

EDUCAÇÃO MATEMÁTICA E SUAS TECNOLOGIAS 2

Felipe Antonio Machado Fagundes Gonçalves
(Organizador)

 **Atena**
Editora
Ano 2019

Felipe Antonio Machado Fagundes Gonçalves
(Organizador)

Educação Matemática e suas Tecnologias 2

Atena Editora
2019

2019 by Atena Editora
Copyright © Atena Editora
Copyright do Texto © 2019 Os Autores
Copyright da Edição © 2019 Atena Editora
Editora Executiva: Prof^a Dr^a Antonella Carvalho de Oliveira
Diagramação: Natália Sandrini
Edição de Arte: Lorena Prestes
Revisão: Os Autores

O conteúdo dos artigos e seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva dos autores. Permitido o download da obra e o compartilhamento desde que sejam atribuídos créditos aos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Conselho Editorial

Ciências Humanas e Sociais Aplicadas

Prof. Dr. Álvaro Augusto de Borba Barreto – Universidade Federal de Pelotas
Prof. Dr. Antonio Carlos Frasson – Universidade Tecnológica Federal do Paraná
Prof. Dr. Antonio Isidro-Filho – Universidade de Brasília
Prof. Dr. Constantino Ribeiro de Oliveira Junior – Universidade Estadual de Ponta Grossa
Prof^a Dr^a Cristina Gaio – Universidade de Lisboa
Prof. Dr. Deyvison de Lima Oliveira – Universidade Federal de Rondônia
Prof. Dr. Gilmei Fleck – Universidade Estadual do Oeste do Paraná
Prof^a Dr^a Ivone Goulart Lopes – Istituto Internazionale delle Figlie de Maria Ausiliatrice
Prof^a Dr^a Juliane Sant’Ana Bento – Universidade Federal do Rio Grande do Sul
Prof. Dr. Julio Candido de Meirelles Junior – Universidade Federal Fluminense
Prof^a Dr^a Lina Maria Gonçalves – Universidade Federal do Tocantins
Prof^a Dr^a Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Prof^a Dr^a Paola Andressa Scortegagna – Universidade Estadual de Ponta Grossa
Prof. Dr. Urandi João Rodrigues Junior – Universidade Federal do Oeste do Pará
Prof^a Dr^a Vanessa Bordin Viera – Universidade Federal de Campina Grande
Prof. Dr. Willian Douglas Guilherme – Universidade Federal do Tocantins

Ciências Agrárias e Multidisciplinar

Prof. Dr. Alan Mario Zuffo – Universidade Federal de Mato Grosso do Sul
Prof. Dr. Alexandre Igor Azevedo Pereira – Instituto Federal Goiano
Prof^a Dr^a Daiane Garabeli Trojan – Universidade Norte do Paraná
Prof. Dr. Darllan Collins da Cunha e Silva – Universidade Estadual Paulista
Prof. Dr. Fábio Steiner – Universidade Estadual de Mato Grosso do Sul
Prof^a Dr^a Girlene Santos de Souza – Universidade Federal do Recôncavo da Bahia
Prof. Dr. Jorge González Aguilera – Universidade Federal de Mato Grosso do Sul
Prof. Dr. Ronilson Freitas de Souza – Universidade do Estado do Pará
Prof. Dr. Valdemar Antonio Paffaro Junior – Universidade Federal de Alfenas

Ciências Biológicas e da Saúde

Prof. Dr. Gianfábio Pimentel Franco – Universidade Federal de Santa Maria
Prof. Dr. Benedito Rodrigues da Silva Neto – Universidade Federal de Goiás
Prof.^a Dr.^a Elane Schwinden Prudêncio – Universidade Federal de Santa Catarina
Prof. Dr. José Max Barbosa de Oliveira Junior – Universidade Federal do Oeste do Pará
Prof.^a Dr.^a Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Prof.^a Dr.^a Raissa Rachel Salustriano da Silva Matos – Universidade Federal do Maranhão
Prof.^a Dr.^a Vanessa Lima Gonçalves – Universidade Estadual de Ponta Grossa
Prof.^a Dr.^a Vanessa Bordin Viera – Universidade Federal de Campina Grande

Ciências Exatas e da Terra e Engenharias

Prof. Dr. Adélio Alcino Sampaio Castro Machado – Universidade do Porto
Prof. Dr. Eloi Rufato Junior – Universidade Tecnológica Federal do Paraná
Prof. Dr. Fabrício Menezes Ramos – Instituto Federal do Pará
Prof.^a Dr.^a Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Prof. Dr. Takeshy Tachizawa – Faculdade de Campo Limpo Paulista

Conselho Técnico Científico

Prof. Msc. Abrãao Carvalho Nogueira – Universidade Federal do Espírito Santo
Prof.^a Dr.^a Andreza Lopes – Instituto de Pesquisa e Desenvolvimento Acadêmico
Prof. Msc. Carlos Antônio dos Santos – Universidade Federal Rural do Rio de Janeiro
Prof.^a Msc. Jaqueline Oliveira Rezende – Universidade Federal de Uberlândia
Prof. Msc. Leonardo Tullio – Universidade Estadual de Ponta Grossa
Prof. Dr. Welleson Feitosa Gazel – Universidade Paulista
Prof. Msc. André Flávio Gonçalves Silva – Universidade Federal do Maranhão
Prof.^a Msc. Renata Luciane Polsaque Young Blood – UniSecal
Prof. Msc. Daniel da Silva Miranda – Universidade Federal do Pará

Dados Internacionais de Catalogação na Publicação (CIP) (eDOC BRASIL, Belo Horizonte/MG)	
E24	Educação matemática e suas tecnologias 2 [recurso eletrônico] / Organizador Felipe Antonio Machado Fagundes Gonçalves. – Ponta Grossa (PR): Atena Editora, 2019. – (Educação Matemática e suas Tecnologias; v. 2) Formato: PDF Requisitos de sistema: Adobe Acrobat Reader Modo de acesso: World Wide Web Inclui bibliografia ISBN 978-85-7247-348-4 DOI 10.22533/at.ed.484192405 1. Matemática – Estudo e ensino – Inovações tecnológicas. 2. Tecnologia educacional. I. Gonçalves, Felipe Antonio Machado Fagundes. II. Série. CDD 510.7
Elaborado por Maurício Amormino Júnior – CRB6/2422	

Atena Editora
Ponta Grossa – Paraná - Brasil
www.atenaeditora.com.br
contato@atenaeditora.com.br

APRESENTAÇÃO

A obra “Educação Matemática e suas tecnologias” é composta por quatro volumes, que vêm contribuir de maneira muito significativa para o Ensino da Matemática, nos mais variados níveis de Ensino. Sendo assim uma referência de grande relevância para a área da Educação Matemática. Permeados de tecnologia, os artigos que compõem estes volumes, apontam para o enriquecimento da Matemática como um todo, pois atinge de maneira muito eficaz, estudantes da área e professores que buscam conhecimento e aperfeiçoamento. Pois, no decorrer dos capítulos podemos observar a matemática aplicada a diversas situações, servindo com exemplo de práticas muito bem sucedidas para docentes da área. A relevância da disciplina de Matemática no Ensino Básico e Superior é inquestionável, pois oferece a todo cidadão a capacidade de analisar, interpretar e inferir na sua comunidade, utilizando-se da Matemática como ferramenta para a resolução de problemas do seu cotidiano. Sem dúvidas, professores e pesquisadores da Educação Matemática, encontrarão aqui uma gama de trabalhos concebidos no espaço escolar, vislumbrando possibilidades de ensino e aprendizagem para diversos conteúdos matemáticos. Que estes quatro volumes possam despertar no leitor a busca pelo conhecimento Matemático. E aos professores e pesquisadores da Educação Matemática, desejo que esta obra possa fomentar a busca por ações práticas para o Ensino e Aprendizagem de Matemática.

Felipe Antonio Machado Fagundes Gonçalves

SUMÁRIO

CAPÍTULO 1	1
O ALGORITMO ESPECTRAL COMO ALTERNATIVA AO ALGORITMO K-MEANS EM CONJUNTO DE DADOS ARTIFICIAIS	
Luciano Garim Garcia Leonardo Ramos Emmendorfer	
DOI 10.22533/at.ed.4841924051	
CAPÍTULO 2	16
NOVAS RELAÇÕES NA MATRIZ DE TRANSFORMAÇÃO DA TRANSFORMADA NUMÉRICA DE PASCAL	
Arquimedes José De Araújo Paschoal Ricardo Menezes Campello De Souza Hélio Magalhães De Oliveira	
DOI 10.22533/at.ed.4841924052	
CAPÍTULO 3	24
ALGORITMOS RÁPIDOS PARA O CÁLCULO DA TRANSFORMADA NUMÉRICA DE PASCAL	
Arquimedes José De Araújo Paschoal Ricardo Menezes Campello De Souza	
DOI 10.22533/at.ed.4841924053	
CAPÍTULO 4	32
ANÁLISE DE CÁLCULO DIFERENCIAL USANDO O SOFTWARE GEOGEBRA	
Amanda Barretos Lima Garuth Brenda Anselmo Mendes Isabela Geraldo Reghin Rosângela Teixeira Guedes	
DOI 10.22533/at.ed.4841924054	
CAPÍTULO 5	46
DEFLEXÃO EM VIGAS DE CONCRETO ARMADO SOLUÇÃO ANALÍTICA E NUMÉRICA VIA MÉTODO DAS DIFERENÇAS FINITAS	
Mariana Coelho Portilho Bernardi Adilandri Mércio Lobeiro Jeferson Rafael Bueno Thiago José Sepulveda da Silva	
DOI 10.22533/at.ed.4841924055	
CAPÍTULO 6	57
MODELO MATEMÁTICO PARA AUXILIAR O PLANEJAMENTO DA MANUTENÇÃO PREVENTIVA DE MOTORES ELÉTRICOS	
Thalita Monteiro Obal Jonatas Santana Obal	
DOI 10.22533/at.ed.4841924056	

CAPÍTULO 7	64
PRINCÍPIO DA SUPERPOSIÇÃO E SOLUÇÃO NUMÉRICA DO PROBLEMA DE FLUXO EM AQUÍFERO CONFINADO	
João Paulo Martins dos Santos Alessandro Firmiano de Jesus Edson Wendland	
DOI 10.22533/at.ed.4841924057	
CAPÍTULO 8	83
RESONANT ORBITAL DYNAMICS OF CBERS SATELLITES	
Jarbas Cordeiro Sampaio Rodolpho Vilhena de Moraes Sandro da Silva Fernandes	
DOI 10.22533/at.ed.4841924058	
CAPÍTULO 9	91
TESTES ADAPTATIVOS ENVOLVENDO O CONTEÚDO DE DERIVADAS: UM ESTUDO DE CASO COM ALUNOS DE ENGENHARIA CIVIL	
Patrícia Liane Grudzinski da Silva Claudia Lisete Oliveira Groenwald	
DOI 10.22533/at.ed.4841924059	
CAPÍTULO 10	104
LOCALIZAÇÃO DE FALTAS EM LINHAS DE TRANSMISSÃO POR ANÁLISE DE SINAIS TRANSITÓRIOS DE TENSÃO	
Danilo Pinto Moreira de Souza Eliane da Silva Christo Aryfrance Rocha Almeida	
DOI 10.22533/at.ed.48419240510	
CAPÍTULO 11	116
MODELAGEM DA PROPAGAÇÃO DE FUMAGINA CAUSADA POR MOSCA-BRANCA EM CULTURAS AGRÍCOLA	
Gustavo Henrique Petrolí Norberto Anibal Maidana	
DOI 10.22533/at.ed.48419240511	
CAPÍTULO 12	133
LOS SUBNIVELES DE DESARROLLO DEL ESQUEMA DE DERIVADA: UN ESTUDIO EXPLORATORIO EN EL NIVEL UNIVERSITARIO	
Claudio Fuentealba Edelmira Badillo Gloria Sánchez-Matamoros Andrea Cárcamo	
DOI 10.22533/at.ed.48419240512	
CAPÍTULO 13	143
OTIMIZAÇÃO BASEADA EM CONFIABILIDADE PARA A MINIMIZAÇÃO DE FUNÇÕES MATEMÁTICAS	
Márcio Aurélio da Silva Fran Sérgio Lobato Aldemir Ap Cavalini Jr Valder Steffen Jr	
DOI 10.22533/at.ed.48419240513	

CAPÍTULO 14	156
SEQUÊNCIAS: INTERVALARES E FUZZY	
Gino Gustavo Maqui Huamán	
Ulcilea Alves Severino Leal	
Geraldo Nunes Silva	
DOI 10.22533/at.ed.48419240514	
CAPÍTULO 15	164
VALIDAÇÃO DO MÉTODO DOS ELEMENTOS DISCRETOS PARA O ESCOAMENTO DE GRÃOS DE SOJA	
Rodolfo França de Lima	
Vanessa Faoro	
Manuel Osório Binelo	
Dirceu Lima dos Santos	
Adriano Pilla Zeilmann	
DOI 10.22533/at.ed.48419240515	
CAPÍTULO 16	181
TAREAS DE GENERALIZACIÓN POR INDUCCIÓN PARA FORMAR EL CONCEPTO DE POTENCIA	
Landy Sosa Moguel	
Guadalupe Cabañas-Sánchez	
Eddie Aparicio Landa	
DOI 10.22533/at.ed.48419240516	
CAPÍTULO 17	192
SINCRONISMO EM UM NOVO MODELO METAPOPOPULACIONAL COM TAXA DE MIGRAÇÃO INDEPENDENTE DA DENSIDADE	
Francisco Helmuth Soares Dias	
Jacques Aveline Loureiro da Silva	
DOI 10.22533/at.ed.48419240517	
CAPÍTULO 18	199
SIMULAÇÃO 3D DO FLUXO DE AR DE UM SISTEMA REAL DE ARMAZENAGEM DE GRÃOS	
Vanessa Faoro	
Rodolfo França de Lima	
Aline Tampke Dombrowski	
Manuel Osório Binelo	
DOI 10.22533/at.ed.48419240518	
CAPÍTULO 19	207
CONTROLE ÓTIMO DO FLUXO DE ÁGUA EM UMA FÔRMA DE GELO	
Xie Jiayu	
João Luis Gonçalves	
DOI 10.22533/at.ed.48419240519	
CAPÍTULO 20	213
CÓDIGOS CÍCLICOS DEFINIDOS POR ANULAMENTO	
Conrado Jensen Teixeira	
Osnel Broche Cristo	
DOI 10.22533/at.ed.48419240520	

CAPÍTULO 21	216
ANÁLISE TEÓRICO-EXPERIMENTAL DE DISPERSÃO DE UM CONTAMINANTE COM TRANSFORMAÇÕES INTEGRAIS E INFERÊNCIA BAYESIANA	
Bruno Carlos Lugão Diego Campos Knupp Pedro Paulo Gomes Watts Rodrigues Antônio José da Silva Neto	
DOI 10.22533/at.ed.48419240521	
CAPÍTULO 22	225
ANÁLISE WAVELET DE TACOGRAMAS TEÓRICOS E EXPERIMENTAIS	
Ronaldo Mendes Evaristo Kelly Cristiane Iarosz Silvio Luiz Thomaz de Souza Ricardo Luiz Viana Moacir Fernandes de Godoy Antonio Marcos Batista	
DOI 10.22533/at.ed.48419240522	
CAPÍTULO 23	235
CONSTRUÇÃO DE UM AEROMODELO DE MACARRÃO NO ENSINO DE MATEMÁTICA E FÍSICA	
Alissan Sarturato Firão Ernandes Rocha de Oliveira Zulind Luzmarina Freitas	
DOI 10.22533/at.ed.48419240523	
SOBRE O ORGANIZADOR	239

PRINCÍPIO DA SUPERPOSIÇÃO E SOLUÇÃO NUMÉRICA DO PROBLEMA DE FLUXO EM AQUÍFERO CONFINADO

João Paulo Martins dos Santos

Academia da Força Aérea-AFA
Pirassununga-SP

Alessandro Firmiano de Jesus

Academia da Força Aérea-AFA
Pirassununga-SP

Edson Wendland

Universidade de São Paulo-USP
São Carlos-SP

RESUMO: O Princípio da Superposição de soluções analíticas do problema do fluxo de águas subterrâneas, em aquífero confinado, fornece uma superfície de rebaixamento provocada pela ação de poços de bombeamento no interior do domínio computacional. Por outro lado, no caso transiente, uma solução aproximada pelo método de elementos finitos pode ser obtida da formulação fraca do modelo matemático desse fenômeno hídrico subsuperficial. Neste capítulo, a linguagem de programação Python foi utilizada para implementar e comparar essas resoluções analítica e numérica para três casos de bombeamento conjunto. Na solução de Theis, o rebaixamento foi determinado por meio das contribuições de poços reais e poços imagens, sob processo adaptativo na função de poço. Após refinamentos sucessivos na malha computacional, bibliotecas do FEniCS

permitiram a resolução automatizada e a visualização da concordância obtida numa comparação gráfica de resultados.

PALAVRAS-CHAVE: Princípio da Superposição. Projeto FEniCS. Método de Elementos Finitos. Equação do Fluxo Subterrâneo.

ABSTRACT: In confined aquifer, the Superposition Principle well analytical solutions of groundwater flow problem, provides the lowering surface caused by the action of pumping wells within the computational domain. On the other way, in the transient case, the approximate solution by the finite element method can be obtained from the weak formulation of the mathematical model of this subsurface hydric phenomenon. Here, the Python programming language was used to implement and to compare the analytical and numerical solutions in three cases of joint pumping. In Theis solution, the lowering surface was determined by the contributions of real and images wells under adaptive process in the function. After successive refinements in the computational mesh, the FEniCS libraries have enabled the automated resolution and in the visualization of the agreement obtained in the graphical results.

KEYWORDS: Principle of Superposition. FEniCS Project. Finite Element Method.

1 | INTRODUÇÃO

São várias as contribuições da Matemática e de suas Tecnologias no estudo quantitativo e qualitativo das Ciências Naturais. Em posse de equações governantes, das condições iniciais e de fronteira estabelecidas sobre um domínio computacional, certas linguagens de programação científica, fundamentadas em conhecimentos matemáticos, serão capazes de disponibilizar códigos numéricos para buscar resolução automatizada de determinada lei da Física ou comportamento observável na Natureza. Especificamente, numa certa região de interesse na subsuperfície, esse suporte tecnológico poderia auxiliar na compreensão da ocorrência dos fenômenos naturais relacionados aos recursos hídricos subterrâneos. Desta forma, situações problemas adotadas neste capítulo, ilustram certas aplicações de conceitos matemáticos e de suas tecnologias, abordando um estudo da distribuição de cargas hidráulicas em águas subterrâneas, no intuito de validar resoluções obtidas com o uso de códigos numéricos e de implementar soluções analíticas para um ambiente natural fisicamente inacessível.

Inicialmente, considere que o rebaixamento da superfície potenciométrica de um aquífero confinado possa ser provocado pela ação de um ou de mais poços de bombeamento atuando, em conjunto, no interior do seu domínio. No caso da ação de um poço, sem influência direta das fronteiras, a solução de Theis disponibiliza uma expressão analítica para a superfície de rebaixamento. Caso o poço, em função do seu raio de influência, sofra alguma alteração dada pela fronteira de Dirichlet ou de Neumann, o Princípio da Superposição acrescenta, nessa solução analítica de Theis, a contribuição do rebaixamento do correspondente poço imagem. Se a abordagem for numérica, tal resolução poderia ser obtida através do método de elementos finitos (MEF) aplicado sobre a formulação fraca da equação de fluxo subterrâneo com termo de fonte não nulo e com apropriadas condições de contorno. Neste sentido, esse capítulo implementa, em linguagem de programação científica Python, a solução de Theis sob o Princípio da Superposição para determinar a superfície de rebaixamento de um aquífero confinado sujeito, inicialmente, à ação de um poço. O mesmo processo será aplicado para o bombeamento conjunto de dois poços e, finalmente, generalizado para descrição do rebaixamento conjunto sob a ação de cinco poços no interior do domínio. Para cada um desses três casos, além da solução analítica, o capítulo ainda apresenta a respectiva resolução numérica MEF. Desta forma, o objetivo é apresentar a validação dessa resolução numérica Python, para a equação do fluxo transiente, através de uma concordância de resultados obtidos com a respectiva solução analítica de Theis, proveniente do Princípio da Superposição aplicado na ação conjunta de vários poços de bombeamento.

A resolução numérica Python, além de uma malha de elementos finitos, considera ainda: a definição de um espaço de funções; a adequada apresentação das condições iniciais e de fronteira, e um apropriado método de resolução numérica para os imensos sistemas lineares resultantes. A utilização da plataforma FEniCS (FENICS, 2018), que engloba uma série de outros softwares para a solução automatizada de equações diferenciais parciais, facilita tanto a modelagem computacional, quanto a simulação do rebaixamento permitindo, ainda, a visualização gráfica dos resultados.

2 | EQUAÇÕES GOVERNANTES DO FLUXO DE ÁGUA SUBTERRÂNEA

Um modelo matemático para a distribuição de cargas hidráulicas que regem o fluxo de água subterrânea em aquífero confinado, obtido por meio da Lei de Darcy e pelo princípio da conservação de massa em um volume elementar representativo, segundo Cleary (2007) é dado pela seguinte expressão linear e transiente:

$$\frac{\partial}{\partial x} \left[K_{xx} \frac{\partial h}{\partial x} \right] + \frac{\partial}{\partial y} \left[K_{yy} \frac{\partial h}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_{zz} \frac{\partial h}{\partial z} \right] + F(x, y, z, t) = S_e \frac{\partial h}{\partial t} \quad (1)$$

sendo $h = h(x, y, z, t)$ a carga hidráulica total [L], K_{xx} , K_{yy} e K_{zz} os componentes principais do tensor de condutividade hidráulica [LT^{-1}], S_e o coeficiente de armazenamento específico [L^{-1}] e $F(X, Y, Z, T)$ o termo de fonte ou sorvedouro agindo no interior (X, Y, Z) do aquífero [L^{-1}] num determinado tempo t .

No caso estacionário, o termo de fonte pode ser representado por:

$$F(x, y, z) = \sum_i Q_i \delta(x - x_i) \delta(y - y_i) \delta(z - z_i) \quad (2)$$

sendo Q_i a taxa de bombeamento ou injeção do poço i na coordenada (x_i, y_i, z_i) [$L^3 T^{-1}$] e δ a função delta de Dirac com unidades [L^{-1}].

Se considerar a hipótese de Dupuit, ou seja, que as variações das cargas hidráulicas são desprezíveis ao longo da dimensão vertical z , e que a dimensão horizontal dos aquíferos em escalas regionais pode ser da ordem de dezenas de quilômetros, então a equação do fluxo subterrâneo pode ser modelada por uma equação bidimensional em x e y e representada, segundo Cleary (2007), por:

$$\frac{\partial}{\partial x} \left[T_{xx} \frac{\partial h}{\partial x} \right] + \frac{\partial}{\partial y} \left[T_{yy} \frac{\partial h}{\partial y} \right] + F(x, y, t) = S \frac{\partial h}{\partial t} \quad (3)$$

sendo $T_{xx} = b \cdot K_{xx}$ e $T_{yy} = b \cdot K_{yy}$ [$L^2 T^{-1}$] as transmissividades nas respectivas direções x e y do aquífero confinado de espessura b . O coeficiente de armazenamento $S = S_e \cdot b$ é [adimensional] e a função F representa os termos de drenança dos fluxos verticais remanescentes em $z = 0$ (camada confinante inferior) e em $z = b$ (camada confinante superior) acrescidos da atividade de um poço i na atuando na coordenada (x_i, y_i) . Assim, a equação governante do fluxo transiente de água subterrânea em aquífero confinado, acrescido de suas condições iniciais e de fronteira, compõe o

seguinte Problema do Fluxo:

$$\left\{ \begin{array}{l} \frac{\partial}{\partial x} \left[T_{xx} \frac{\partial h}{\partial x} \right] + \frac{\partial}{\partial y} \left[T_{yy} \frac{\partial h}{\partial y} \right] + F(x, y, t) = S \frac{\partial h}{\partial t} \quad \text{em } \Omega \times (0, t_f] \\ h = h_0 \quad \text{sobre } \Gamma_D \times (0, t_f] \\ \mathbf{n} \cdot D\nabla h = g \quad \text{sobre } \Gamma_N \times (0, t_f] \\ h(\cdot, t_0) = h_0 \quad \text{em } \Omega \quad \text{para } t = 0 \end{array} \right. \quad (4)$$

sendo $\Omega \subset \mathbb{R}^2$ um domínio poligonal limitado e com fronteira Lipschitz Γ que consiste de duas partes disjuntas: a fronteira de Dirichlet Γ_D e a fronteira de Neumann Γ_N , tais que $\Gamma_D \cup \Gamma_N = \partial\Omega$. O tempo final t_f é arbitrário, no entanto Ω , precisa ser especificado.

Segundo Bear (1972), se o domínio possui um poço de extração atuando em seu interior, sem influência da fronteira $\partial\Omega$, então a solução de Theis fornecerá a carga hidráulica h da equação (4) por meio da seguinte expressão analítica:

$$h = h_0 - \frac{Q}{4\pi T} \left[-0,5772 - \ln(u) + \sum_{n=1}^{\infty} (-1)^{n+1} \frac{u^n}{n \cdot n!} \right] = h_0 - \frac{Q}{4\pi T} W(u) \quad (5)$$

sendo a transmissividade $T = K \cdot b$ para espessura b do aquífero e condutividade hidráulica K e a função $u = r^2 \cdot \frac{S_e}{4Tt}$ para r a distância radial até o poço no tempo t . E ainda, $W(u)$ define a função de poço, Q representa a vazão e h_0 é a carga hidráulica inicial. Em contrapartida, a respectiva resolução numérica aproximada pelo método de elementos finitos, requer uma formulação fraca do problema do fluxo da equação, conforme detalhado em Verfürth (2004) e em Santos (2015).

3 | PRINCÍPIO DA SUPERPOSIÇÃO E RESOLUÇÃO NUMÉRICA

O Princípio da Superposição considera a construção de poços imagens, para fornecer rebaixamentos adicionais através da soma de rebaixamentos individuais (BEAR, 1972), (REILLY *et al.*, 1984). Para ilustrar esse princípio no caso de dois poços de bombeamento no domínio computacional unidimensional, obtido por secção vertical do domínio bidimensional retangular, considere Ω de dimensões $M \times M$ e dois poços de bombeamento de mesma vazão localizados nas posições $P_1 \left(\frac{M}{3}, \frac{M}{2} \right)$ e $P_2 \left(\frac{2M}{3}, \frac{M}{2} \right)$ conforme representado na figura 1. Além de receber a imposição da condição de fronteira do tipo Dirichlet $h=h_0$, a fronteira $x = 0$ define, em $P_{imag1} \left(-\frac{M}{3}, \frac{M}{2} \right)$, um poço imagem na posição simétrica à P_1 e a fronteira $x = M$ define, em $P_{imag2} \left(\frac{4M}{3}, \frac{M}{2} \right)$, outro poço imagem na posição simétrica à P_2 . Esses novos poços imagens, em virtude da linearidade da equação, atuarão como sendo poços de injeção com a mesma vazão observada em seus respectivos poços reais. As respectivas cargas hidráulicas de P_1 e P_2 são $h_1 = h_0 - S_1$ e $h_2 = h_0 - S_2$, sendo $S_1 \geq 0$ e $S_2 \geq 0$ os rebaixamentos individuais causados pelo bombeamento dos poços P_1 e P_2 . Para satisfazer as condições nas fronteiras de Dirichlet, os poços imagens de injeção são necessários, pois, esses contribuem com pequenas parcelas para os respectivos rebaixamentos.

Especificamente, o poço imagem contribui com uma pequena parcela para o rebaixamento do poço P1, enquanto que o poço imagem 1 contribui, da mesma forma, com o poço P2. Observe que o poço imagem 1 não contribui no rebaixamento do poço P2 e nem o poço imagem 2 não contribui no rebaixamento do poço P1.

Desta forma, o rebaixamento em um ponto específico do domínio será a soma dos rebaixamentos individuais. Consequentemente, a carga hidráulica h em um ponto específico do domínio será fornecido pela expressão analítica:

$$h = h_0 - (S_1 + S_2) + (S_{1imag} + S_{2imag}) \quad (6)$$

sendo $S_{1imag} \geq 0$ e S_{2imag} respectivas contribuições de cada poço imagem.

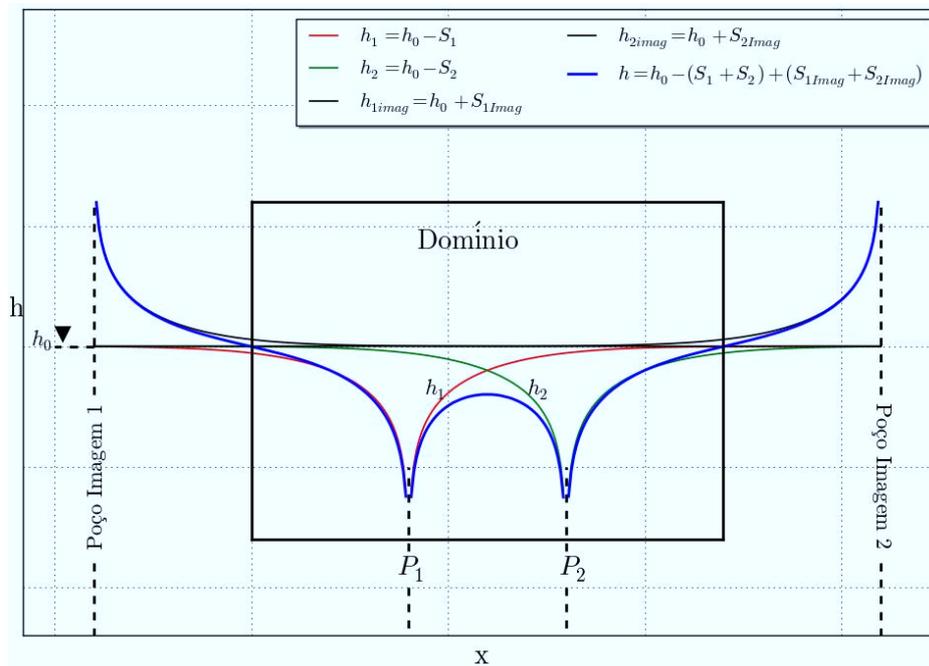


Figura 1 - O Princípio da Superposição para domínio unidimensional com duas fronteiras de Dirichlet. Os poços imagens são de injeção com localização simétrica em relação às fronteiras. A parcela $h_0 - S_j$ representa a carga hidráulica devido o bombeamento do poço j enquanto que $h_0 + S_{jimag}$ representa a carga hidráulica devido à contribuição de injeção do poço imagem j para $j = 1$ e 2 .

Na figura 1 pode ser verificado que $S_1 = S_{1imag}$ no perfil da fronteira $x=0$ e que o valor da carga hidráulica h coincide com a média das funções h_1 e h_{1imag} , ou seja,

$$h_{x=0} = \frac{h_1 + h_{1imag}}{2} = \frac{(h_0 - S_1) + (h_0 + S_{1imag})}{2} = \frac{(h_0 - S_1) + (h_0 + S_1)}{2} = h_0$$

e ainda, $S_2 = S_{2imag}$ no perfil da fronteira $x=M$ com valor de h coincidindo com o valor médio das funções h_2 e h_{2imag} , isto é,

$$h_{x=M} = \frac{h_2 + h_{2imag}}{2} = \frac{(h_0 - S_2) + (h_0 + S_{2imag})}{2} = \frac{(h_0 - S_2) + (h_0 + S_2)}{2} = h_0$$

Assim, a solução analítica obtida pelo Princípio da Superposição satisfaz a condição de Dirichlet do problema de fluxo (4) na qual impõe $h = h_0$ sobre $\Gamma_D \times (0,tf]$.

Esse princípio representado na equação (6) pode ser estendido para a obtenção da superfície de rebaixamento em um domínio Ω com vários poços de bombeamento, ou de injeção, mesmo sob atuação com taxa de vazões distintas.

Além disso, condições de fronteira de Neumann podem ser adotadas em conjunto com as condições de Dirichlet. Nas simulações deste capítulo, apenas as condições de Dirichlet serão adotadas, pois o foco inicial está na apresentação didática e tecnológica para implementação computacional da solução analítica (5) sujeita à generalização pela equação (6). No entanto, no interesse de considerar a condição de Neumann, basta definir cada poço imagem simétrico à fronteira em questão como sendo poço de bombeamento sob a mesma vazão do seu respectivo poço real.

Se a abordagem computacional inicia na definição da lista de pares ordenados, então $L = [(X_1, Y_1); (X_2, Y_2); \dots; (X_k, Y_k)]$, dará a posição de cada poço no interior e as sublistas $listaCoorX = [X_1; X_2; \dots; X_k]$ e $listaCoorY = [Y_1; Y_2; \dots; Y_k]$ serão para as posições cartesianas na direção dos respectivos eixos x e y. Adicionalmente, a taxa de vazão Q_j e o raio r_j do poço na posição (X_j, Y_j) seguirão armazenados pelas listas $L_{Q_w} = [Q_1; Q_2 \dots; Q_k]$ e $L_{r_w} = [r_1; r_2 \dots; r_k]$. A obtenção da resolução numérica da equação (4), sujeita a uma lista L de fontes pontuais no interior do domínio Ω e taxa de vazões L_{Q_w} , se apoiará no método de elementos finitos e utilizará as ferramentas computacionais e conceitos de adaptatividade disponíveis na programação científica do projeto FEniCS (LOGG *et al.* 2012). A validação dessa resolução numérica, obtida por programada em linguagem Python, será baseada na comparação gráfica de resultados obtidos com a respectiva solução analítica de Theis que generaliza a expressão sobre o conjunto de k poços.

Além do funcionamento conjunto dos poços atuando sob vazões distintas, dadas pela lista L_{Q_w} , será ainda estabelecido que:

i.) o rebaixamento resultante, que não pode ser superior a h_0 , dependa apenas da distância entre o ponto de observação (X, Y) e das coordenadas $(CoorX, CoorY)$ do poço, ou seja, dependa da distância $r = \sqrt{(X - CoorX)^2 + (Y - CoorY)^2}$;

ii.) o índice n na função de poço $W(u)$ da solução analítica de Theis (5) não precisa ser especificado, pois o cálculo será realizado iterativamente por procedimento adaptativo implementado para cada ponto do domínio computacional;

iii.) supondo Ω um domínio retangular definido pelas as coordenadas da diagonal $(x_0, y_0) = (0, 0)$ e (x_1, y_1) , então as posições dos possíveis 8 poços imagens de um poço real situado em (P_x, P_y) serão: $(-P_x, P_y)$, $(P_x, -P_y)$, $(-P_x, -P_y)$, $(2x_1 - P_x, P_y)$, $(2x_1 - P_x, 2y_1 - P_y)$, $(P_x, 2y_1 - P_y)$, $(2x_1 - P_x, -P_y)$ e $(-P_x, 2y_1 - P_y)$. Dessa forma, dado , todas as listas de coordenadas dos poços imagens podem ser obtidas;

iv.) para a lista L de k poços, o rebaixamento ocasionado em um ponto específico será dado pela soma de todas as contribuições dos poços reais e imagens, sendo que, em função do raio de influência, alguns poços imagens construídos em **iii.)** podem ser descartados por não apresentarem efeitos de rebaixamento em certas regiões do domínio computacional.

3.1 Implementação das soluções Python-FEniCS

Baseado nos itens *i.)* a *iv.)* estabelecidos na seção anterior, a implementação da solução analítica para a contribuição de todos os poços da lista L, será fornecido por:

$$WellFunctionAdaptiveTheis(mesh, listaCoorX, listaCoorY, listaR_w, listaQ_w, V) \quad (7)$$

um código Python-FEniCS, programado para implementar a generalização da equação (6) disponibilizando a solução analítica do problema de fluxo subterrâneo (4). Esse código é alimentado por malha refinada no entorno dos poços reais e por informações sobre coordenadas de posicionamento, raios de abertura, taxas de bombeamento e um espaço V de funções interpoladoras. E devolve a solução analítica obtida pelo Princípio da Superposição. As principais linhas de programação do código Python (7) são:

```
1 def WellFunctionAdaptiveTheis(mesh, listaCoorX, listaCoorY, listaRw, listaQw, V):
2     class Well(Expression):
3         def eval( self, value, x):
4             tol=1.0E-15
5             TheisPartialSum=0.0
6             TolForTheis=1.0E-07
7             valor1=0.0
8             for i in range(rangeToRun):
9                 Rad=np.sqrt((x[0]- listaCoorX[ i ])**2+(x[1]- listaCoorY[ i ])**2);
10                u=(Rad**2*S*e)/(4.0*T*t)
11                if (Rad<=listaRw[i]):
12                    TheisPartialSum`iWell0=-0.5772-np.log(u)+u-u**2/(2.0*2.0)+u**3/(3.0*6.0)-u**4/(4.0*24.0)
13                    TheisPartialSum`iWell1=-0.5772-np.log(u)+u-u**2/(2.0*2.0)+u**3/(3.0*6.0)-u**4/(4.0*24.0)
14                    l=5;
15                    AbsValue=1000.0
16                    while( Abs Value<TolForTheis):
17                        TheisPartialSum`iWell1 += (-1)**(l+1)*u**l/(1*math. factorial( l))
18                        Abs Value=np.abs(TheisPartialSum`iWell1 - TheisPartialSum`iWell0)
19                        TheisPartialSum`iWell0=TheisPartialSum`iWell1
20                        if TheisPartialSum`iWell1 <= lconstant*listaQw[ i ]/(4 *np.pi*T):
21                            TheisPartialSum`iWell1=0
22                            TheisPartialSum`iWell0=0
23                            Abs Value=0
24                            l+=1
25                            valor1 +=(listaQw[ i ]/(4.0 *np.pi*T))*TheisPartialSum`iWell1
26                value[0]= valor1
27                if abs( value [0]) > 1.0E-10:
28                    value [0]=0.0
29                elif ( value [0]>0.0) :
30                    value [0]=0.0
31                value [0]=value [0]
32            Imagx= Well(element=V.ufl`element());
33            Imagx = interpolate (Imagx,V)
34            return Imagx
```

E ainda, os poços imagens indicados em *iii.)* foram obtidos por estratégias de reflexão e translação das coordenadas da lista , implementados pelos códigos Python:

```
1 def MakeImag(listaCoor):
2     lista =[]
3     for i in range(rangeToRun):
4         lista .append(- listaCoor [ i ])
5     return lista
```

```
1 def MakeImagMove(listaCoor,xi):
2     lista =[]
3     for i in range(rangeToRun):
4         distanceFromXi=xi- listaCoor [ i ]
5         valor=xi+distanceFromXi;
6         lista .append(valor)
7     return lista
```

Para evitar contribuições não significativas, devido o raio de influência de certos poços da lista L, foi adotado um critério de parada para o cálculo das somas parciais em (5). Considerando que a função de poço não depende do tipo de bombeamento, então $W(u) > h_0$ foi usado como critério de detecção daqueles poços que não fornecem contribuições significativas ao rebaixamento.

Em relação à resolução obtida por métodos numéricos, a habilidade de uma malha computacional representar a discretização do domínio Ω influenciará a qualidade de seus resultados. Na obtenção da resolução Python MEF do problema de fluxo (4), o processo de geração de malhas não estruturadas também será facilitado pela utilização direta de classes disponíveis nas bibliotecas do projeto FEniCS (FENICS, 2018). Nessa malha MEF não estruturada, a definição na posição dos poços requer que os mesmos sejam vértices da malha de elementos finitos (ISTOK, 1989). Assim, uma mudança de coordenadas para que vértices da malha assumam a posição do poço mais próximo será realizada pela função Python, *MeshPassingToWellAsVertex(**)*. Essa função recebe a malha inicial, as listas *listaCoorX* e *listaCoorY* e retorna a malha final em que as coordenadas da lista L são vértices dos elementos finitos. O *script* abaixo descreve a implementação do código mencionado:

```

1 def MeshPassingToWellAsVertex(mesh,listaCoorX,listaCoorY):
    coords=mesh.coordinates(); tree = mesh.bounding_box_tree()
3     point_cloud = [ dolfin.Point(point) for point in mesh.coordinates() ]
    tree.build(point_cloud, 2); p=Point(coorx[0],coory[0])
5     p_i, distance = tree.compute_closest_point(p); lengthCoor=np.shape(coords)[0]
    for i in range(len(coorx)):
7         p=Point(coorx[i],coory[i])
        p_i, distance = tree.compute_closest_point(p); print "p:", p.str()
9         coords[p_i][0]=coorx[i]; coords[p_i][1]=coory[i];
    return mesh

```

Uma estratégia de refinamento local, conforme descrição do código a seguir, foi adotada para as regiões radiais em torno dos poços para elevar o nível de acurácia da solução numérica na proximidade das coordenadas que representam os poços reais.

```

def FunctionToRefine(mesh,listaCoorX, listaCoorY, lista ):
2     for i in range(len( lista )):
        cell_markers = MeshFunction("bool",mesh, mesh.topology().dim())
4         cell_markers.set_all(False)
        for c in cells(mesh):
6             for w in range(len(listaCoorX)):
                p=Point(listaCoorX[w],listaCoorY[w])
8                 if (c.midpoint().distance(p) < lista[i]):
                    cell_markers[c] = True
10            mesh = refine(mesh, cell_markers);
    return mesh

```

A manipulação da resolução numérica utilizou a biblioteca NumPy (WALT *et al.*, 2011) e resultados gráficos foram gerados por meio da biblioteca Matplotlib (HUNTER, 2007). A integração entre componentes do Projeto FEniCS (FENICS, 2018) a a programação científica dos demais códigos foi realizada computacionalmente em ambiente de desenvolvimento Python da Plataforma Eclipse (ECLIPSE, 2015). A arquitetura do ParaView (AHRENS *et al.*, 2005) também contribuiu para a visualização científica dos níveis de detalhes na sobreposição de resultados analíticos e numéricos.

É importante ressaltar que a totalidade da metodologia computacional descrita nesta seção empregou a filosofia colaborativa para a utilização de códigos abertos.

4 | COMPILAÇÃO DE RESULTADOS

Nesta seção, quatro simulações numéricas serão consideradas para exemplificar e ilustrar a metodologia computacional e os códigos Python previamente apresentados. O primeiro exemplo considera uma discussão sobre a adaptatividade que otimiza os resultados numéricos para aproximar o termo de série da equação (5) e ilustra o cálculo iterativo da função de poço $W(u)$ com quantidade distintas de termos em cada ponto do domínio computacional Ω . O segundo problema trata da solução numérica MEF e analítica de Theis com ação de um único poço no domínio. O terceiro problema ilustra a aplicação do Princípio da Superposição com a comparação de resultados analítico e numérico. E, finalmente, o quarto exemplo discute os resultados para um conjunto de cinco poços reais no domínio computacional através de comparação entre as respectivas soluções numérica MEF e analítica de Theis.

Os exemplos envolvendo a resolução do problema do fluxo de água subterrânea (4), sob a ação de poços de bombeamento serão obtidos, analiticamente, pela solução de Theis generalizada na equação (6) e submetidas, em conjunto aos poços imagens, no código Python (7) do Princípio da Superposição. A respectiva resolução numérica, fornecida pelo método de elementos finitos que utilizam bibliotecas do projeto FEniCS (FENICS, 2018), considera domínios computacionais retangulares sobre malhas triangulares não estruturadas. A geração dessas malhas MEF, de extensão .xml, são obtidas pelo código FEniCS `generatemesh(rect,l,"cgal")` sob o domínio retangular Ω predefinido na linha de comando Python `rect = Rectangle(P0,P1)`. A figura 2 ilustra o domínio computacional onde cada vértice da malha MEF, que está mais próximo de um poço real, foi redefinido para assumir a respectiva coordenada antes de aplicar as estratégias de refinamento, através dos códigos `MeshPassingToWellAsVertex(**)` e `FunctionToRefine(**)`.

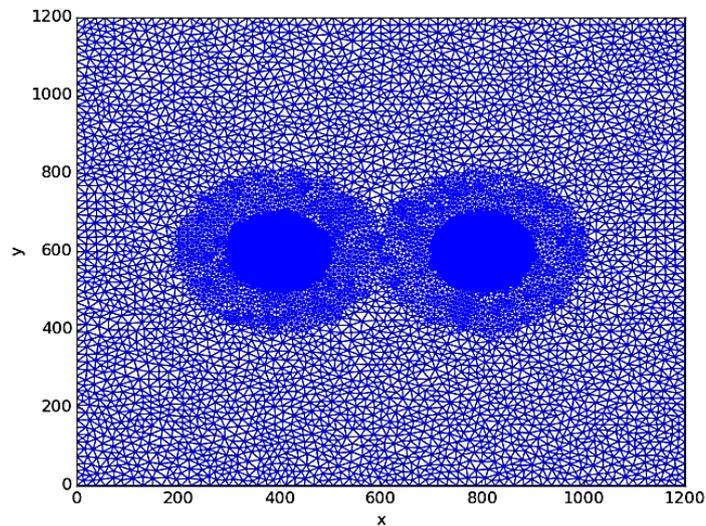


Figura 2 – Domínio computacional MEF e estratégias de refinamento adaptativo

4.1 Adaptatividade para o cálculo da função de poço $W(u)$

Conforme apresentado na equação (5), a solução analítica de Theis envolve o cômputo da função de poço $W(u)$ que depende de uma soma com infinitos termos. Uma forma de estabelecer um critério de parada para o termo de truncamento da série é realizar a soma parcial e obter uma aproximação S_n definida por:

$$W(u) \approx S_n = -0,5772 - \ln(u) + u - \frac{u^2}{2 \cdot 2!} + \frac{u^3}{3 \cdot 3!} - \frac{u^4}{4 \cdot 4!} + \dots + (-1)^{n+1} \frac{u^n}{n \cdot n!} \quad (8)$$

Esta abordagem não é muito apropriada, pois conduz a uma escolha empírica do índice n , no qual pode não refletir na quantidade de termos necessários em um ponto específico do domínio computacional. Ou seja, algumas regiões do domínio retangular Ω podem requerer mais termos da série, enquanto que, em outras regiões, um número menor de termos pode ser aceitável. Uma alternativa para contornar este problema foi utilizar somas parciais S_n e S_{n+1} e um critério de parada tal que $|S_{n+1} - S_n| < \epsilon$ para um valor tolerável do erro de truncamento $\epsilon > 0$. Neste caso, a soma aproximada $W(u) \approx S_{n+1}$ satisfaz o mesmo critério de parada para todo ponto do domínio. Com esta estratégia de adaptatividade nas simulações numéricas, ao impor $\epsilon = 10^{-8}$, o valor máximo requerido foi de $n = 22$ termos e o valor mínimo foi de apenas $n = 3$ termos, evidenciando uma apropriada otimização computacional que foi obtida com esse critério adaptativo adotado.

No intuito de representar a influência da estratégia adaptativa no critério de parada, visando a qualidade da solução analítica, as figuras 3(a) e 3(b) ilustram mapas de diferenças entre expressões calculadas de forma distinta para a aproximação de (7). Para isto, considere que W_1 , W_2 e W_3 representem as aproximações obtidas com os respectivos critérios: $\epsilon = 10^{-8}$, $n=10$ e $\epsilon = 10^{-3}$.

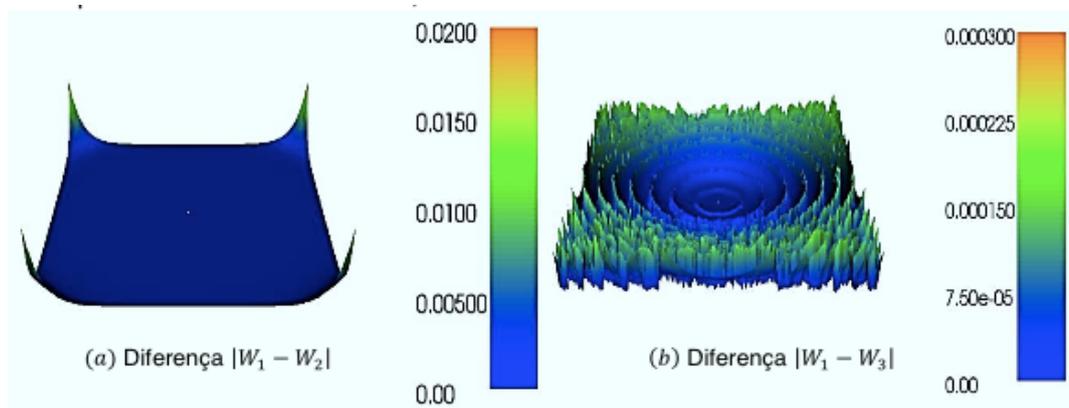


Figura 3 - Diferenças entre as aproximações W_1 , W_2 , e W_3 , sendo que W_1 foi obtida com critério de parada $\epsilon = 10^{-8}$, W_2 uma expressão obtida com $n=10$ termos em (7) e W_3 obtido com $\epsilon = 10^{-3}$

Na figura 3(a), ilustrando o gráfico da diferença $|W_1 - W_2|$, verifica-se que nessa estratégia existe uma discordância da ordem 10^{-2} em regiões nas proximidades dos cantos do domínio Ω . Na figura 3(b), que ilustra o gráfico da diferença $|W_1 - W_3|$, a discordância entre as estratégias é da ordem 10^{-4} em todo domínio Ω . Ou seja, o grau de grandeza observado nos valores graduados em cada escala dessas figuras revela a equivalência entre W_1, W_2 e W_3 . Assim, ao comparar W_1 com W_2 , observa-se regiões onde $n=10$ termos não foram suficientes para fornecer uma aproximação adequada de $W(u)$. Desta forma, essa análise nos resultados gráfico evidencia que a aproximação W_3 apresenta maior qualidade numérica do que W_2 , pois, observando as diferenças, essa estratégia é a que mais se aproxima de W_2 . Logo, o erro de aproximação envolvido ao usar W_3 será menor do que o da estratégia W_2 que necessita $n=10$ termos para representar $W(u)$ em todos os pontos do domínio Ω .

Portanto, pode ser sugerido que a estratégia adaptativa que adota o critério de parada $|S_{n+1} - S_n| < \epsilon$ disponibiliza aproximações de melhores qualidade do que a estratégia que aplica um valor constante de termos para todo o domínio Ω .

4.2 Comparação entre Solução de Theis e Solução Numérica para 1 poço

Considere um poço de bombeamento operando, com taxa de vazão constante Q , as águas subterrâneas no interior de um aquífero confinado de domínio com dimensões retangulares $\Omega = 1.200\text{m} \times 1.200\text{m}$. Para determinar a solução numérica da distribuição de cargas hidráulicas do problema de fluxo, a programação Python empregou o método de elementos finitos na formulação fraca do problema (4), conforme descrição nas linhas de comando do código abaixo:

```

1 # VARIATIONAL FORMULATION OF FINITE ELEMENTS
2 u = TrialFunction(V)
3 v = TestFunction(V)
4 f2=Constant(0.0)
5 f2=WellsLocFunction
6 a = S'e*b*u*v*dx + theta*dt*K*b*inner(nabla'grad(u), nabla'grad(v))*dx
7 L = (S'e*b*u'1*v + dt*f2*v -(1.0- theta)*dt*T*inner(nabla'grad(u'1), nabla'grad(v)))*dx
8
9 A = assemble(a)
10 b = None # variable used for memory savings in assemble calls
11
12 u = Function(V)
13
14 # LOOPING START
15 t = dt
16 while t <= t'stop :
17     b = assemble(L, tensor=b)
18     for bc in bcs:
19         bc.apply(A,b)
20     solve(A, u.vector(), b)
21     if t == dt:
22         plot(u'1, interactive =False, title =' Solution by FEM of Flux Equation')
23     t += dt
24     u'1.assign(u)
25     plot(u'1)

```

Para o cálculo dessa solução numérica MEF, foi gerado uma malha inicial não estruturada contendo $N_v = 3.281$ vértices distribuídos num total de $N_e = 6.304$ elementos triangulares. Na sequência, visando uma melhor qualidade da solução MEF, foram realizados refinamentos sucessivos nas regiões próximas ao centro do poço cujas distâncias radiais são inferiores à [600m, 100m, 50m,10m]. O resultado final foi uma malha mais fina contendo $N_v = 12.743$ vértices sobre $N_e = 25,187$ elementos.

Na figura 4, sobreposta à malha do domínio computacional Ω , está representada a solução numérica MEF da equação do fluxo subterrâneo (4) sob a atuação de 1 poço de bombeamento em seu interior.

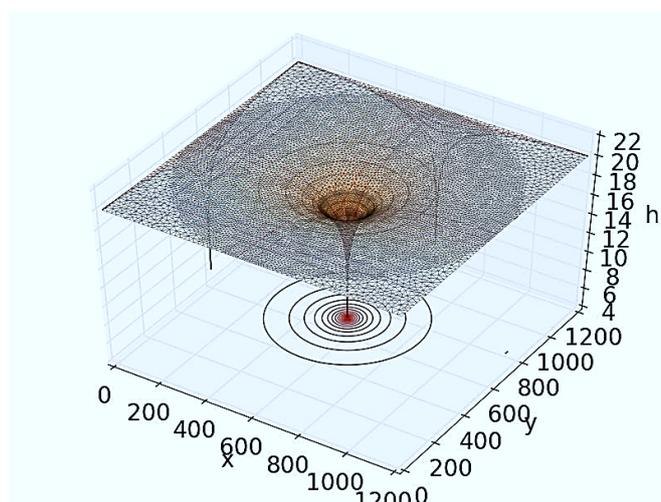
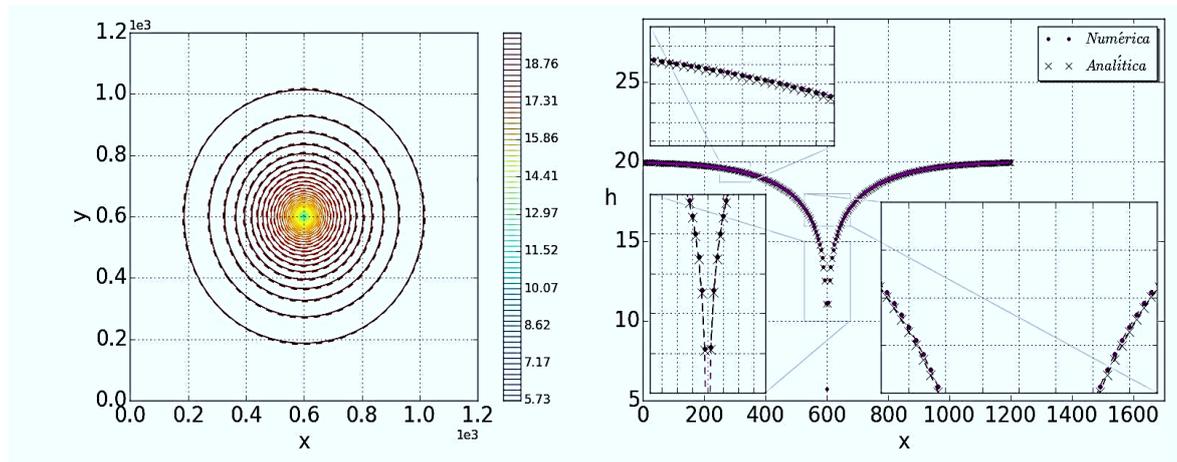


Figura 4 – Domínio computacional $\Omega = 1.200\text{m} \times 1.200\text{m}$ e a solução numérica MEF do rebaixamento $h(x, y, t)$ seguido de projeções de perfis sobre os planos coordenados

Em relação à solução analítica de Theis para o problema de fluxo (4), não houve a necessidade da aplicação do Princípio da Superposição, pois, o poço ficou situado no centro do domínio Ω e, em função da vazão constante adotada, o raio de influência

permanece menor do que a distância do poço à fronteira $\partial\Omega$. Assim, para nesta primeira simulação, a solução de Theis foi obtida com a aplicação direta da equação por meio da função Python implementada no código (7) *WellFunctionAdaptiveTheis(**)*.

Uma comparação dos resultados observados nas avaliações numérica e analítica do problema de fluxo (4) são apresentados pelas curvas de nível e pela projeção de perfis das figuras 5(a) e 5(b).



Curvas de nível das soluções MEF e Theis Perfis do rebaixamento

Figura 5 – Sobreposição das curvas de nível das soluções numérica e analítica e detalhes da concordância dos perfis das soluções MEF e de Theis para o rebaixamento

Se considerar desprezível o esforço computacional no cálculo da solução de Theis através da estratégia $\epsilon = 10^{-8}$ para o critério de parada, observa-se, pela sobreposição gráfica das soluções MEF e analítica do problema de fluxo (4), uma adequada concordância entre os resultados simulados sob o domínio Ω .

A finalidade da comparação entre a solução numérica e a solução analítica da equação (4), para situações simplificadas, é a validação do código numérico MEF implementado em linguagem Python apoiada em bibliotecas computacionais do projeto FEniCS. Assim, a solução MEF obtida por esse código numérico será capaz de disponibilizar, mesmo em situações inevitáveis em que a solução analítica não se encontra disponível ou é inexistente, resultados robustos e apropriados para o problema de fluxo subterrâneo em aquífero confinado, sob imposição de outras condições iniciais e de fronteira ou ação transiente de um poço de bombeamento.

4.3 Princípio da Superposição e a Solução Numérica para 2 poços

Nesta seção de aplicação do Princípio da Superposição e obtenção da solução numérica MEF, serão considerados dois poços atuando no mesmo domínio computacional do exemplo anterior, ou seja, a superfície de rebaixamento resultará da atuação conjunta desses dois poços de bombeamento, conforme ilustra a figura 6.

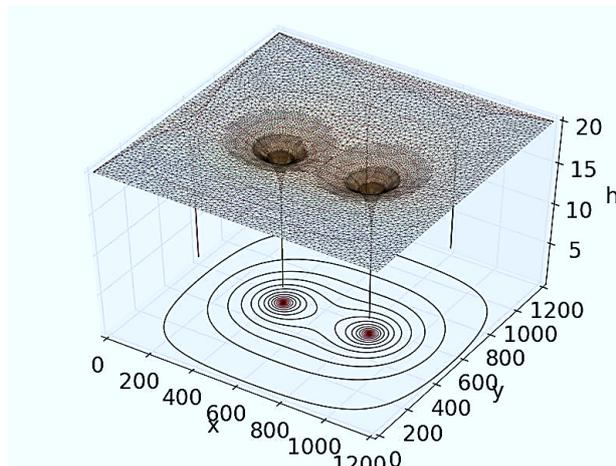


Figura 6 – Solução Python MEF do rebaixamento $h(x, y, t)$ e perfis da sobreposição com a respectiva solução de Theis obtida com a aplicação do Princípio da Superposição para 2 poços de bombeamento

Nesta situação, além da interação entre os poços reais P1 e P2, haverá a necessidade da construção dos poços imagens de modo que as condições de fronteira de Dirichlet sejam satisfeitas. É importante notar que nem todos os poços imagens fornecerão contribuições significativas ao rebaixamento (x, y, t) . Por exemplo, em função das disposições adotadas, o poço imagem de P1, em relação a fronteira de Dirichlet $x=0$, contribuirá para o rebaixamento de P1, mas, em função da distância e vazão atribuída, não contribuirá para o rebaixamento de P2. Analogamente, o poço imagem de P2, em relação a fronteira $x=1.200$, contribuirá para o rebaixamento de P2, mas não terá influência no rebaixamento de P1.

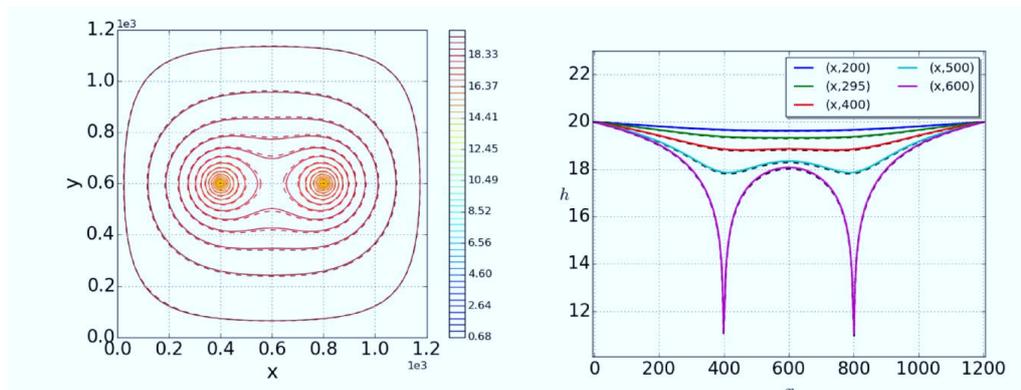
Baseado na lista L de coordenadas dos poços, todas as listas de coordenadas dos poços imagens podem ser obtidas por meio das funções *MarkelMag*(□) e *MarkelMagMove*(□) apresentadas no final da seção Implementação Python-FEniCS. Assim, enquanto a solução analítica, que aplica o Princípio da Superposição, foi obtida por meio da aplicação sucessiva da função *WellFunctionAdaptiveTheis*(**) nas listas de poços reais e imagens, a correspondente resolução numérica da equação do fluxo (4), com 2 poços de bombeamento, foi obtida com a implementação do método de elementos finitos baseada nas bibliotecas do projeto FEniCS (FENICS, 2018).

Nessa obtenção da solução numérica Python MEF, considerou-se uma malha inicial não estruturada com $N_v = 4.624$ vértices distribuídos em $N_e = 8.990$ elementos triangulares. Em seguida, após realizado estratégias de refinamentos sucessivos em região de proximidades radiais a cada poço com distância inferior à

$$[200m, 100m, 50m, 10m, 1m, 0.50m, 0.25m, 0.20m, 0.19m, 0.18m, 0.17m]$$

obteve-se uma malha final com $N_v = 26.690$ vértices sobre $N_e = 53.122$ elementos.

A adequada concordância entre a solução analítica de Theis, com $\epsilon = 10^{-8}$, para aproximação da função de poço $W(u)$ e a solução numérica Python MEF pode ser observada pela sobreposição gráfica das correspondentes curvas de nível e perfis de suas projeções em planos coordenados, conforme visto na figura 7.



(a) Curvas de nível das soluções MEF e analítica (b) Perfis das soluções em malha regular

Figura 7 - Solução numérica Python MEF em adequada concordância com a solução analítica obtida pelo Princípio da Superposição para 2 poços de bombeamento em domínio retangular

As figuras 7(a) e 7(b) ilustram, respectivamente, as curvas de nível e os perfis de soluções com sobreposições adequadas dos resultados numéricos Python MEF sobre os resultados analíticos do Princípio da Superposição.

A reutilização da função Python FEniCS (7) *WellFunctionAdaptiveTheis(**)* para todas as listas de poços imagens foi responsável pelo fornecimento de todas as contribuições necessárias para a definição adequada do rebaixamento final.

Outra vantagem observada nessa metodologia do Princípio da Superposição, sob a ação de 2 poços reais para o agrupamento das contribuições no rebaixamento $h(x, y, t)$ dado pelos poços imagens, foi a independência do número de poços reais no domínio computacional. Assim, qualquer quantidade de poços imagens podem ser construídos por meio das funções de reflexão e translação apresentadas anteriormente.

De forma semelhante ao que foi discutido na seção anterior, a comparação e verificação de equivalência entre as soluções numérica e analítica para a equação, revela, novamente, uma apropriada robustez e confiabilidade do código Python FEniCS para aplicação na resolução do problema de fluxo de água subterrânea sob a ação de dois poços de bombeamento em situações em que a solução analítica não está disponível. Ou seja, para outros problemas de fluxo (4) que podem estar sujeitos às condições de fronteiras aplicadas sobre domínios poligonais de formas irregulares e complexas ou sob condições iniciais de taxas de vazões mudando com o tempo.

4.4 Princípio da Superposição e a Solução Numérica para 5 poços

Para finalizar a comparação entre os resultados da solução Python FEniCS com os respectivos resultados da solução analítica obtida pelo Princípio da Superposição no problema do fluxo em aquífero confinado, serão considerados cinco poços de bombeamento atuando, conjuntamente, no mesmo domínio computacional Ω da seção anterior. A lista L5 de coordenadas para a posição dos poços reais será dada por:

$$L_5 = [(300,300); (300,900); (600,600); (900,300); (900,900)]$$

Nesta situação, a estratégia de refinamentos locais sucessivos será conduzida por lista contendo valores para as distâncias radiais $R = [200\text{m}, 100\text{m}, 50\text{m}, 10\text{m}, 1\text{m}]$. A figura 8 ilustra a malha final refinada em regiões de proximidades dos 5 vértices que representam a posição dos poços. Conforme apresentado anteriormente, a obtenção da solução analítica para este caso também considerou a construção de todas as possíveis listas com informações dos poços imagens e a aplicação sucessiva do código Python *WellFunctionAdaptiveTheis(**)* para o cálculo das contribuições de rebaixamento para cada um deles numa lista de poços.

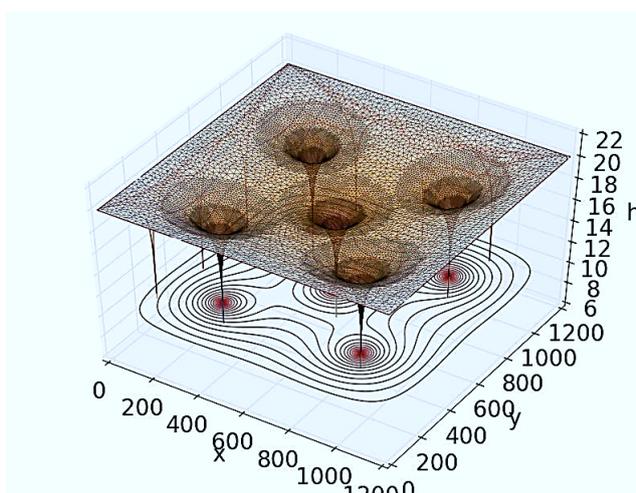
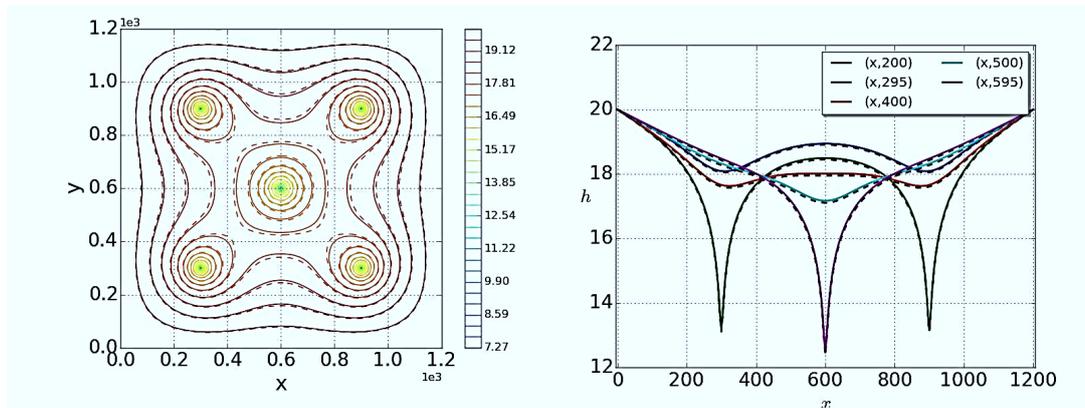


Figura 8 - Solução Python MEF sobreposta à solução analítica obtida pela generalização do Princípio da Superposição em domínio computacional sob a ação conjunta de 5 poços de bombeamento

Este caso pode ser considerado como uma generalização do Princípio da Superposição, implementada pela equação (6), para obtenção da superfície do rebaixamento provocado na interação de 5 poços reais do interior do domínio Ω e os seus respectivos poços imagens. O valor $\epsilon = 10^{-8}$ foi novamente usado na estratégia adaptativa para o cálculo de cada função de poço $W(u)$. Este exemplo difere do caso para dois poços, em termos das listas de poços imagens, apenas pela quantidade de elementos, pois a quantidade de listas é a mesma. No entanto, se considerar as contribuições dos poços imagens, em todas as listas geradas sempre haverá pelo menos um poço imagem com contribuição significativa para o rebaixamento, e poços que não fornecem contribuições significativas. Neste caso, o critério de parada para estabelecer se a contribuição é significativa será ativado. Este critério, em conjunto com a adaptatividade da solução analítica, fornece uma maneira eficiente para o cálculo das contribuições e facilita a obtenção da solução analítica via o Princípio da Superposição. A partir de resultados gráficos, a figura 9(a) ilustra uma adequada concordância observada na sobreposição das curvas de nível da solução analítica e a correspondente solução numérica Python FEniCS, e a figura 9(b) ilustra a projeção de cinco perfis de solução analítica determinada pelo Princípio da Superposição satisfatoriamente sobrepostos às respectivas soluções numéricas Python FEniCS.



(a) Sobreposição das curvas de nível

(b) Perfis das soluções MEF e analítica

Figura 9 - Curvas de nível das soluções analítica e numérica e a comparação entre os perfis de solução obtido pelo Princípio da Superposição com a respectiva solução Python FEniCS

Esses resultados numéricos e analíticos revelam a equivalência nos dois métodos de resolução do problema do fluxo sob a atuação de 5 poços de bombeamento em aquífero confinado. A escolha dessa quantidade e da disposição dos poços reais foi empírica, no entanto, de caráter didático. Em relação à implementação da solução numérica Python FEniCS para representação do comportamento hidráulico, o código se demonstrou isento de maiores dificuldades para aplicação em situações de aquíferos confinados que possuam geometria complexa da sua fronteira irregular ou estão submetidos a ação simultânea de diversas fontes de extração em seu interior.

Sob condições simplificadas do domínio, a eficiência computacional e robustez para obtenção da solução analítica, através do proposto código Python 7, também se apresenta de forma independente da quantidade e disposição dos poços reais.

5 | CONSIDERAÇÕES FINAIS

Especificamente, neste capítulo, um código Python-FEniCS foi desenvolvido para a validação da implementação computacional de uma solução aproximada para o problema do fluxo de água subterrânea, que simula a superfície de rebaixamento sob a ação de poços de bombeamento, em aquífero confinado. A solução analítica de Theis e o Princípio da Superposição foram implementadas para o mesmo conjunto de poços de bombeamento sob condições de fronteira de Dirichlet. Resultados disponíveis em visualizadores científicos apresentaram adequada concordância, entre as soluções numéricas e as respectivas soluções analíticas, ao considerar 1, 2 e 5 poços de bombeamento no interior do domínio computacional. Essas simulações numéricas mostraram tanto uma adequação e robustez da solução aproximada pelo método de elementos finitos, quanto a capacidade da correspondente solução analítica, baseada no Princípio da Superposição, para englobar uma lista de k poços de bombeamento ou de injeção atuando, em conjunto, na mesma região de um determinado aquífero confinado. Portanto, importantes conceitos matemáticos e certos aplicativos de códigos abertos

foram utilizados para a modelagem, simulação, descrição e previsão da distribuição de cargas hidráulicas no meio subterrâneo. O emprego das citadas tecnologias, tais como, plataformas de programação Eclipse e FEniCS, linguagem Python científico de alto nível, bibliotecas MEF para códigos numéricos, geradores de malhas Dolfin, visualizadores gráficos Matplotlib e ParaView, de forma adequadamente integrada, serviram como ferramentas essenciais para a compreensão de um fenômeno físico de complexidade considerável. Na intenção de proporcionar maior leveza ao texto desse capítulo, as demonstrações matemáticas das equações (1) a (6), a descrição da formulação fraca do Problema de Fluxo, os detalhes da expressiva programação Python FEniCS e dos visualizadores científicos, ficaram apenas como sugestões para encorajar um aprofundamento nas obras científicas citadas na Referência.

Evidenciando a característica multidisciplinar que pode ser extraída de várias áreas do conhecimento, são muitos os conceitos da Matemática e suas Tecnologias que contribuem para resolução de problemas provenientes das Ciências Naturais. E ainda, sobreposto à uma mera apresentação de resultados numéricos em comparação aos resultados analíticos, a concordância obtida em diferentes enfoques ressalta, além da validação de implementação, a integração de ferramentas digitais e a confiabilidade computacional para que metodologias de simulação sigam seus caminhos distintos.

Para complementar, é possível que qualquer compreensão do comportamento de fenômenos hídricos na subsuperfície desperte interesses que vão além da comunidade de pesquisadores da hidrogeologia ou de especialistas em representação de leis da natureza através da modelagem matemática. Ou seja, programadores gráficos e os desenvolvedores de tecnologias computacionais, com algum conhecimento avançado de métodos numéricos, também podem contribuir nos estudos e desenvolvimentos relacionados nessa importante área da Dinâmica dos Fluidos Computacional.

REFERÊNCIAS

AHRENS, J. *et al.*, **ParaView: An End-User Tool for Large Data Visualization, Visualization Handbook**, Elsevier, 2005,

BEAR, J. **Hydraulics of Groundwater**. New York: McGraw Hill, 2012.

CLEARY, R. **Águas Subterrâneas**. Clean Environment Brasil e Princeton Groundwater, 2007.

FENICS PROJECT. **Documentation for DOLFIN-1.6.0 (Python)**. Disponível em: <https://fenicsproject.org/olddocs/dolfin/1.6.0/python/demo/index.html>. Acesso em: 13 jul. 2018.

HUNTER, J.D. **Matplotlib: A 2D graphics environment**. Computing In Science & Engineering, v.9, n.3, pg. 90-95, 2007.

ISTOK, J. **Groundwater Modeling by the Finite Element Method**. Water Resources Monograph 13, American Geophysical Union, 1989.

LOGG, A. *et al.* **Automated Solution of Differential Equations by the Finite Element Method**.

Springer, 2012.

REILLY, T. E.; FRANKE, L.; BENNETT, G. D. **The Principle of superposition and its application in ground-water hydraulics**, U.S.GEOLOGICAL SURVEY, Open-File Report 84-459, 1984.

SANTOS, J. P. M. **Método Multigrid Algébrico: Reutilização das Estruturas Multigrid no Transporte de Contaminantes**. 2015. 191 f. Tese (Doutorado em Ciências) – Escola de Engenharia de Sao Carlos, EESC/USP, Sao Carlos, 2015.

VAN DER WALT, S.; COLBERT, S. C.; VAROQUAUX, G. **The NumPy Array: A Structure for Efficient Numerical Computation**, Computing in Science Engineering, v13, n2, 2011.

VERFÜRTH, R. **Adaptive Finite Element Methods**: Lecture Notes Winter Term 2013/14. Disponível em <http://www.ruhr-unibochum.de/num1/files/lectures/AdaptiveFEM.pdf>. Acesso em: 24 maio 2018.

SOBRE O ORGANIZADOR

FELIPE ANTONIO MACHADO FAGUNDES GONÇALVES Mestre em Ensino de Ciência e Tecnologia pela Universidade Tecnológica Federal do Paraná(UTFPR) em 2018. Licenciado em Matemática pela Universidade Estadual de Ponta Grossa (UEPG), em 2015 e especialista em Metodologia para o Ensino de Matemática pela Faculdade Educacional da Lapa (FAEL) em 2018. Atua como professor no Ensino Básico e Superior. Trabalha com temáticas relacionadas ao Ensino desenvolvendo pesquisas nas áreas da Matemática, Estatística e Interdisciplinaridade.

Agência Brasileira do ISBN
ISBN 978-85-7247-348-4

