundwater Research and Training, Australia, 2019.
- OLIVEIRA, A. K. M.; FERNANDES, V.; PIRAJÁ, R. V.; SILVA, M. H. S. Multitemporal evaluation of landscape of the Area of Environmental Protection (APA) of the water catchment Guariroba Stream, Campo Grande, Mato Grosso do Sul, by means of satellite images. **R.Ra'eGa**, v. 42, p. 8-20, dez./2017. Curitiba, 2017.
- ORACLE. **Oracle Technology Network**. Topics. New to Java. Java Basics. Oracle, 2019. Disponível em: “<https://www.oracle.com/technetwork/topics/newtojava/downloads/index.html>”. Acesso em: dezembro de 2019.
- PINTO-COELHO, R. M.; HAVENS, K. **Gestão de Recursos Hídricos em Tempos de Crise**. Artmed Editora, 2016.
- QUADROS, R. S.; BORTOLI, A. L. **Fundamentos de cálculo numérico para engenheiros**. Porto Alegre, 2009.
- REDDY, J. N. **An introduction to the Finite Element Method**. 3.ed. McGraw-Hill, 2006.
- SCILAB. **Open source**. Scilab, 2019. Disponível em: “<https://www.scilab.org/about/scilab-open-source-software>”. Acesso em: dezembro de 2019.
- SCIORE, E. **Java program design: principles, polymorphism, and patterns**. Newton, MA, USA: Apress, 2019.
- SEBESTA, R. W. **Concepts of programming languages**. 9 ed. Pearson, 2009. (ISBN 978-0-13-607347-5)
- SEGERLIND, L. J. **Applied finite element analysis**. John Wiley & Sons, 1984.
- STRACK, O. D. L. Three-dimensional streamlines in Dupuit-Forchheimer models. **Water Resour. Res.**, n. 20, p. 812-822, 1984.
- TOPODATA. **20S555ZN**. Topodata, 2008. Disponível em: “<http://www.dsr.inpe.br/topodata/index.php>”. Acesso em: novembro de 2019.

UECHI, D. A.; GABAS, S. G.; LASTORIA, G. Analysis of the heavy metals in the Bauru Aquifer System in Mato Grosso do Sul. **Eng. Sanit. Ambiental**, v. 22, n. 1, p. 155-167, jan/fev 2017.

WANG, F. Z.; ZHENG, K. H. The method of fundamental solutions for steady-state groundwater flow problems. **Journal of the Chinese Institute of Engineers**, 2015. (DOI: 10.1080/02533839.2015.1082936)

YOUNG, D. L.; TSAI, C. C.; CHEN, C. W.; FAN, C. M. The method of fundamental solutions and condition number analysis for inverse problems of Laplace equation. **Computers & Mathematics with Applications**, v. 55, n. 6, p. 1189-1200, 2008. (ISSN 0898-1221)

ZIJL, W.; DE SMEDT, F.; EL-RAWY, M.; BATELAAN, O. **The Double Constraint Inversion Methodology**: Equations and applications in forward and inverse modeling of groundwater flow. SpringerBriefs in Applied Sciences and Technology, 2018. (ISBN 2191-5318)

## APÊNDICE A – CLASSE *MFSROOT*

Esta classe foi escrita em Java e implementa a aplicação básica do Método das Soluções Fundamentais. Para sua correta operação é necessária a existência de classes acessórias que preparem os dados de entrada no formato dos vetores que são utilizados como parâmetros nos métodos internos.

```
import org.apache.commons.math3.linear.LUDecomposition;
import org.apache.commons.math3.linear.MatrixUtils;
import org.apache.commons.math3.linear.RealMatrix;

public class MFSRoot {

    public MFSRoot() {
    }

    public double phiFS (double[] point, double[] pointi) {
        double phi = -(Math.log(Math.sqrt(Math.pow((point[0]-
pointi[0]), 2)+Math.pow((point[1]-pointi[1]),2))))/(2*Math.PI);
        return phi;
    }

    public double[] phiLine(double[] point, double[][] offPointVec) {
        int nSize = offPointVec.length;
        double[] line = new double[nSize];
        for (int i=0; i<nSize; i++) {
            line[i] = phiFS(point,offPointVec[i]);
        }
        return line;
    }

    public double phiDFS (double[] point, double[] pointi) {
        double phi = ((-(pointi[0]-
point[0])/(2*Math.PI*Math.pow((Math.pow((pointi[0]-point[0]),
2)+Math.pow((pointi[1]-point[1]), 2)),1))) * point[4]) +
            ((-(pointi[1]-
point[1])/(2*Math.PI*Math.pow((Math.pow((pointi[0]-point[0]),
2)+Math.pow((pointi[1]-point[1]), 2)),1))) * point[5]);
        return phi;
    }

    public double[] dphiLine(double[] point, double[][] offPointVec) {
        int nSize = offPointVec.length;
        double[] dline = new double[nSize];
        for (int i=0; i<nSize; i++) {
            dline[i] = phiDFS(point,offPointVec[i]);
        }
        return dline;
    }
}
```

```

    public double[][] mainMatrix(double[][] pointVec, double[][]
offPointVec){
        int mSize = pointVec.length;
        double[][] ma = new double[mSize][mSize];
        for (int i=0; i<mSize; i++) {
            if(pointVec[i][2]==1||pointVec[i][2]==0) {
                ma[i] = phiLine(pointVec[i], offPointVec);
            }else{
                if(pointVec[i][2]==2) {
                    ma[i] = dphiLine(pointVec[i], offPointVec);
                }
            }
        }
        return ma;
    }

    public double[][] lambdaVec(double[][] maMatrix, double[]
boundaryConditions) {
        int n = maMatrix.length;
        double[][] lambda = new double[n][0];
        RealMatrix bc =
MatrixUtils.createColumnRealMatrix(boundaryConditions);
        RealMatrix ma = MatrixUtils.createRealMatrix(maMatrix);
        RealMatrix maInverse = new
LUdecomposition(ma).getSolver().getInverse();
        RealMatrix rmLambda = maInverse.multiply(bc);
        lambda = rmLambda.getData();
        return lambda;
    }

    public double[][] numericValue(double[][] domainPointMatrix,
double[][] maDomainMatrix, double[][] maBoundaryMatrix, double[]
boundaryConditions){
        double[][] lambdaVec = lambdaVec(maBoundaryMatrix,
boundaryConditions);
        int n = maDomainMatrix.length;
        double [][] ans = new double[n][3];
        RealMatrix dm = MatrixUtils.createRealMatrix(maDomainMatrix);
        RealMatrix lambdaRM = MatrixUtils.createRealMatrix(lambdaVec);
        RealMatrix ansRM = dm.multiply(lambdaRM);
        double[][] temp = ansRM.getData();
        for (int i=0; i<n; i++) {
            ans[i][0] = domainPointMatrix[i][0];
            ans[i][1] = domainPointMatrix[i][1];
            ans[i][2] = temp[i][0];
        }
        return ans;
    }
}
}

```

## APÊNDICE B – IMPLEMENTAÇÃO DO MSF EM SCILAB

É importante notar que esse código realiza o cálculo do erro entre os valores de campo e os valores calculados pelo MSF para um conjunto de dados que correspondem aos pontos do contorno e seus valores de condição de contorno, aos pontos do domínio com seus valores de campo e um valor de *off-set* variável. Esse código foi utilizado especificamente para a obtenção dos resultados apresentados nesta pesquisa.

```
//MESHLESS FUNDAMENTAL SOLUTION METHOD (FSM)
//Autor: Guilherme Costa
//      Erlandson Vasconcelos
//Contato: ***
//UFC - Laboratório de Hidráulica Computacional - ago.2019

/*
Descrição: Rotina para comparar os valores calculados pelo MFS e os
valores
observados em campo para uma serie de valores de offset em um intervalo
i0, iF
*/

//Reseta o console
clear();
clc();
clf(0);
clf(1);

//#[1] Importacao de pontos do contorno e do domínio (pontos de interesse)

/*Modificar o caminho do arquivo para o arquivo correspondente
Observar formatacao:
1- O arquivo será do tipo xls, arquivo do excel
2- Cada coluna correspondera a um tipo de informacao:
    [1] coordenadas x
    [2] coordenadas y
    [3] tipo de condicao de contorno
    [4] valor da condicao de contorno
    [5] coordenada x do vetor unitario normal ao ponto
    [6] coordenada y do vetor unitario normal ao ponto
*/

sheet = readxls("C:\...\boundaryWells.xls")
s1 = sheet(1)
boundaryMatrix(:,1) = s1(:,1) //coordenadas x
boundaryMatrix(:,2) = s1(:,2) //coordenadas y
boundaryMatrix(:,3) = s1(:,3) //tipo de condicao de contorno (1 ou 2)
boundaryMatrix(:,4) = s1(:,4) //valor da condicao de contorno
boundaryMatrix(:,5) = s1(:,5) //coordenada x do vetor unitario normal ao
ponto de contorno
```

```
boundaryMatrix(:,6) = s1(:,6) //coordenada y do vetor unitario normal ao
ponto de contorno
```

```
sheet2 = readxls("C:\...\domainWells.xls")
s21 = sheet2(1)
domainMatrix(:,1) = s21(:,1) //coordenadas x
domainMatrix(:,2) = s21(:,2) //coordenadas y
domainMatrix(:,3) = s21(:,4) //valor de leitura
```

```
///[2] Funcoes para a criacao das matrizes principais do metodo
```

```
function phi=phiFS(x, xi, y, yi)
    phi = log(sqrt((x-xi)^2+(y-yi)^2))/(2*pi)
endfunction
```

```
function line=phiLine(point, pointVec)
    for i=1:size(pointVec)(1)
        line(i) = phiFS(point(1),pointVec(i,1),point(2),pointVec(i,2))
    end
endfunction
```

```
function phi=phiDFS(x1, x2, y1, y2, n1, n2)
    phi = (-(x2-x1)/(2*pi*((x2-x1)^2+(y2-y1)^2)))*n1+(-(y2-
y1)/(2*pi*((x2-x1)^2+(y2-y1)^2))*n2
endfunction
```

```
function line=dphiLine(point, pointVec, n)
    for i=1:size(pointVec)(1)
        line(i) =
phiDFS(point(1),pointVec(i,1),point(2),pointVec(i,2),n(1),n(2))
    end
endfunction
```

```
///[3] Funcoes para obtencao dos vetores normais aos pontos do contorno
```

```
function normalAnsVector=getNormalVectors(boundaryMatrix)
    n = size(boundaryMatrix)(1)
    ///[3.1] Calculo dos vetores unitarios
    for i=1:n
        /*
        temp[1] - modulo dos vetores
        temp[2] - coordenada x trasportada para origem
        temp[3] - coordenada y trasportada para origem
        temp[4] - coordenada x do vetor unitario
        temp[5] - coordenada y do vetor unitario
        */
        if i~=n
            then
                temp(i,1) = (((boundaryMatrix(i,1)-
boundaryMatrix(i+1,1))^2)+((boundaryMatrix(i,2)-
boundaryMatrix(i+1,2))^2))^(1/2)
                temp(i,2) = boundaryMatrix(i+1,1) - boundaryMatrix(i,1)
                temp(i,3) = boundaryMatrix(i+1,2) - boundaryMatrix(i,2)
            else
```

```

        temp(i,1) = (((boundaryMatrix(i,1)-
boundaryMatrix(1,1))^2)+((boundaryMatrix(i,2)-
boundaryMatrix(1,2))^2))^(1/2)
        temp(i,2) = boundaryMatrix(1,1) - boundaryMatrix(i,1)
        temp(i,3) = boundaryMatrix(1,2) - boundaryMatrix(i,2)
    end
    temp(i,4) = temp(i,2)/temp(i,1)
    temp(i,5) = temp(i,3)/temp(i,1)
end

    //[3.2] Determinacao dos vetores normais aos pontos, com base em seus
vizinhos
    //anterior e posterior

for i=1:n
    if i~=1
        then
            normalVector(i,1) = temp(i-1,4) - temp(i,4)
            normalVector(i,2) = temp(i-1,5) - temp(i,5)
        else
            normalVector(i,1) = temp(n,4) - temp(i,4)
            normalVector(i,2) = temp(n,5) - temp(i,5)
        end

        //Pontos colineares
        if normalVector(i,1)==0 & normalVector(i,2)==0
            then
                angle = acos(temp(i,4))
                normalVector(i,1) = cos(angle+%pi/2)
                normalVector(i,2) = sin(angle+%pi/2)
            end

            module = ((normalVector(i,1)^2)+(normalVector(i,2)^2))^(1/2)
            normalAnsVector(i,1) = normalVector(i,1)/module
            normalAnsVector(i,2) = normalVector(i,2)/module
        end

        //Pontos colineares e alinhados a (0,-1) (tratamento de exceção)

for i=1:n
    if i~=1
        then
            if(temp(i,5)==-1 & temp(i-1,5)==-1)
                then
                    normalAnsVector(i,1) = cos(2*%pi)
                    normalAnsVector(i,2) = 0
                end
            else
                if(temp(i,5)==-1 & temp(n,5)==-1)
                    then
                        normalAnsVector(i,1) = cos(2*%pi)
                        normalAnsVector(i,2) = 0
                    end
                end
            end
        end
    end
end

```

```

    end
endfunction

```

```

function centroidCoordinates=getCentroidCoordinates(boundaryMatrix)
    n = size(boundaryMatrix)(1)
    tempCoordinates(1) = 0
    tempCoordinates(2) = 0

    for i=1:n
        tempCoordinates(1) = tempCoordinates(1) + boundaryMatrix(i,1)
        tempCoordinates(2) = tempCoordinates(2) + boundaryMatrix(i,2)
    end

    centroidCoordinates(1,1) = tempCoordinates(1)/n
    centroidCoordinates(1,2) = tempCoordinates(2)/n
endfunction

```

```

function offMatrix=getOffSetMatrix(offset, boundaryMatrix)
    n = size(boundaryMatrix)(1)
    centroidCoordinates = getCentroidCoordinates(boundaryMatrix)
    normalVector = getNormalVectors(boundaryMatrix)

    for i=1:n
        offMatrix(i,1) = boundaryMatrix(i,1) + offset*normalVector(i,1)
        offMatrix(i,2) = boundaryMatrix(i,2) + offset*normalVector(i,2)

        ncx1 = offMatrix(i,1) - boundaryMatrix(i,1)
        ncy1 = offMatrix(i,2) - boundaryMatrix(i,2)
        ncx2 = centroidCoordinates(1,1) - boundaryMatrix(i,1)
        ncy2 = centroidCoordinates(1,2) - boundaryMatrix(i,2)
        m1 = sqrt(ncx1*ncx1+ncy1*ncy1)
        m2 = sqrt(ncx2*ncx2+ncy2*ncy2)

        cos12 = acos(((ncx1*ncx2)+(ncy1*ncy2))/(m1*m2))

        if cos12<%pi/2
            then
                offMatrix(i,1) = boundaryMatrix(i,1) -
offset*normalVector(i,1)
                offMatrix(i,2) = boundaryMatrix(i,2) -
offset*normalVector(i,2)
            end
        end
    end
endfunction

```

```

///#[4] Funcoes da aplicacao do metodo

```

```

function mMatrix=mainMatrix(boundaryMatrix, offsetMatrix)
    for i=1:size(boundaryMatrix)(1)
        if boundaryMatrix(i,3)==1
            then
                mMatrix(i,:) = phiLine(boundaryMatrix(i,:),offsetMatrix)
            else
                if boundaryMatrix(i,3)==2

```



```

        then
            n(1) = boundaryMatrix(i,5)
            n(2) = boundaryMatrix(i,6)
            mMatrix(i,:) =
dphiLine(boundaryMatrix(i,:),offsetMatrix,n(:))
        end
    end
end
endfunction

function dMatrix=domainMainMatrix(domainMatrix, offsetMatrix)
    for i=1:size(domainMatrix)(1)
        dMatrix(i,:) = phiLine(domainMatrix(i,:), offsetMatrix)
    end
endfunction

function fgVector=boundaryConditionsVector(boundaryMatrix)
    for i=1:size(boundaryMatrix)(1)
        fgVector(i,1) = boundaryMatrix(i,4)
    end
endfunction

function numericalResult=fsmResult(offsetValue, domainMatrix,
boundaryMatrix)
    offSetMatrix = getOffSetMatrix(offsetValue, boundaryMatrix)
    fgVector = boundaryConditionsVector(boundaryMatrix)

    mMatrix = mainMatrix(boundaryMatrix, offSetMatrix)
    invMMatrix = inv(mMatrix)
    lambda = invMMatrix * fgVector

    dMatrix = domainMainMatrix(domainMatrix, offSetMatrix)

    numericalResult = dMatrix * lambda
endfunction

///#[5] BLOCO DE EXECUCAO SCILAB

startValue = 100
finalValue = 200000
stepValue = 100
fieldValue = domainMatrix(:,3)

limit = round((finalValue-startValue)/stepValue)

offsetValue = startValue
count = 1

while count ~= limit+1
    disp(count)
    tempResult = fsmResult(offsetValue, domainMatrix, boundaryMatrix)
    offsetValue = offsetValue + stepValue
    chainAnswer(count,:) = tempResult'
    count = count + 1
end

```

```
end

for i=1:size(chainAnswer)(1)
    for j=1:size(fieldValue)(1)
        percentAnswer(i,j) = abs(100*(chainAnswer(i,j)-
fieldValue(j))/fieldValue(j))
    end
end

for m=1:size(percentAnswer')(1)
    plot(percentAnswer(:,m))
end

xlabel("passo")
ylabel("% erro")
```