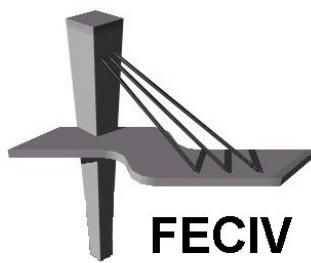


Dissertação DE MESTRADO

**MODELAGEM NUMÉRICA E COMPUTACIONAL
DA HIDRODINÂMICA DE RESERVATÓRIOS
UTILIZANDO AS EQUAÇÕES
BIDIMENSIONAIS DE SAINT-VENANT**

GUILHERME RESENDE MAGRO



FACULDADE DE ENGENHARIA CIVIL
Programa de Pós-graduação em Engenharia Civil



Guilherme Resende Magro

**MODELAGEM NUMÉRICA E COMPUTACIONAL DA
HIDRODINÂMICA DE RESERVATÓRIOS UTILIZANDO
AS EQUAÇÕES BIDIMENSIONAIS DE SAINT-VENANT**

Dissertação apresentada à Faculdade de Engenharia Civil da Universidade Federal de Uberlândia, como parte dos requisitos para a obtenção do título de **Mestre em Engenharia Civil**.

Orientador: Prof. Dr. José Eduardo Alamy Filho

Uberlândia, dezembro de 2018

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

M212m Magro, Guilherme Resende, 1993-
2018 Modelagem numérica e computacional da hidrodinâmica de reservatórios utilizando as equações bidimensionais de Saint-Venant [recurso eletrônico] / Guilherme Resende Magro. - 2018.

Orientador: José Eduardo Alamy Filho.
Dissertação (mestrado) - Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Civil.
Modo de acesso: Internet.
Disponível em: <http://dx.doi.org/10.14393/ufu.di.2019.301>
Inclui bibliografia.
Inclui ilustrações.

1. Engenharia civil. 2. Escoamento. 3. Equações diferenciais parciais. I. Alamy Filho, José Eduardo (Orient.) II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Engenharia Civil. III. Título.

CDU: 624

Maria Salete de Freitas Pinheiro - CRB6/1262



UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Coordenação do Programa de Pós-Graduação em Engenharia Civil
Av. João Naves de Ávila, 2121, Bloco 1Y - Bairro Santa Monica, Uberlândia-MG, CEP 38400-902
Telefone: 34 3239-4137 - www.feciv.ufu.br/ppgec - posgradcivil@ufu.br



ATA

ATA DA DEFESA DE DISSERTAÇÃO DE MESTRADO ACADÊMICO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL - PPGEC

ATA Nº: 232/2018

CANDIDATO: Guilherme Resende Magro **Nº. Matrícula:** 11622ECV004

ORIENTADOR: Prof. Dr. José Eduardo Alamy Filho

TÍTULO: "Modelagem numérica e computacional da hidrodinâmica de reservatórios utilizando as equações bidimensionais de Saint-Venant".

ÁREA DE CONCENTRAÇÃO: Engenharia Urbana, Recursos Hídricos e Saneamento Ambiental

LINHA DE PESQUISA: Recursos Hídricos e Saneamento Ambiental

PROJETO DE PESQUISA: Gestão de reservatórios: incentivos aos serviços ambientais para a sustentabilidade dos regimes hidrossedimentológicos

DATA DA DEFESA: 18 de dezembro de 2018

LOCAL: Sala de Projeções Prof. Celso Franco de Gouvêa, bloco 1Y.

HORÁRIO INÍCIO/TÉRMINO: **14:00 - 18:00**

Reuniu-se na **Sala de Projeções Prof. Celso Franco de Gouvêa, bloco 1Y - Campus Santa Mônica** da Universidade Federal de Uberlândia, a Banca Examinadora, designada pelo Colegiado do PPGEC, assim composta pelos professores doutores: **Iran Eduardo Lima Neto – CT/UFC; Carlos Eugênio Pereira – FECIV/UFU e José Eduardo Alamy Filho – FECIV/UFU**, orientador do candidato. Ressalta-se que o Prof. Iran Eduardo Lima Neto participou da banca via webconferência desde a cidade Fortaleza/CE e o discente e os demais membros da banca participaram in loco.

Iniciando os trabalhos o presidente da mesa Prof. Dr. José Eduardo Alamy Filho, apresentou a Comissão Examinadora e concedeu ao discente a palavra para a exposição do trabalho. A seguir, o senhor presidente concedeu a palavra aos examinadores, que passaram a arguir o candidato. Ultimada a arguição, a Banca, em sessão secreta, atribuiu os conceitos finais. Em face do resultado obtido, a Banca Examinadora considerou o candidato **APROVADO**. Esta defesa de Dissertação de Mestrado Acadêmico é parte dos requisitos necessários à obtenção do título de Mestre. O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU. Nada mais havendo a tratar foram encerrados os trabalhos e foi lavrada a presente ata que após lida e aprovada foi assinada pela Banca Examinadora.

Membros titulares:

Prof. Dr. José Eduardo Alamy Filho – FECIV/UFU

Prof. Dr. Iran Eduardo Lima Neto – CT/UFC

Prof. Dr. Carlos Eugênio Pereira – FECIV/UFU

Uberlândia, 18 de dezembro de 2018.



Documento assinado eletronicamente por **José Eduardo Alamy Filho, Professor(a) do Magistério Superior**, em 19/12/2018, às 11:05, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Iran Eduardo Lima Neto, Usuário Externo**, em 19/12/2018, às 12:10, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Carlos Eugênio Pereira, Professor(a) do Magistério Superior**, em 20/12/2018, às 10:20, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0893824** e o código CRC **EED400B0**.

Referência: Processo nº 23117.086200/2018-77

SEI nº 0893824

A GRADECIMENTOS

Agradeço primeiramente aos meus pais, Paulo e Denise, pelo apoio, motivação e amor que sempre me deram e pelos quais procuro sempre evoluir pessoalmente e profissionalmente.

À minha irmã, Isabela, pelo carinho e por sempre estar ao meu lado nos momentos difíceis, na qual sem o seu apoio eu não teria conseguido chegar aonde eu cheguei.

Aos meus colegas de mestrado, pelos conhecimentos partilhados e pelas inúmeras ajudas.

Aos amigos de Mineiros e Uberlândia, pelas cervejas, risadas e conselhos.

Ao meu orientador José Eduardo Alamy Filho, por aceitar me acompanhar ao longo desta empreitada, e que se mostrou um grande amigo, sempre atencioso e presente, me ajudando de diversas formas.

Aos demais professores, que tanto colaboraram para a minha formação profissional e pessoal.

Por fim, agradeço à Deus por ter me dado a oportunidade de aprender e viver tudo isto, com saúde e ao lado de pessoas tão especiais.

“Se você pensa que pode ou se pensa que não pode, de qualquer forma você está certo.”

Henry Ford

Magro, G. R. Modelagem numérica e computacional da hidrodinâmica de reservatórios utilizando as equações bidimensionais de Saint-Venant. 118 p. Dissertação de Mestrado, Faculdade de Engenharia Civil, Universidade Federal de Uberlândia, 2018.

RESUMO

Acumulações ou represamentos de água são soluções utilizadas tanto para abastecimento de populações, como para inúmeras atividades, entre elas a geração de energia e a irrigação. Os estudos da hidrodinâmica de corpos de água naturais ou artificiais permitem descrições mais detalhadas acerca do comportamento dos campos de velocidade e de profundidade e são de suma importância para que se possa planejar o uso destes recursos, de forma a garantir aproveitamento sustentável e eficiente. Existem programas computacionais no mercado que executam simulações hidrodinâmicas, entretanto, apresentam certas peculiaridades que, embora não sejam limitadoras, exigem mais tempo de aprendizado por parte dos usuários. O objetivo deste trabalho é, partindo das Equações de Saint-Venant em 2D, produzir o SimHidro2D, que se trata de uma rotina de cálculos feita em um programa computacional de simples aprendizagem para que se torne mais fácil e prático a realização de simulações numéricas de escoamentos. Foram realizados testes e comparados os resultados obtidos com os resultados de outros pesquisadores para a validação da rotina de cálculo. Posteriormente, foi feita a simulação hidrodinâmica do Reservatório Bom Jardim, situado no município de Uberlândia no estado de Minas Gerais, e obtido os seus campos de profundidades e velocidades e, em seguida, realizado uma análise comparativa destes resultados com os resultados da simulação feita com o auxílio da ferramenta computacional HEC-RAS. A ferramenta desenvolvida provou ser capaz de simular o comportamento de corpos hídricos gerando bons resultados.

Palavras-chave: Equações de Saint-Venant 2D; Equações de águas rasas; Modelagem numérica; Simulação numérica; Equações diferenciais parciais; HEC-RAS.

MAGRO, G. R. Numerical and computational modeling of reservoir hydrodynamics using the two-dimensional equations of Saint-Venant. 118 pp. MSc Dissertation, College of Civil Engineering, Federal University of Uberlândia, 2018.

ABSTRACT

Accumulations or water impoundments are solutions used both for population supply and for numerous activities, including power generation and irrigation. Studies of the hydrodynamics of natural or artificial bodies of water allow for more detailed descriptions of the behavior of velocity and depth fields and are of paramount importance in order to plan the use of these resources in order to ensure sustainable and efficient utilization. There are computer programs in the market that execute hydrodynamic simulations, however, they present certain peculiarities that, although they are not limiting, require more time of learning on the part of the users. The purpose of this work is, starting from the Saint Venant equations in 2D, to produce the SimHidro2D, which is a routine of calculations made in a computer program of simple learning to make it easier and more practical to perform numerical simulations of flows. Tests were performed and the results obtained were compared with the results of other researchers for the validation of the calculation routine. Subsequently, was made the hydrodynamic simulation of the Bom Jardim Reservoir, located in the municipality of Uberlândia in the state of Minas Gerais, and obtained its fields of depths and velocities and then performed a comparative analysis of these results with the results of the simulation made with the aid of the computational tool HEC-RAS. The tool developed proved to be able to simulate the behavior of water bodies generating good results.

Keywords: 2D Equations of Saint-Venant; Shallow water equations; Numerical modeling; Numerical simulation; Partial differential equations; HEC-RAS.

SÍMBOLOS, ABREVIATURAS E SIGLAS

SÍMBOLOS

∇^2	- Operador Laplace
A	- Coeficiente de viscosidade (m^2/s)
A_m	- Área molhada (m^2)
C_D	- Coeficiente de arrasto de fundo
D	- Constante empírica para cálculo da viscosidade turbulenta
$dfh1$	- Diferença de altitudes entre as superfícies líquidas das células $i+1,j$ e i,j (m)
$dfh2$	- Diferença de altitudes entre as superfícies líquidas das células $i-1,j$ e i,j (m)
$dfh3$	- Diferença de altitudes entre as superfícies líquidas das células $i,j+1$ e i,j (m)
$dfh4$	- Diferença de altitudes entre as superfícies líquidas das células $i,j-1$ e i,j (m)
$dfu1$	- Comprimento de parede em contato com a água na face entre os nós i,j e $i,j+1$ (m)
$dfu2$	- Comprimento de parede em contato com a água na face entre os nós $i+1,j$ e $i+1,j+1$ (m)
$dfu3$	- Comprimento de parede em contato com a água na face entre os nós i,j e $i,j-1$ (m)
$dfu4$	- Comprimento de parede em contato com a água na face entre os nós $i+1,j$ e $i+1,j-1$ (m)
$dfv1$	- Comprimento de parede em contato com a água na face entre os nós i,j e $i-1,j$ (m)
$dfv2$	- Comprimento de parede em contato com a água na face entre os nós $i,j+1$ e $i-1,j+1$ (m)
$dfv3$	- Comprimento de parede em contato com a água na face entre os nós $i,j+1$ e $i+1,j+1$ (m)
$dfv4$	- Comprimento de parede em contato com a água na face entre os nós i,j e $i+1,j$ (m)

f	- Constante de Coriolis (1/s)
g	- Aceleração gravitacional (m/s ²)
gz	- Força gravitacional por unidade de massa na direção do eixo Z (m/s ² .kg)
h	- Profundidade de água (m)
h_{cap}	- Profundidade de água instantânea na captação de água (m)
h_{rel}	- Profundidade relativa de água (m)
$hT1$	- Altura da superfície da água do nó $i+1,j$ em relação ao solo do nó i,j (m)
$hT2$	- Altura da superfície da água do nó $i-1,j$ em relação ao solo do nó i,j (m)
$hT3$	- Altura da superfície da água do nó $i,j+1$ em relação ao solo do nó i,j (m)
$hT4$	- Altura da superfície da água do nó $i,j-1$ em relação ao solo do nó i,j (m)
i	- Numeração dos nós ao longo do eixo X
j	- Numeração dos nós ao longo do eixo Y
n	- Coeficiente de Manning
N_{CFL}	- Número de Courant-Friedrichs-Lewy
p	- Pressão (Pa)
P_m	- Perímetro molhado (m)
Q	- Vazão (m ³ /s)
Q_{cap}	- Vazão adotada de captação (m ³ /s)
R_h	- Raio hidráulico (m)
t	- Tempo atual da simulação (s)
$t_{máx}$	- Tempo final da simulação adotado (s)
u	- Velocidade média de escoamento ao longo da profundidade na direção do eixo X (m/s)
v	- Velocidade média de escoamento ao longo da profundidade na direção do eixo Y (m/s)
vel_{cap}	- Velocidade de escoamento calculada na captação (m/s)
v_x	- Componente da velocidade do escoamento na direção do eixo X (m/s)
v_y	- Componente da velocidade do escoamento na direção do eixo Y (m/s)

v_z	- Componente da velocidade do escoamento na direção do eixo Z (m/s)
W	- Soma da profundidade de água com a altitude do solo da célula analisada (m)
Z	- Altitude do terreno ou do fundo do corpo de água (m)
γ	- Coeficiente de viscosidade (m^2/s)
Δt	- Incremento de tempo entre cada iteração (s)
Δt_o	- Incremento de tempo inicial (s)
ΔVol	- Variação de volume (m^3/s)
Δx	- Distância entre nós ao longo do eixo X (m)
Δy	- Distância entre nós ao longo do eixo Y (m)
ΔZ	- Diferença de altitudes do solo (m)
μ	- Viscosidade dinâmica ($N.s/m^2$)

ABREVIATURAS

$2D$	- 2 Dimensões
kg	- Quilograma
m	- Metro
N	- Newton
Pa	- Pascal
s	- Segundo

SIGLAS

CFL	- <i>Courant-Friedrichs-Lowy</i>
$COPPE$	- Coordenação de Programas de Pós Graduação em Engenharia
EDP	- Equação Diferencial Parcial
$HEC-RAS$	- <i>Hydrologic Engineering Center's – River Analysis System</i>
MDF	- Método das Diferenças Finitas
MEF	- Método dos Elementos Finitos
RMA	- <i>Resource Management Associates, Inc.</i>
$SimHidro2D$	- Simulação Hidrodinâmica em 2 Dimensões
$SISBAHIA$	- Sistema Base de Hidrodinâmica Ambiental

UFRJ - Universidade Federal do Rio de Janeiro

USACE - *U. S. Army Corps of Engineers*

VBA - *Visual Basic for Applications*

LISTA DE **I**LUSTRAÇÕES

FIGURAS

Figura 1 – Vista superior de malha utilizando o Método dos Elementos Finitos.....	14
Figura 2 – Simulação de uma gota de água utilizando o Método Sem Malha.	15
Figura 3 - Área discretizada utilizando o Método do Ponto Finito.	16
Figura 4 - Nomenclaturas das posições dos nós na malha regular.	17
Figura 5 - Interface do programa computacional SisBaHiA.	19
Figura 6 - Resultado de uma simulação feita através do Mike 21C.	21
Figura 7 - Interface do programa computacional River2D.....	22
Figura 8 - Interface da saída de resultados do programa HEC-RAS.....	23
Figura 9 - Fluxograma geral da rotina de cálculos.	27
Figura 10 - Posicionamento das variáveis calculadas.	32
Figura 11 - Condição de total reflexão.	33
Figura 12 – Condição de contorno sem reflexão.	34
Figura 13 - Condição de nó seco vizinho de nós secos ou molhados baixos.	36
Figura 14 – Nó seco vizinho de nó molhado alto e nós secos.	37
Figura 15 - Condição de nó molhado vizinho de nó seco alto.	38
Figura 16 - Média das profundidades de água.....	40
Figura 17 - Uso da profundidade relativa para o cálculo de h	42
Figura 18 - Cálculo de raio hidráulico.....	45
Figura 19 - Situações de relevos durante o cálculo de perímetro molhado.	46
Figura 20 - Aba de configuração da versão de testes do SimHidro2D.....	50
Figura 21 - Aba de configuração do SimHidro2D.	50
Figura 22 - Aba de entrada de dados do SimHidro2D.	51
Figura 23 - Aba de saída de resultados do SimHidro2D.	51
Figura 24 - Composição inicial do Teste 1.....	52
Figura 25 - Resultados nos tempos 0,5 a 3,0 segundos do Teste 1.	54
Figura 26 - Resultados nos tempos 3,5 a 5,0 segundos do Teste 1	55
Figura 27 - Geometria do teste de rompimento de barragem.	57
Figura 28 - Disposição inicial do teste 2.	57
Figura 29 - Resultados de profundidade de água do teste 2.	58

Figura 30 - Mapa de contorno dos resultados de profundidade de água do teste 2.....	59
Figura 31 - Mapa de contorno dos resultados de velocidade do teste 2.....	59
Figura 32 - Comparação dos resultados do teste 2 com os resultados de: a) Li e Fan (2017); b) Zhendong e Junqiang (2015); c) Erduran (2013).	61
Figura 33 - Geometria do teste 3.....	62
Figura 34 - Resultados de profundidade de água do teste 3.....	63
Figura 35 - Resultados de velocidade de escoamento do teste 3.....	65
Figura 36 - Localização do Reservatório Bom Jardim.....	67
Figura 37 - Entradas e saídas de água do reservatório.....	68
Figura 38 - Mapa topo-batimétrico digital.....	69
Figura 39 - Mapa topo-batimétrico digital com barragem.....	70
Figura 40 – Resultados do SimHidro2D de profundidades de água após 1 hora de simulação.	
.....	73
Figura 41 – Resultados do SimHidro2D de velocidades de escoamento após 1 hora de simulação.....	74
Figura 42 - Resultados do HEC-RAS de profundidades de água após 1 hora de simulação..	75
Figura 43 - Resultados do HEC-RAS de velocidades de escoamento após 1 hora de simulação.....	76
Figura 44 - Divisão da área de estudo.....	78
Figura 45 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 1.....	79
Figura 46 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 2.....	80
Figura 47 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 3.....	81
Figura 48 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 4.....	82
Figura 49 - Escoamentos com sentido inverso ao sentido do escoamento principal, na simulação de 30 minutos do HEC-RAS.....	85

QUADROS

Quadro 1 – Configurações da condição de contorno com reflexão total.....	33
Quadro 2 - Configurações da condição de contorno sem reflexão.....	34
Quadro 3 - Obtenção da profundidade de água e altitude do solo nas faces das células.....	41
Quadro 4 - Esquema de profundidades relativas para o termo de suavização de h	43

TABELAS

Tabela 1 - Valores de coeficiente para cálculo da viscosidade turbulenta.	31
Tabela 2 - Configurações para o teste 1.	53
Tabela 3 - Configurações para o teste 2.	58
Tabela 4 - Configurações utilizadas no teste 3.	62
Tabela 5 - Valores utilizados para a entrada contínua de água.	63
Tabela 6 – Vazões fixas de entrada e saída utilizadas.	70
Tabela 7 - Configurações utilizadas na simulação do reservatório.	71
Tabela 8 - Valores de profundidades e velocidades utilizados para as células de entrada de vazão.....	71

SUMÁRIO

1 INTRODUÇÃO.....	8
2 REVISÃO BIBLIOGRÁFICA.....	11
2.1 Equações de Saint-Venant.....	11
2.2 Condição de estabilidade de Courant-Friedrichs-Lowy (CFL).....	13
2.3 Métodos para a discretização espacial da área de estudo.....	14
2.3.1 Método dos Elementos Finitos.....	14
2.3.2 Método Sem Malha.....	14
2.3.3 Método das Diferenças Finitas.....	16
2.4 Programas computacionais que utilizam as equações bidimensionais de Saint-Venant.....	18
2.4.1 SisBaHiA.....	18
2.4.2 Mike 21C.....	20
2.4.3 River2D.....	21
2.4.4 HEC.....	23
RAS.....	
2.5 Método Numérico para a solução de Equações Diferenciais Parciais.....	24
3 METODOLOGIA.....	26
3.1 Linguagem de programação e programas utilizados.....	26
3.2 Fluxograma geral da rotina de cálculos.....	26
3.3 Cálculo do incremento de tempo (Δt).....	28
3.4 Resolução das equações governantes.....	29
3.5 Condições de contorno.....	32
3.5.1 Condição de contorno: Total reflexão.....	33
3.5.2 Condição de contorno: Sem reflexão.....	34
3.6 Condições internas.....	35
3.6.1 Nô seco vizinho de nós secos ou de nós molhados baixos.....	35
3.6.2 Nô seco vizinho de nós molhados altos e nós secos ou nós molhados baixos.....	36
3.6.3 Nô molhado vizinho de nós secos altos.....	37
3.7 Esquema de cálculo para a profundidade de água e altitude do solo nas faces	

CAPÍTULO 1

INTRODUÇÃO

Acumulações ou represamentos de água são soluções utilizadas tanto para abastecimento de populações, como para inúmeras atividades, entre elas a geração de energia e a irrigação. Os estudos da hidrodinâmica de corpos de água naturais ou artificiais permitem descrições mais detalhadas acerca do comportamento dos campos de velocidade e de profundidade e são de suma importância para que se possa planejar o uso destes recursos, de forma a garantir aproveitamento sustentável e eficiente.

Neste caso, um dos estudos refere-se à modelagem numérica do escoamento dentro e a jusante de reservatórios, que pode ser aplicado para a análise de rompimento de barragens (HE et al., 2017; HU e SUEYOSHI, 2010; KAO e CHANG, 2012; SINGH, ALTINAKAR e DING, 2011; ZHOU et al. 2017) e, através dos mapas de inundação gerados, pode-se prever regiões de risco e possíveis danos causados.

Outra aplicação da modelagem numérica para escoamentos é a análise da hidrodinâmica no interior de reservatórios (DUMKWU, ISLAM e CARLSON, 2012). Neste tipo de estudo, podem-se identificar zonas mais propícias à sedimentação, fenômeno natural que conduz ao assoreamento progressivo destes corpos hídricos; e também zonas sujeitas à recirculação de água, redundando em setores favoráveis a processos de eutrofização, por exemplo.

Muitas cidades no Brasil e no mundo têm o seu abastecimento de água através de captação superficial em reservatórios de acumulação e de regularização de vazões, bem como em zonas represadas por barragens de nível. Uma mesma represa pode apresentar qualidades diferentes de água em diferentes posições da massa líquida. Em regiões de baixas velocidades, há uma maior deposição de areia e de outras partículas, enquanto que, em regiões de grande recirculação de água, podem ocorrer acúmulos transitórios de poluentes em função de tempos de residência maiores.

Nesse sentido, conhecer os escoamentos dentro de um reservatório é importante para obter um melhor posicionamento da captação de água, para que se possa obter uma água de melhor qualidade a fim de reduzir os custos com sua potabilização nas Estações de Tratamento de Água.

A modelagem numérica de escoamento, incluindo equações de concentração de sedimentos, é uma maneira de se prever estas regiões de maiores concentrações de sedimentos (GAMVROUDIS et al., 2015; KESHTPOOR et al., 2015; ZHENDONG e JUNQIANG, 2015; ZHENDONG, JUNQIANG e ZHENGHUI, 2015).

Uma outra importante aplicação de simulação numérica é a previsão de inundações causadas tanto por enchentes (VACONDIO et al., 2016), quanto por tsunamis (GRILLI et al., 2015; KILINC, HAYIR e CIGIZOGLU, 2009; SATRIA et al., 2012), principalmente em regiões no mundo onde são frequentes estes tipos de desastres naturais. Por exemplo, simulações numéricas de um tsunami podem orientar medidas preventivas, identificar zonas inundáveis e setores com elevadas tensões cisalhantes entre a água e o terreno. Isto, aliado a medidas de alerta e estratégias de evacuação, pode contribuir para salvar vidas.

Logo, é possível notar que as modelagens de escoamentos de águas superficiais, são de grande utilidade, tanto no ponto de vista econômico quanto social. A despeito disto, alguns programas computacionais têm entradas de dados complexas, o que acaba dificultando a execução de simulações, mesmo para corpos de água de pequeno porte. A própria descrição matemática do fenômeno, governado por equações diferenciais parciais, torna as soluções destes problemas não triviais, quando tratadas em duas ou três dimensões.

Diante disto, o objetivo geral deste trabalho é, partindo das Equações de Saint-Venant em duas dimensões, desenvolver uma rotina de cálculo, em uma linguagem computacional amigável, para simular a hidrodinâmica em reservatórios. Isto pressupõe a obtenção de seus campos de velocidades e de profundidades de água. Esta rotina de cálculo foi batizada de SimHidro2D, abreviação de Simulação Hidrodinâmica em 2 Dimensões. Dentre os objetivos secundários, citam-se a construção do código em linguagem *Visual Basic for Applications* (VBA), a simulação do reservatório do Bom Jardim, situado ao sul do perímetro urbano de Uberlândia (MG), além de comparações com um programa comercial de uso livre. Neste último caso, as comparações servirão para elucidar se o escoamento simulado com o código proposto apresenta o mesmo padrão daquele simulado pelo código comercial.

Esta dissertação é constituída de cinco capítulos, como segue:

Capítulo 1: Introdução, que engloba as aplicações da simulação numérica e o objetivo geral deste trabalho;

Capítulo 2: Revisão bibliográfica, que reúne informações sobre as equações governantes, condição de estabilidade, métodos para a discretização espacial, programas que utilizam as equações de Saint-Venant e a solução numérica utilizada no estudo;

Capítulo 3: Metodologia, que inclue todas as partes da construção e informações necessárias para o desenvolvimento da rotina de cálculo, onde é descrito a linguagem de programação e programas utilizados, cálculo do incremento de tempo, resolução das equações governantes, condições de contorno, condições internas, esquema de cálculo para a profundidade de água e altitude do solo nas faces das células; esquema de cálculo de profundidades relativas no cálculo de h , esquema de cálculo de profundidades relativas no termo de suavização de h , esquema de cálculo dos raios hidráulicos, esquema de cálculo de profundidades relativas e alterações no Método de Lax-Friedrichs;

Capítulo 4: Resultados e discussões, que contempla as interfaces de configuração, entrada de dados e saída dos resultados no SimHidro2D; apresentação e discussão dos três principais testes elaborados para a análise de confiabilidade da rotina de cálculo do SimHidro2D; simulação do Reservatório Bom Jardim utilizando o SimHidro2D; simulação do Reservatório Bom Jardim utilizando o HEC-RAS; análise comparativa dos resultados;

Capítulo 5: Conclusões, onde é apresentado um desfecho do trabalho, comentando as conclusões obtidas e recomendações para trabalhos futuros.

Referências Bibliográficas.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

CAPÍTULO 2

REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta uma síntese de aspectos teóricos básicos, porém importantes sobre o equacionamento e modelagem de escoamentos superficiais bidimensionais. Considera-se que estas descrições sejam úteis para que o leitor possa se familiarizar com os principais símbolos, equações e métodos numéricos aplicados na solução do problema. Além disto, também são abordados métodos e contéudos apresentados em publicações científicas e programas computacionais que simulam escoamentos bidimensionais em corpos de água. Desta forma, fornece-se uma base conceitual para a compreensão do tema e do desenvolvimento do trabalho.

2.1 Equações de Saint-Venant

As equações de Saint-Venant foram obtidas a partir da integração através da profundidade nas equações de continuidade (Equação 1) e conservação de quantidade de movimento (Equações 2, 3 e 4) de Navier-Stokes para fluidos incompressíveis.

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0 \quad (1)$$

$$\frac{\partial v_x}{\partial t} + v_x \frac{\partial v_x}{\partial x} + v_y \frac{\partial v_x}{\partial y} + v_z \frac{\partial v_x}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\mu}{\rho} \nabla^2 v_x \quad (2)$$

$$\frac{\partial v_y}{\partial t} + v_x \frac{\partial v_y}{\partial x} + v_y \frac{\partial v_y}{\partial y} + v_z \frac{\partial v_y}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\mu}{\rho} \nabla^2 v_y \quad (3)$$

$$\frac{\partial v_z}{\partial t} + v_x \frac{\partial v_z}{\partial x} + v_y \frac{\partial v_z}{\partial y} + v_z \frac{\partial v_z}{\partial z} = g_z - \frac{1}{\rho} \frac{\partial p}{\partial z} + \frac{\mu}{\rho} \nabla^2 v_z \quad (4)$$

Nas Equações 1 a 4, v_x , v_y e v_z são as componentes de velocidade ao longo das direções dos eixos X, Y e Z, respectivamente, em m/s; g_z a força gravitacional por unidade de massa ao longo da direção Z em $\text{m/s}^2/\text{kg}$; μ a viscosidade dinâmica, em N.s/m^2 ; p a pressão, em Pa; ∇^2 é o operador de Laplace (Equação 5).

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \quad (5)$$

Para se realizar a integração das equações de Navier-Stokes, ao longo da profundidade dos corpos de água, são feitas algumas simplificações, dentre elas:

- Nenhuma partícula de água abandona o volume pela superfície livre;
- Declividades de fundo pequenas;
- Fluido é incompressível e com densidade constante;
- Velocidade vertical da água desconsiderada;
- Velocidades longitudinais constantes através da altura;
- Fundo considerado sem irregularidades, dentro do volume discreto analisado.

Estas hipóteses são plausíveis para a simulação de corpos hídricos cujas profundidades são nitidamente inferiores ao comprimento e à largura. Estes são casos típicos de reservatórios, lagoas, estuários, baías e de trechos largos de rios.

Após a integração das equações de Navier-Stokes para fluidos incompressíveis, ao longo da profundidade do corpo de água, têm-se as equações governantes, chamadas também de Equações de Saint-Venant (Equações 6, 7 e 8) ou de equações de águas rasas.

$$\frac{\partial h}{\partial t} + \frac{\partial(uh)}{\partial x} + \frac{\partial(vh)}{\partial y} = 0 \quad (6)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = A \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - g \frac{\partial(h+Z)}{\partial x} \quad (7)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = A \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - g \frac{\partial(h+Z)}{\partial y} \quad (8)$$

Sendo, u a velocidade média de escoamento ao longo da profundidade na direção do eixo X, em m/s; v a velocidade média de escoamento ao longo da profundidade na direção do eixo Y, em m/s; A um coeficiente de viscosidade, em m^2/s ; h a profundidade de água, em m; Z a altitude do terreno ou do fundo do corpo de água, em m.

Zhendong, Junqiang e Zhenghui (2015) utilizaram estas equações, adicionando um termo na equação de continuidade (Equação 6) para difundir as diferenças entre as profundidades de água de nós vizinhos, e dois termos nas equações de quantidade de movimento (Equações 7 e 8), sendo, um termo referente à aceleração causada devido à rotação do planeta Terra, chamada de constante de Coriolis e o outro referente à dissipação de energia devido ao atrito. Com a adição destes termos, têm-se então as Equações 9, 10 e 11.

$$\frac{\partial h}{\partial t} + \frac{\partial(uh)}{\partial x} + \frac{\partial(vh)}{\partial y} = \gamma \left(\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2} \right) \quad (9)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} - fv = A \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - g \frac{\partial(h+z)}{\partial x} - \frac{C_D u \sqrt{u^2 + v^2}}{h} \quad (10)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + fv = A \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - g \frac{\partial(h+z)}{\partial y} - \frac{C_D v \sqrt{u^2 + v^2}}{h} \quad (11)$$

Sendo, γ um coeficiente de viscosidade, em m^2/s ; f a constante de Coriolis, em s^{-1} ; g a aceleração gravitacional, em m/s^2 ; C_D é o coeficiente de arrasto de fundo.

2.2 Condição de estabilidade de Courant-Friedrichs-Lowy (CFL)

A simulação em marcha temporal é realizada obtendo os campos hidrodinâmicos para diferentes instantes, a partir do tempo inicial, até se atingir o tempo final pretendido. Entretanto, devido a erros residuais contidos nas equações, em consequência das simplificações realizadas para se tornar possível simular matematicamente os efeitos naturais, se torna preciso limitar o incremento de tempo. Esta limitação é necessária para que os erros residuais contidos nas equações não se acumulem ao longo da simulação, desestabilizando as soluções numéricas e desta forma, tornando os resultados incoerentes.

Uma condição de estabilidade utilizada em soluções numéricas de equações diferenciais parciais pelo método das diferenças finitas é a condição de estabilidade de Courant-Friedrichs-Lowy (CFL), mostrada na Equação 12.

$$\max \left[\frac{u \Delta t}{\Delta x} + \frac{v \Delta t}{\Delta y} \right] \leq N_{CFL} \quad (12)$$

Sendo, Δt o incremento de tempo da iteração, em s; Δx a distância entre nós na direção do eixo X, em m; Δy a distância entre nós na direção do eixo Y, em m; N_{CFL} o número de Courant adotado.

A CFL calcula um limite da marcha de tempo das iterações durante os cálculos, garantindo que, ao utilizar um valor de incremento de tempo menor que este limite calculado, seja reduzida a possibilidade de se produzirem resultados incorretos. A acumulação destes resultados espúrios instabiliza a solução numérica, conduzindo à sua literal destruição nos passos de tempo seguintes.

Não há consenso universal acerca do valor do número N_{CFL} , ainda que a literatura destaque alguns valores que servem de balizamento para as simulações. Singh, Altinakar e Ding (2011) utilizaram 0,25 para o número de Courant, enquanto Cannata e Marzocchi (2012) e Sun et al. (2013) utilizaram o valor de 1. Ambos relataram que todas as suas aplicações utilizando estes valores foram estáveis, ou seja, não foram gerados resultados incorretos. O valor utilizado de 0,25 para o número de Courant garante que, durante o intervalo de uma iteração, a maior velocidade da água não percorra 25% da distância da menor célula da simulação.

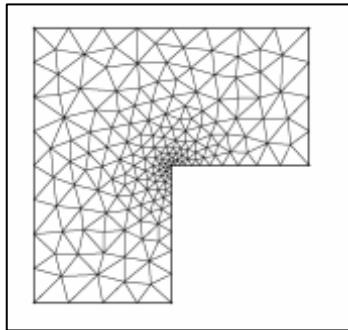
2.3 Métodos para a discretização espacial da área de estudo

Existem vários métodos diferentes para se dividir uma área de estudo, cada um tendo as suas peculiaridades e vantagens de acordo com o seu propósito. Serão apresentados os principais métodos de discretização espacial.

2.3.1 Método dos Elementos Finitos

O Método dos Elementos Finitos (MEF) é o método no qual se divide a área de estudo em vários elementos podendo ter dimensões variadas ou não, como mostrado na Figura 1, em que El Kacimi et al. (2019) utilizaram o MEF para discretizar uma área em L aumentando o número de células na região com maior interesse.

Figura 1 – Vista superior de malha utilizando o Método dos Elementos Finitos.

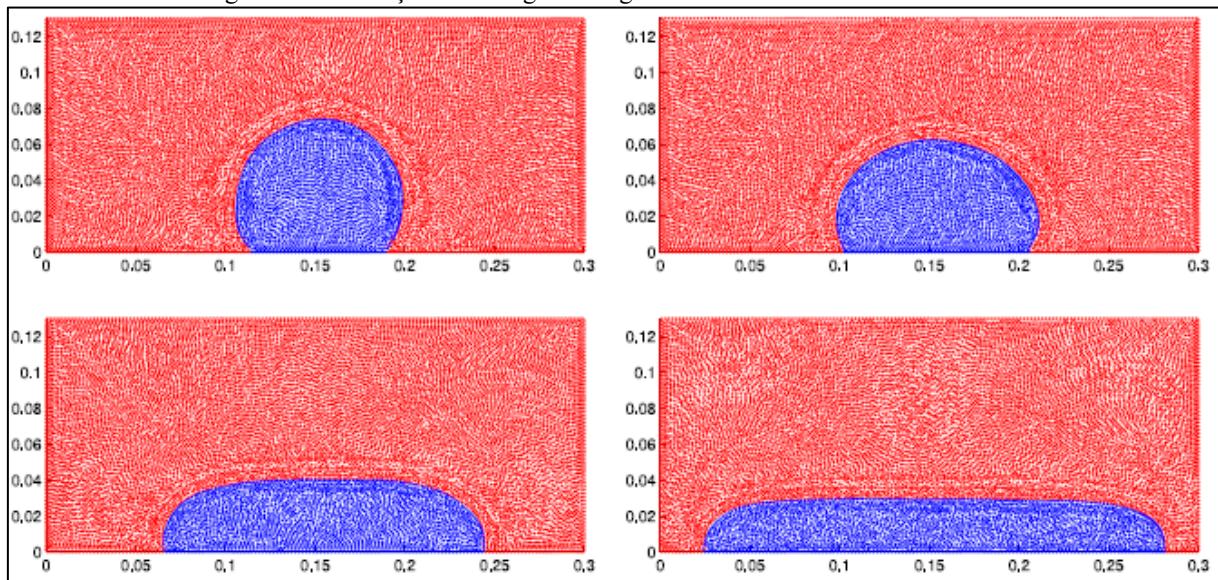


Fonte: El Kacimi (2019).

2.3.2 Método Sem Malha

No Método Sem Malha (*Meshless* também conhecido como *Meshfree*) não há uma malha explícita pré-determinada, sendo que, para cada nó, é atribuído um subdomínio fechado. Estes subdomínios são sobrepostos e a única exigência é que a associação destes domínios cubra toda a área de estudo. Tiwari, Klar e Hardt (2016) utilizaram este método para simular as interações de partículas de água com partículas de gás, como mostrado na Figura 2.

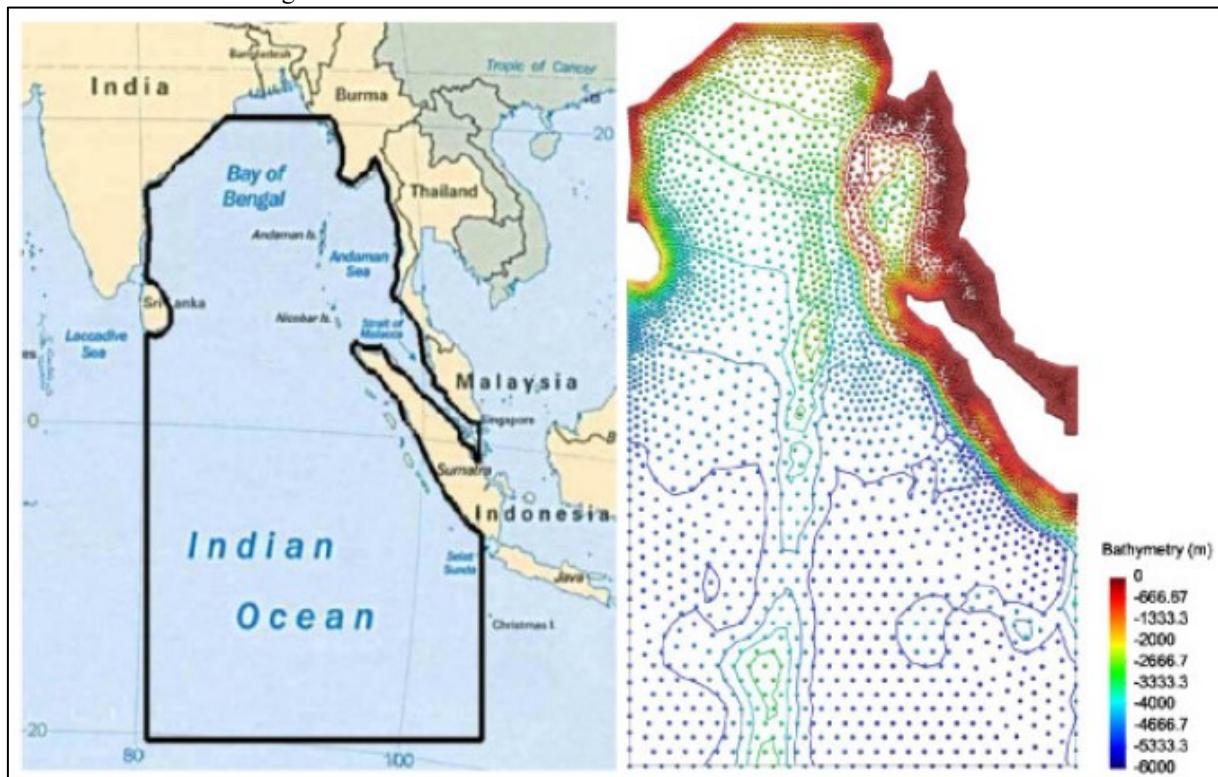
Figura 2 – Simulação de uma gota de água utilizando o Método Sem Malha.



Fonte: Tiwari (2016).

Oñate et al. (1996) apresentaram uma técnica gerada a partir do Método Sem Malha conhecida como Método do Ponto Finito. Através deste método, é gerada uma maior quantidade de pontos de cálculo nas regiões mais críticas da simulação. Na Figura 3 é ilustrada uma simulação de um Tsunami feita por Ortega et al. (2011), no qual foi utilizado o Método do Ponto Finito para discretizar a área de estudo. Neste caso, pode-se observar a maior quantidade de pontos de cálculo nas regiões de maior interesse. Esta capacidade de concentrar pontos em setores estratégicos do escoamento é um dos principais atributos dos métodos sem malha.

Figura 3 - Área discretizada utilizando o Método do Ponto Finito.



Fonte: Ortega (2011).

2.3.3 Método das Diferenças Finitas

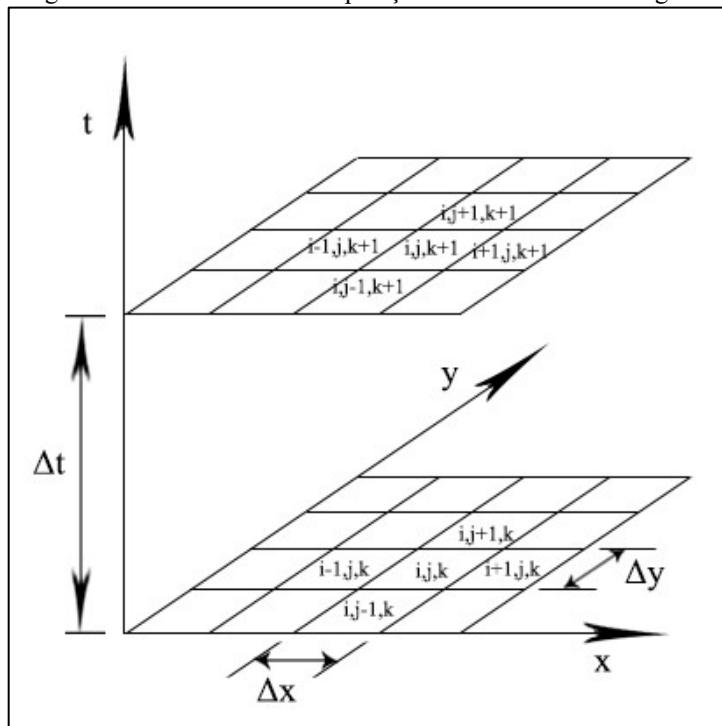
Já o Método das Diferenças Finitas (MDF) normalmente utiliza malha simétrica retangular, o que facilita nos cálculos, pois todos os nós apresentam espaçamentos regulares e constantes ao longo do domínio. Porém, uma desvantagem deste método é que não há como refinar um setor específico do corpo hídrico, no qual se deseja uma melhor resolução dos resultados. Tanto nos métodos sem malha, como nos métodos baseados em elementos finitos, esta adaptatividade da malha é possível. Uma solução para isto, nos métodos de diferenças finitas, consiste em refinar globalmente o domínio, fator que pode resultar em uma quantidade elevada de células.

Ao nomear as variáveis no MDF, é comum colocar, como sobreescrito, o instante (tempo) em que a variável é quantificada e, como subscrito, a posição do nó referente à variável desejada. Como exemplo, uma variável h no tempo $k+1$, referente ao nó $i,j+1$, é escrita desta forma:

$$h_{i,j+1}^{k+1}$$

A direção X é descrita pelo subscrito i , a direção Y pelo subscrito j , e o tempo t descrito pelo sobreescrito k , como mostra a Figura 4.

Figura 4 - Nomenclaturas das posições dos nós na malha regular.



Fonte: Elaborado pelo autor.

Segundo Burden e Faires (2011), o Método das Diferenças Finitas (MDF) consiste em um método de resolução de equações diferenciais através de aproximações, obtidas pela função derivada da série de Taylor, de derivadas por diferenças finitas.

As séries de Taylor para as funções $f(x+h)$ e $f(x-h)$ são citadas pelas Equações 13 e 14.

$$f(x+h) = f(x) + f'(x)h + \frac{f''(x)h^2}{2} + \frac{f'''(x)h^3}{6} + o(h^4) \quad (13)$$

$$f(x-h) = f(x) - f'(x)h + \frac{f''(x)h^2}{2} - \frac{f'''(x)h^3}{6} + o(h^4) \quad (14)$$

Sendo, $f(\cdot)$ a função inicial; $f'(\cdot)$ a derivada primeira da função inicial; $f''(\cdot)$ a derivada segunda da função inicial; $f'''(\cdot)$ a derivada terceira da função inicial; x um valor aleatório independente; h um valor de incremento; $o(h^4)$ um erro de truncamento de quarta ordem gerado por se desprezar o restante da série.

A partir das equações (13) e (14), tem-se então três formas de escrita para a derivada primeira, que são elas:

- Fórmula das diferenças progressivas (Equação 15);

$$f'(x) = \frac{f(x+h) - f(x)}{h} + o(h) \quad (15)$$

- Fórmula das diferenças regressivas (Equação 16);

$$f'(x) = \frac{f(x) - f(x-h)}{h} + o(h) \quad (16)$$

- Fórmula das diferenças centradas (Equação 17).

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + o(h^2) \quad (17)$$

Sendo, $o(h)$ um erro de truncamento de primeira ordem; $o(h^2)$ um erro de truncamento de segunda ordem gerado por se desprezar o restante da série.

2.4 Programas computacionais que utilizam as equações bidimensionais de Saint-Venant

Já foram produzidos alguns programas computacionais para a realização de modelagens numéricas para o estudo de cenários aquáticos, sendo alguns deles gratuitos e outros disponíveis apenas através de licenças comerciais. Neste item, serão abordados alguns programas que realizam simulações de escoamentos hídricos bidimensionais.

2.4.1 SisBaHiA

O Sistema Base de Hidrodinâmica Ambiental (SisBaHiA) é um programa computacional desenvolvido em conjunto pela Área de Engenharia Costeira & Oceanográfica do Programa de Engenharia de Oceânica e pela Área de Banco de Dados do Programa de Engenharia de Sistemas & Computação, ambos da Coordenação de Programas de Pós Graduação em Engenharia (COPPE) da Universidade Federal do Rio de Janeiro (UFRJ), sob os auspícios da Fundação Coppetec.

Este programa é gratuito, entretanto, necessita que sejam adquiridos independentemente dois outros programas comerciais (*Surfer* e *Grapher*) para o seu funcionamento.

Segundo Sampaio (2017), o SisBaHiA “trata-se de um sistema de modelos computacionais em elementos finitos para previsão do escoamento ou movimento das águas e também para a previsão da qualidade das águas ou transporte de grandezas escalares em corpos de água naturais”.

Ainda segundo Sampaio (2017), o SisBaHiA, na versão 9b, pode ser utilizado para os seguintes fins:

- Módulo Hidrodinâmico:

- Circulação hidrodinâmica tridimensional (3D);
- Circulação hidrodinâmica promediada na vertical (2DH);
- Transporte Euleriano de contaminantes dissolvidos na coluna de água;
- Transporte Lagrangeano de contaminantes em qualquer faixa da coluna de água, inclusive flutuantes, ou dissolvidos em toda a coluna de água, inclusive modelo probabilístico;
- Parâmetros de Qualidade da Água:
 - Sal;
 - Temperatura;
 - Sal e Temperatura;
 - Oxigênio Dissolvido & Demanda Bioquímica de Oxigênio;
 - Nitrogênio;
 - Fósforo;
 - Fósforo e Nitrogênio;
- Transporte de Sedimentos, coesivos e não coesivos;
- Geração de ondas;
- Propagação de ondas.

A interface do programa computacional SisBaHiA é mostrada na Figura 5.

Figura 5 - Interface do programa computacional SisBaHiA.



Fonte: Sampaio (2017).

Andrade (2018) utilizou o SisBaHiA para realizar simulações hidrodinâmicas, modelos de taxa de renovação e idade do volume de água, e modelos de qualidade da água para ampliar o

conhecimento sobre o sistema estuarino da Baía de Guanabara, situado no município de Rio de Janeiro.

Carvalho (2018) com o auxílio do SisBaHiA analisou o comportamento hipersalino da lagoa de Araruama, situada no estado do Rio de Janeiro, frente a interferências naturais e antrópicas utilizando modelagem hidrodinâmica e de transporte de sal.

2.4.2 Mike 21C

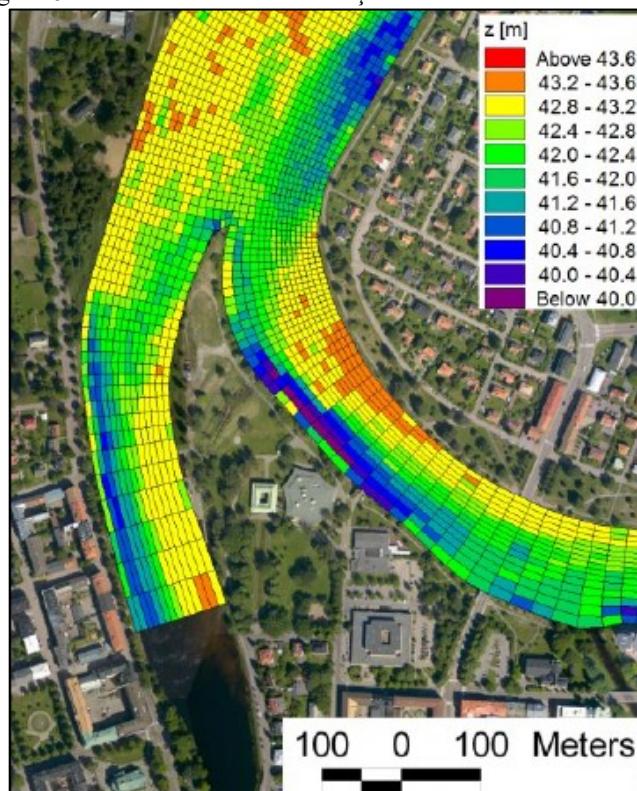
O Mike 21C é um programa computacional pago, feito para simulações de rios e canais. Ele foi produzido pela empresa DHI, a qual é proprietária, além do Mike 21C, de outros programas também voltados para análises de ambientes aquáticos, cada um tendo a sua especialidade.

O programa Mike21C utiliza um esquema de discretização de diferenças finitas com malhas curvilíneas.

Este programa computacional tem capacidade de realizar simulações hidrodinâmicas bidimensionais, em conjunto com modelagens de transporte de sedimentos coesivos e não-coesivos com análises de erosões nas margens dos rios e atualizações momentâneas do leito do corpo hídrico. Além da capacidade de acompanhar a alteração do leito do rio, outra vantagem neste programa é a possibilidade de gerar malhas não-lineares, o que favorece uma melhor resolução dos resultados finais.

São mostrados na Figura 6 os resultados de uma simulação feita através do programa computacional Mike 21C.

Figura 6 - Resultado de uma simulação feita através do Mike 21C.



Fonte: DHI (201-?).

Morianou et al. (2016) utilizaram o Mike21C para obter os campos de velocidades de escoamento e as concentrações de sedimentos suspensos ao longo do Rio *Koiliaris*, situado na Grécia.

Dang e Tran (2016) aplicaram o modelo Mike21C para simular a profundidade de água, distribuição de velocidades de escoamento, e transporte de sedimentos suspensos ao longo do Rio *Bassac*, situado no Vietnã.

Vale ressaltar que por o Mike21C se tratar de um programa computacional comercial, ele não se torna acessível a todos os pesquisadores, sendo necessário, portanto, destinar uma verba para o custeio da licença do programa.

2.4.3 River2D

O River2D é um programa computacional gratuito e foi desenvolvido na Universidade de Alberta (University of Alberta), no Canadá, com o apoio financeiro recebido do Conselho de Pesquisa em Ciências Naturais e Engenharia do Canadá (Natural Sciences and Engineering Research Council of Canada), Departamento de Pesca e Oceano (Department of Fisheries and

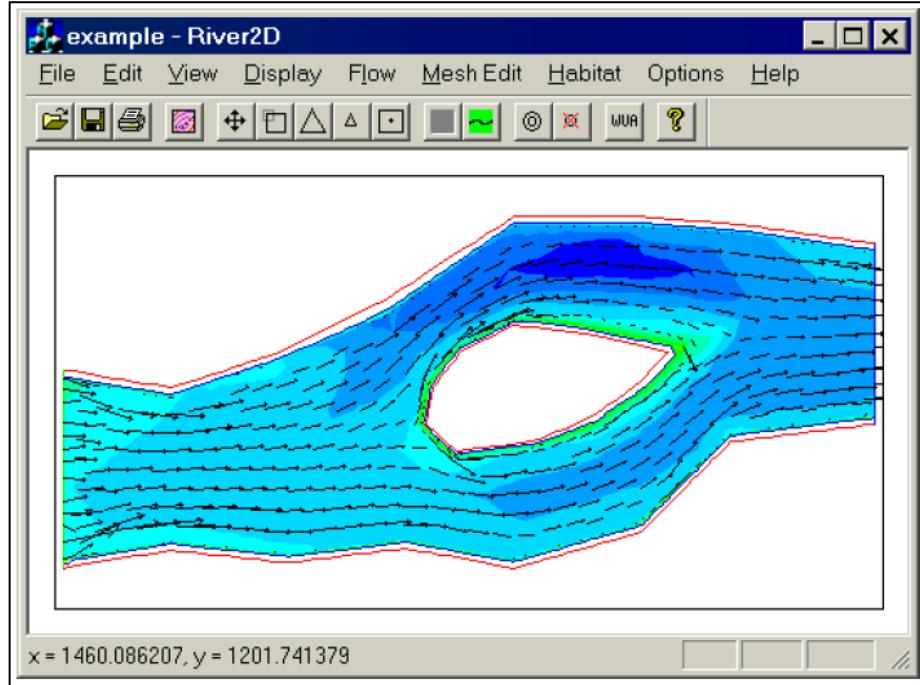
Oceans), Proteção Ambiental de Alberta (Alberta Environmental Protection) e Pesquisa Geológica dos Estados Unidos (United States Geological Survey).

Este programa computacional foi desenvolvido para simulações hidrodinâmicas bidimensionais específicas para córregos e rios. Foi feito com o esquema de Elementos Finitos, baseado no modelo de soluções conservativas de Petrov-Galerkin.

Um diferencial do River2D é a sua capacidade de modelar rios com áreas descontínuas e de diferentes espessuras de camadas de gelo, conseguindo, desta forma, uma simulação mais exata de rios situados em regiões sujeitas a temperaturas inferiores a zero graus Celsius.

Na Figura 7 é ilustrada a interface do programa computacional River2D.

Figura 7 - Interface do programa computacional River2D.



Fonte: Steffler (2002).

Pecini (2016) aplicou o modelo computacional River2D na simulação de escoamento fluvial em um estudo de caso do desvio de um rio para a construção de aproveitamentos hidrelétricos.

Kwan (2009) apresentou simulações hidrodinâmicas e modelo de transporte de sedimentos em diversos casos realizados com o auxílio do River2D.

2.4.4 HEC-RAS

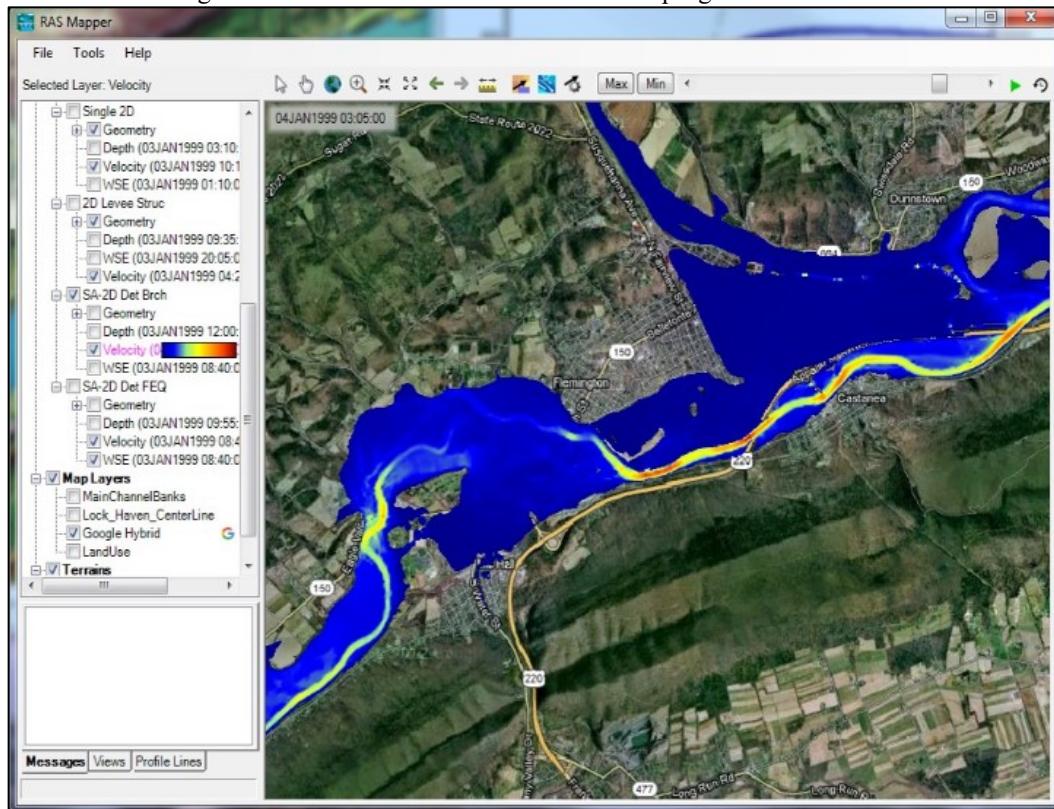
O HEC-RAS (*Hydrologic Engineering Center's – River Analysis System*) é um programa computacional gratuito, desenvolvido pelo Corpo de Engenheiros do Exército Americano (USACE).

Este programa computacional é capaz de realizar simulações de escoamentos permanentes em uma dimensão; escoamentos não-permanentes em uma e duas dimensões; cálculos de transporte de sedimentos com leito móvel e modelagens de qualidade e temperatura de água.

Para a solução das equações de Saint-Venant bidimensionais, o HEC-RAS utiliza a técnica de solução numérica de Crank-Nicolson, enquanto a discretização espacial é feita através de um esquema híbrido de diferenças finitas e volumes finitos.

É ilustrada, na Figura 8, a interface da saída de resultados do programa HEC-RAS.

Figura 8 - Interface da saída de resultados do programa HEC-RAS.



Fonte: USACE (2016?).

Yakti et al. (2018) simularam a inundação induzida pelo rompimento da barragem natural de *Way Ela* que ocorreu em 2013, situada na ilha Ambon - Indonésia, utilizando o HEC-RAS para obter os mapas de inundação, profundidade de água e tempo de chegada da inundação.

Estas informações são importantes para produzir planos de mitigação para possíveis rompimentos de barragens no futuro.

Vianini Neto (2016) utilizou o HEC-RAS para realizar simulações de ruptura da Barragem da Pampulha, que fica localizada no município de Belo Horizonte, alterando alguns parâmetros, como o coeficiente de rugosidade de Manning e tamanho das células do grid, e avaliou como as manchas de inundação se comportaram nestes diversos cenários hipotéticos de ruptura.

Vale lembrar que estes programas computacionais apresentados realizam as simulações hidrodinâmicas em duas dimensões na horizontal, desconsiderando as velocidades verticais e trabalhando com as velocidades de escoamento médias ao longo da altura. Porém, existem outros modelos de simulações hidrodinâmicas, como os modelos que trabalham com 1 dimensão (*DYRESM*), 2 dimensões na vertical, sendo elas as dimensões longitudinais e verticais (*CE-QUAL-W2*) e 3 dimensões (*Delft3D*).

2.5 Método numérico de Lax-Friedrichs para a solução de Equações Diferenciais Parciais

Este subitem descreve, de forma geral, o método numérico de Lax-Friedrichs segundo abordagem apresentada em Rezzola (2011). Observando a Equação 18 nota-se que se trata de uma Equação Diferencial Parcial (EDP).

$$\frac{\partial s}{\partial t} = -\frac{\partial F_x}{\partial x} - \frac{\partial F_y}{\partial y} \quad (18)$$

Esta equação pode ser resolvida pelo Método das Diferenças Finitas, como mostrado através da Equação 19.

$$s_{i,j}^{n+1} = s_{i,j}^n - \frac{\Delta t}{2\Delta x} [(F_x^n)_{i+1,j} - (F_x^n)_{i-1,j}] - \frac{\Delta t}{2\Delta y} [(F_y^n)_{i,j+1} - (F_y^n)_{i,j-1}] \quad (19)$$

Sendo, $s_{i,j}^{n+1}$ a variável a ser encontrada no nó i,j e tempo $n+1$; $s_{i,j}^n$ a variável já calculada no nó i,j no tempo n ; $(F_x^n)_{i+1,j}$ uma função dependente do nó vizinho $i+1,j$; $(F_x^n)_{i-1,j}$ uma função dependente do nó vizinho $i-1,j$; $(F_y^n)_{i,j+1}$ uma função dependente do nó vizinho $i,j+1$; $(F_y^n)_{i,j-1}$ uma função dependente do nó vizinho $i,j-1$.

Porém, ao se verificar a Equação 19 através da análise de estabilidade de Von Neumann, percebe-se que esta é instável, ou seja, os erros produzidos em um espaço de tempo do cálculo se acumulam à medida que os cálculos avançam. Para evitar problemas como estes, é

necessário o uso de métodos numéricos mais específicos, de maneira a reduzir os riscos de instabilidade da solução numérica.

Nesse sentido, o método de Lax Friedrichs é bastante utilizado em problemas que resolvem as equações de Saint Venant, em uma ou duas dimensões. O método introduz uma adaptação na Equação 19, ao se substituir o valor da variável do tempo n no nó i,j pelo valor da média das variáveis vizinhas, como a seguir:

$$s_{i,j}^n = \frac{1}{4} [s_{i+1,j}^n + s_{i-1,j}^n + s_{i,j+1}^n + s_{i,j-1}^n]$$

Obtém-se então a Equação 20, que é o Método Numérico de Lax-Friedrichs.

$$\begin{aligned} s_{i,j}^{n+1} = & \frac{1}{4} [s_{i+1,j}^n + s_{i-1,j}^n + s_{i,j+1}^n + s_{i,j-1}^n] - \frac{\Delta t}{2\Delta x} [(F_x^n)_{i+1,j} - (F_x^n)_{i-1,j}] \\ & - \frac{\Delta t}{2\Delta y} [(F_y^n)_{i,j+1} - (F_y^n)_{i,j-1}] \end{aligned} \quad (20)$$

O Método Numérico de Lax-Friedrichs é um método numérico explícito que calcula os dados do tempo posterior utilizando apenas os dados do tempo atual. Já os métodos numéricos implícitos utilizam os dados de ambos os tempos atual e posterior para se encontrar os dados do tempo posterior. A implementação de métodos explícitos em códigos computacionais é, em geral, mais simplificada do que a dos métodos implícitos. Por outro lado, os métodos implícitos são menos susceptíveis a instabilidades numéricas.

Neste item, citou-se apenas o método de Lax Friedrichs, pois este foi o único utilizado no desenvolvimento do trabalho. Porém existem vários outros métodos numéricos explícitos e implícitos.

CAPÍTULO 3

METODOLOGIA

Este capítulo apresenta aspectos mais específicos da metodologia utilizada para o desenvolvimento da ferramenta computacional SimHidro2D, construída neste trabalho. Esta ferramenta é baseada nas equações bidimensionais de Saint-Venant, onde, através da sua rotina de cálculo, é possível quantificar os valores de profundidade de água e velocidades de escoamentos das células analisadas na área de estudo ao longo do tempo.

3.1 Linguagem de programação e programas utilizados

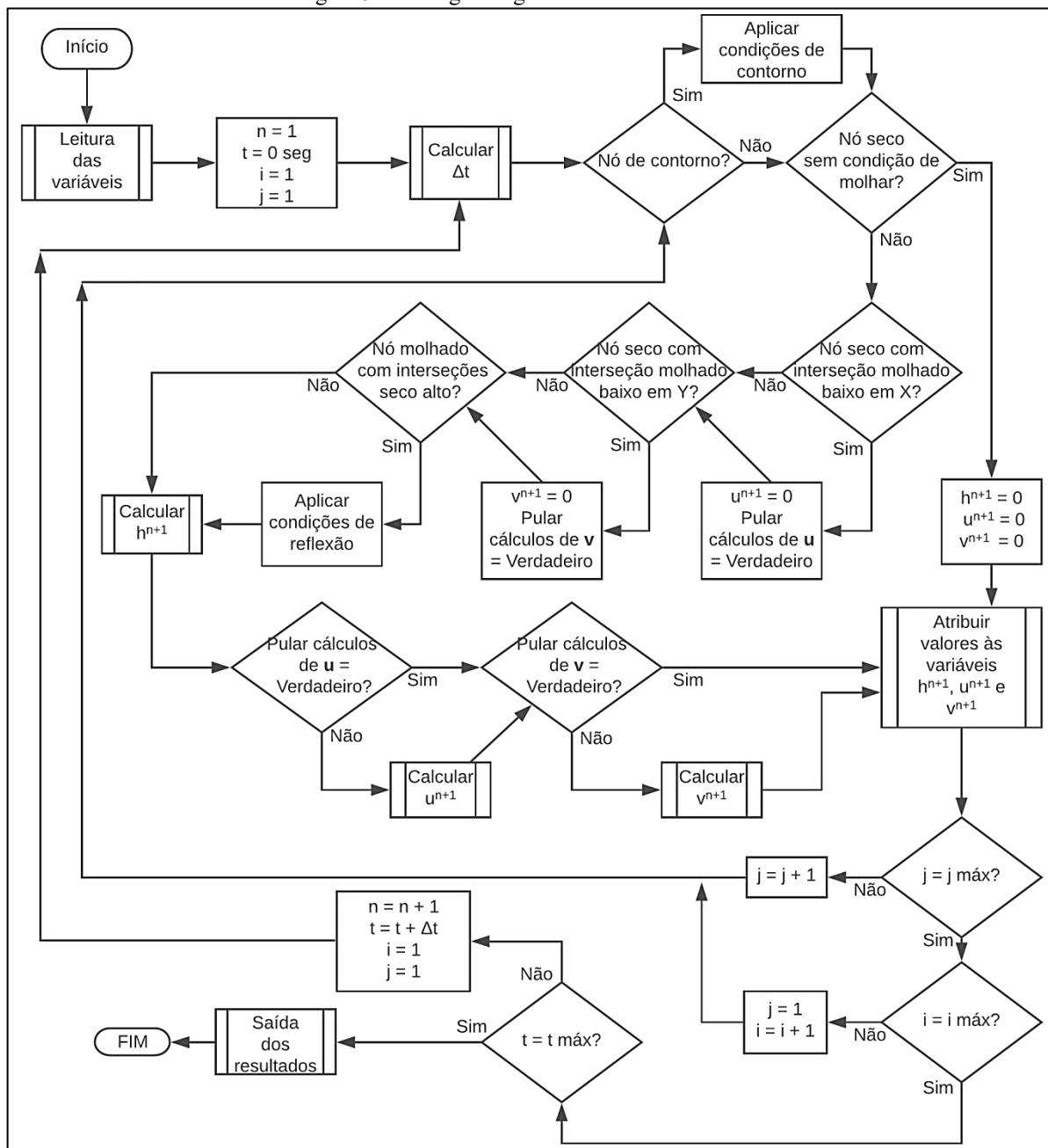
A linguagem de programação utilizada para a produção do SimHidro2D foi a *Visual Basic for Applications* (VBA), através do programa computacional *Microsoft Office Excel 2010*. Foi feita esta escolha devido ao autor já ter o domínio sobre esta linguagem e, também, pela facilidade de entrada e saída de dados através das planilhas do próprio programa. Este último aspecto pode constituir-se um facilitador aos eventuais usuários, que normalmente são profissionais mais afeitos ao ambiente do *Excel*. Destaca-se que os modelos computacionais citados no capítulo anterior – SisBahia, Mike 21, River 2D e HEC-RAS – apresentam certas peculiaridades que, embora não sejam limitadoras, exigem mais tempo de aprendizado por parte dos usuários.

Já para a comparação e validação dos dados produzidos, foi utilizado o programa computacional *HEC-RAS 5.0.5* (citado em 2.4.4). Foi utilizado este programa, pois, além de ser gratuito, o mesmo apresenta grande difusão no meio técnico e científico. Além disto, para preparar os dados dos mapas a serem importados para o *HEC-RAS 5.0.5* foi utilizado o programa computacional *QGIS 2.18*, o qual também é gratuito.

3.2 Fluxograma geral da rotina de cálculos

A rotina de cálculos feita durante as simulações está descrita no Apêndice A, porém, para o seu melhor entendimento, foi elaborado o fluxograma, ilustrado na Figura 9.

Figura 9 - Fluxograma geral da rotina de cálculos.



Fonte: Elaborado pelo autor.

Neste fluxograma destacam-se os seguintes símbolos: n o número do passo atual; i a numeração dos nós na direção do eixo X; j a numeração dos nós na direção do eixo Y; t o tempo atual da simulação, em s; $t_{\text{máx}}$ o tempo final da simulação previamente determinado, em s.

Como pode ser observada na Figura 15, a rotina de cálculos se inicia no instante de 0 segundo, na célula $i=1$ e $j=1$, então são feitas uma série de análises deste nó, antes de calcular as suas variáveis para, em seguida, calcular as variáveis do nó vizinho, isto se repete até que as variáveis de todos os nós deste instante sejam calculadas para, a partir daí, fazer os cálculos

para o instante seguinte, que é o $t + \Delta t$. Este processo se repete até que o último instante analisado seja o tempo final inicialmente estipulado, ou seja, até que $t = t_{máx}$.

3.3 Cálculo do incremento de tempo (Δt)

Os incrementos de tempo entre cada rotina de cálculo foram definidos seguindo a condição de Courant-Friedrichs-Lowy (CFL) para duas direções, descrita na Equação 21 e também comentada no subitem 2.2.

$$\max \left[\frac{u\Delta t}{\Delta x} + \frac{v\Delta t}{\Delta y} \right] \leq N_{CFL} \quad (21)$$

Isolando o incremento de tempo na Equação 21, tem-se então a Equação 22.

$$\frac{u\Delta t}{\Delta x} + \frac{v\Delta t}{\Delta y} \leq N_{CFL}$$

$$\frac{u\Delta t\Delta y + v\Delta t\Delta x}{\Delta x\Delta y} \leq N_{CFL}$$

$$\frac{\Delta t(u\Delta y + v\Delta x)}{\Delta x\Delta y} \leq N_{CFL}$$

$$\Delta t \leq \frac{\Delta x\Delta y N_{CFL}}{u\Delta y + v\Delta x} \quad (22)$$

Entretanto, utilizando a Equação 22, ocorrem problemas no início de simulações nas quais são feitas apenas a entrada de dados das profundidades de água dos nós. Desta maneira, todos os nós teriam velocidade nula, o que resultaria em um incremento de tempo muito grande, gerando resultados incoerentes. Para solucionar este problema, foi adicionado, nas velocidades para cálculo do incremento de tempo, um termo de celeridade de onda. Este termo, obtido a partir de Porto (2006), é descrito na Equação 23.

$$V_o = \sqrt{gh} \quad (23)$$

Sendo, V_o a velocidade da onda, em m/s; g a aceleração da gravidade, em m/s²; h a profundidade de água, em m.

Este termo nada mais é do que a velocidade de escoamento que a água atinge ao ser liberada, como por exemplo, a partir de uma abertura brusca de um vertedouro.

Adicionando então o termo de celeridade da Equação 23, na Equação 22, tem-se a equação final utilizada (Equação 24):

$$\Delta t \leq \frac{\Delta x \Delta y N_{CFL}}{(u + \sqrt{gh}) \Delta y + (v + \sqrt{gh}) \Delta x} \quad (24)$$

3.4 Resolução das equações governantes

As equações governantes utilizadas no estudo foram baseadas nas Equações 9, 10 e 11, descritas no subitem 2.1. Porém, não foi utilizada a parcela referente ao efeito de Coriolis devido a sua contribuição ser muito pequena quando comparado com os outros termos das equações. Além disto, a parcela de efeito de atrito de fundo foi substituída pelo termo descrito por Brunner (2016), na qual são utilizadas as Equações 25 e 26 para o cálculo do termo de atrito nas direções X e Y, respectivamente.

$$\frac{n^2 g u \sqrt{u^2 + v^2}}{R^{4/3}} \quad (25)$$

$$\frac{n^2 g v \sqrt{u^2 + v^2}}{R^{4/3}} \quad (26)$$

Com estas adaptações, as equações governantes podem ser descritas através das expressões 27, 28 e 29.

$$\frac{\partial h}{\partial t} + \frac{\partial(uh)}{\partial x} + \frac{\partial(vh)}{\partial y} = \gamma \left(\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2} \right) \quad (27)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = A \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - g \frac{\partial(h+Z)}{\partial x} - \frac{n^2 g u \sqrt{u^2 + v^2}}{R^{4/3}} \quad (28)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = A \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - g \frac{\partial(h+Z)}{\partial y} - \frac{n^2 g v \sqrt{u^2 + v^2}}{R^{4/3}} \quad (29)$$

Resolvendo estas equações utilizando o Método das Diferenças Finitas pelos esquemas de Diferenças Centradas e das Diferenças Progressivas, como citado no subitem 2.3.3, e utilizando uma malha deslocada para os cálculos das velocidades, têm-se então as matrizes descritas pela Equação 30.

$$\mathbf{P} - \mathbf{D} - \mathbf{E} - \mathbf{F} - \mathbf{K} - \mathbf{S} = 0 \quad (30)$$

Sendo,

$$\mathbf{P} = \begin{pmatrix} \frac{h_{i,j}^{n+1} - h_{i,j}^n}{\Delta t} \\ \frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}^n}{\Delta t} \\ \frac{v_{i,j+1/2}^{n+1} - v_{i,j+1/2}^n}{\Delta t} \end{pmatrix};$$

$$\mathbf{D} = \begin{pmatrix} \gamma \left(\frac{h_{i+1,j}^n - 2h_{i,j}^n + h_{i-1,j}^n}{\Delta x^2} + \frac{h_{i,j+1}^n - 2h_{i,j}^n + h_{i,j-1}^n}{\Delta y^2} \right) \\ A \left(\frac{u_{i+3/2,j}^n - 2u_{i+1/2,j}^n + u_{i-1/2,j}^n}{\Delta x^2} + \frac{u_{i+1/2,j+1}^n - 2u_{i+1/2,j}^n + u_{i+1/2,j-1}^n}{\Delta y^2} \right) \\ A \left(\frac{v_{i+1,j+1/2}^n - 2v_{i,j+1/2}^n + v_{i-1,j+1/2}^n}{\Delta x^2} + \frac{v_{i,j+3/2}^n - 2v_{i,j+1/2}^n + v_{i,j-1/2}^n}{\Delta y^2} \right) \end{pmatrix};$$

$$\mathbf{E} = \begin{pmatrix} -\frac{u_{i+1/2,j}^n h_{i+1/2,j}^n - u_{i-1/2,j}^n h_{i-1/2,j}^n}{\Delta x} \\ -\frac{u_{i+1/2,j}^n (u_{i+1,j}^n - u_{i,j}^n)}{\Delta x} \\ -\frac{u_{i,j+1/2}^n (v_{i+1/2,j+1/2}^n - v_{i-1/2,j+1/2}^n)}{\Delta x} \end{pmatrix};$$

$$\mathbf{F} = \begin{pmatrix} -\frac{v_{i,j+1/2}^n h_{i,j+1/2}^n - u_{i,j-1/2}^n h_{i,j-1/2}^n}{\Delta y} \\ -\frac{v_{i+1/2,k}^n (u_{i+1/2,j+1/2}^n - u_{i+1/2,j-1/2}^n)}{\Delta y} \\ -\frac{v_{i,j+1/2}^n (v_{i,j+1}^n - v_{i,j}^n)}{\Delta y} \end{pmatrix};$$

$$\mathbf{K} = \begin{pmatrix} 0 \\ -g \frac{h_{i+1,j}^n + Z_{i+1,j}^n - h_{i,j}^n - Z_{i,j}^n}{\Delta x} \\ -g \frac{h_{i,j+1}^n + Z_{i,j+1}^n - h_{i,j}^n - Z_{i,j}^n}{\Delta y} \end{pmatrix};$$

$$\mathbf{S} = \begin{pmatrix} 0 \\ -\frac{n^2 \times g \times u_{i+1/2,j}^n \sqrt{(u_{i+1/2,j}^n)^2 + (v_{i+1/2,j}^n)^2}}{R^{4/3}} \\ -\frac{n^2 \times g \times v_{i,j+1/2}^n \sqrt{(u_{i,j+1/2}^n)^2 + (v_{i,j+1/2}^n)^2}}{R^{4/3}} \end{pmatrix}$$

Sendo A , um termo difusivo definido pela Equação 31, em que a constante, D , é adotada segundo a Tabela 1.

$$A = \frac{Dhn\sqrt{g}\sqrt{u^2+v^2}}{R^{1/6}} \quad (31)$$

Tabela 1 - Valores de coeficiente para cálculo da viscosidade turbulenta.

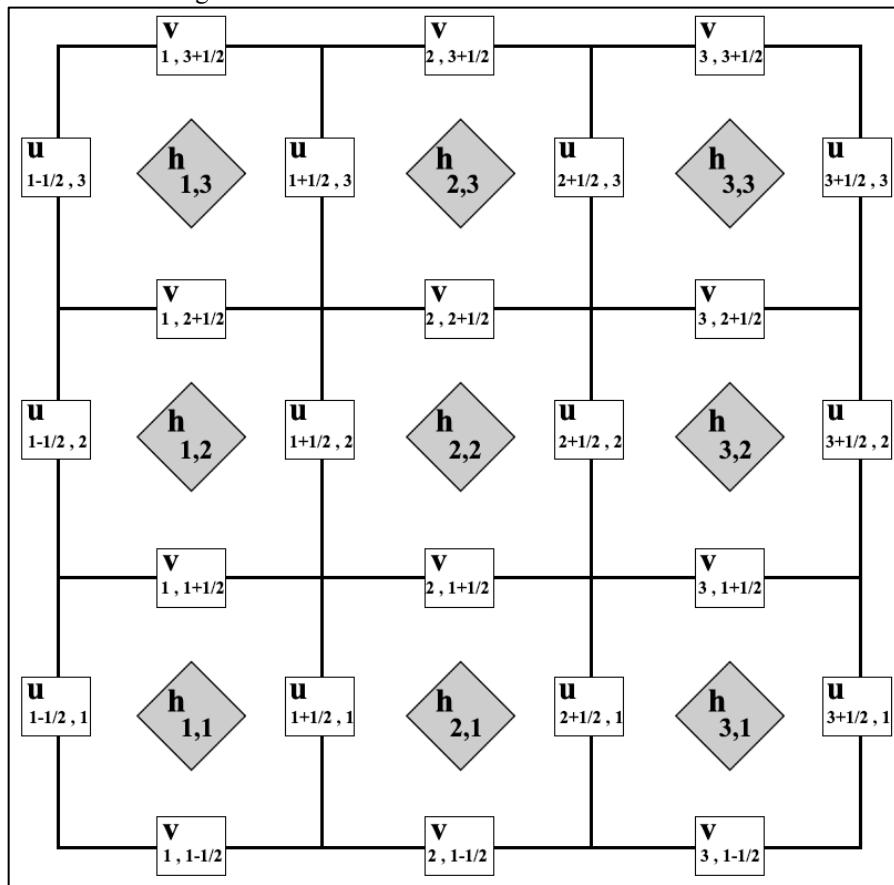
D	Intensidade da mistura	Geometria e superfície
0.11 até 0.26	Pouca mistura	Canal reto, superfície lisa
0.30 até 0.77	Mistura moderada	Meandros suaves, superfície com irregularidades moderadas
2.0 até 5.0	Mistura intensa	Meandros bruscos, superfície áspera

Fonte: Brunner (2016), tradução nossa.

Sendo, D uma constante empírica para cálculo da viscosidade turbulenta; n o coeficiente de rugosidade de Manning; R o raio hidráulico, em m.

É ilustrada, na Figura 10, a disposição das variáveis calculadas de profundidades de água e velocidades de escoamento ao longo da área de estudo, simulando um domínio de 3 células ao longo do eixo X por 3 células ao longo do eixo Y.

Figura 10 - Posicionamento das variáveis calculadas.



Fonte: Elaborado pelo autor.

Observa-se que, para uma grade de 3x3 células, haverá uma grade de variáveis h de 3x3, enquanto haverá uma grade de 4x3 para as variáveis u e, por fim, uma grade de 3x4 para as variáveis v .

Vale ressaltar que, resolvendo o sistema de matrizes da Equação 30, serão calculadas as variáveis h , u e v , nas posições i,j , $i+1/2,j$ e $i,j+1/2$, respectivamente, por ciclo de cálculo. Tem-se, neste caso, o arranjo das variáveis em malha deslocada (*staggered grid*), onde os escalares, como a profundidade, são armazenados no centro das células e os vetores, como as componentes de velocidade, são armazenados nas faces das células.

3.5 Condições de contorno

Para todas as situações, é necessário que se tenha uma configuração adequada para os limites da área de estudo para que a simulação não dê resultados incoerentes nestas regiões.

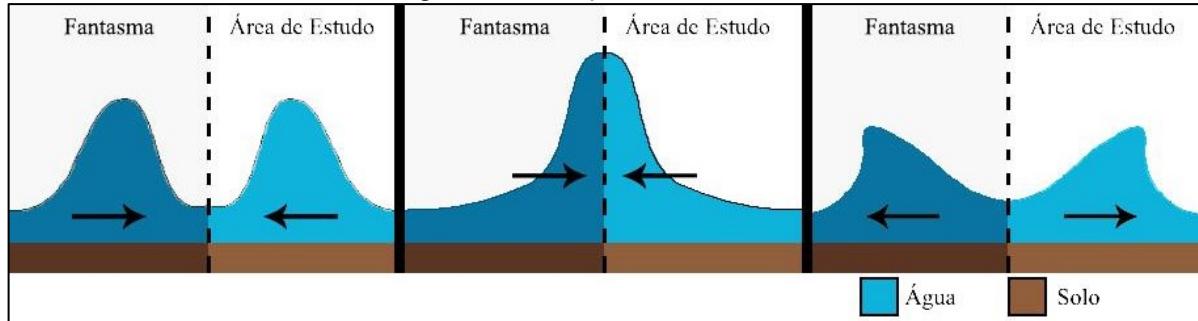
No presente estudo, foram utilizados dois modelos de condições de contorno, sendo um com uma configuração de total reflexão, e o outro sem reflexão. Estas condições serviram para

modelar o efeito de paredes sólidas ou de interfaces que permitem livre fluxo da água, conforme descrevem os subitens seguintes.

3.5.1 Condição de contorno: Total reflexão

Esta configuração de total reflexão é utilizada quando se deseja que nenhum volume de fluido saia da área de estudo, sendo necessário então, que todo volume que atingir os limites da área seja rebatido e volte para dentro do corpo de água. Este comportamento é como se realmente existissem paredes delimitando a área de estudo, conforme está ilustrado na Figura 11.

Figura 11 - Condição de total reflexão.



Fonte: Elaborado pelo autor.

Para que aconteça esta total reflexão, são anuladas todas as velocidades na parede e são simulados nós fantasma, ou seja, nós que não existem no interior do domínio analisado. Estes nós são necessários para os cálculos nas extremidades da área de estudo. Nos nós fantasma, são copiados os valores de algumas das variáveis dos nós imediatamente próximos no interior da área de estudo, conforme os critérios do Quadro 1. Em vermelho estão descritas as variáveis dos nós fantasma.

Quadro 1 – Configurações da condição de contorno com reflexão total.

Parede i=1	Parede i=i máx	Parede j=1	Parede j=j máx
$h_{i-1/2,j} = h_{i,j}$	$u_{i+1/2,j} = 0$	$h_{i,j-1/2} = h_{i,j}$	$h_{i,j+1} = h_{i,j}$
$h_{i-1,j} = h_{i,j}$	$h_{i+1/2,j} = h_{i,j}$	$h_{i,j-1} = h_{i,j}$	$v_{i,j+1/2} = 0$
$u_{i-1/2,j} = 0$	$h_{i+1,j} = h_{i,j}$	$u_{i+1/2,j-1/2} = 0$	$h_{i,j+1/2} = h_{i,j}$
$v_{i-1/2,j+1/2} = 0$	$v_{i+1,j+1/2} = -v_{i,j+1/2}$	$u_{i+1/2,j-1} = -u_{i+1/2,j}$	$u_{i+1/2,j+1} = -u_{i+1/2,j}$
$v_{i-1,j+1/2} = -v_{i,j+1/2}$	$v_{i+1/2,j+1/2} = 0$	$v_{i,j-1/2} = 0$	$u_{i+1/2,j+1/2} = 0$
$z_{i-1,j} = z_{i,j}$	$z_{i+1,j} = z_{i,j}$	$z_{i,j-1} = z_{i,j}$	$z_{i,j+1} = z_{i,j}$
$z_{i-1,j+1} = z_{i,j+1}$	$u_{i+3/2,j} = -u_{i-1/2,j}$	$z_{i+1,j-1/2} = z_{i+1,j}$	$v_{i,j+3/2} = -v_{i,j+1/2}$
	$u_{i+1,j} = -(u_{i-1/2,j})/2$		$u_{i,j+1/2} = 0$
	$v_{i+1/2,j} = 0$		$v_{i,j+1} = -(v_{i,j+1/2})/2$
	<i>Omitir cálculos de u</i>		$z_{i+1,j+1} = z_{i+1,j}$
			<i>Omitir cálculos de v</i>

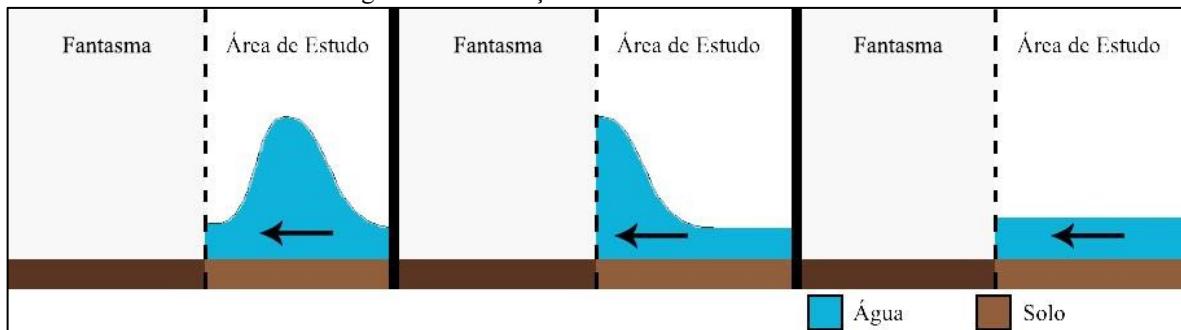
Fonte: Elaborado pelo autor.

Esta condição de contorno foi utilizada nos testes realizados quando se desejava que nenhum volume saísse da área de estudo.

3.5.2 Condição de contorno: Sem reflexão

A condição de contorno sem reflexão é aplicada quando se deseja que os volumes que entrarem em contato com os limites da área de estudo continuem com o seu fluxo, como se não existissem estes limites, como ilustrado na Figura 12.

Figura 12 – Condição de contorno sem reflexão.



Fonte: Elaborado pelo autor.

Para esta configuração foi adotado que todo o fluxo que tocasse as paredes dos limites saísse da área de estudo. Para que isto ocorresse, foi determinado que todos os nós fantasma tivessem as configurações descritas conforme o Quadro 2. As variáveis dos nós fantasma estão escritas em vermelho.

Quadro 2 - Configurações da condição de contorno sem reflexão.

Parede i=1	Parede i=i máx	Parede j=1	Parede j=j máx
$h_{i-1/2,j} = h_{i,j}$	$h_{i+1/2,j} = h_{i,j}$	$h_{i,j-1/2} = h_{i,j}$	$h_{i,j+1} = h_{i,j}$
$h_{i-1,j} = h_{i,j}$	$h_{i+1,j} = h_{i,j}$	$h_{i,j-1} = h_{i,j}$	$v_{i,j+1/2} = v_{i,j-1/2}$
$u_{i-1/2,j} = u_{i+1/2,j}$	$u_{i+1/2,j} = u_{i-1/2,j}$	$u_{i+1/2,j-1/2} = u_{i+1/2,j}$	$h_{i,j+1/2} = h_{i,j}$
$v_{i-1/2,j+1/2} = v_{i,j+1/2}$	$v_{i+1,j+1/2} = v_{i,j+1/2}$	$u_{i+1/2,j-1} = u_{i+1/2,j}$	$u_{i+1/2,j+1} = u_{i+1/2,j}$
$v_{i-1,j+1/2} = v_{i,j+1/2}$	$v_{i+1/2,j+1/2} = v_{i,j+1/2}$	$v_{i,j-1/2} = v_{i,j+1/2}$	$u_{i+1/2,j+1/2} = u_{i+1/2,j}$
$z_{i-1,j} = z_{i,j}$	$z_{i+1,j} = z_{i,j}$	$z_{i,j-1} = z_{i,j}$	$z_{i,j+1} = z_{i,j}$
$z_{i-1,j+1} = z_{i,j+1}$	$u_{i+3/2,j} = u_{i-1/2,j}$	$z_{i+1,j-1} = z_{i+1,j}$	$v_{i,j+3/2} = v_{i,j-1/2}$
	$u_{i+1,j} = u_{i-1/2,j}$		$u_{i,j+1/2} = u_{i,j}$
	$v_{i+1/2,j} = v_{i,j}$		$v_{i,j+1} = (v_{i,j-1/2})/2$
	<i>Omitir cálculos de u</i>		$z_{i+1,j+1} = z_{i+1,j}$
			<i>Omitir cálculos de v</i>

Fonte: Elaborado pelo autor.

Esta condição de contorno foi utilizada durante as simulações, pois se desejava que todo volume que entrasse em contato com as fronteiras da área de estudo fossem eliminados, já que a região de importância das simulações se encontrava longe destes limites.

3.6 Condições internas

As condições internas são necessárias para que o modelo simule corretamente as interfaces onde existem interações de terreno-água. Isto ocorre quando, por exemplo, um volume de água se choque com uma elevação no solo, como uma ilha, ou em qual momento um nó seco receberá água, fato que dependerá da profundidade de água dos nós vizinhos.

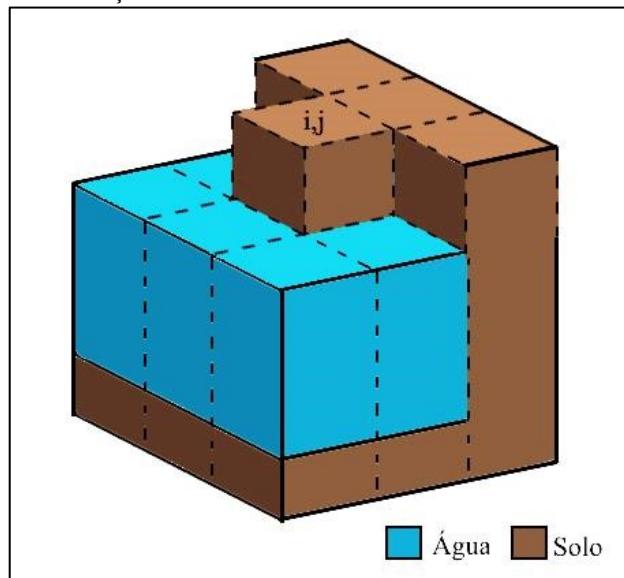
São então analisadas três situações: quando um nó seco é vizinho apenas de nós secos ou nós molhados baixos; quando um nó seco é vizinho de nós molhados altos e nós secos ou nós molhados baixos; quando um nó molhado é vizinho de nós secos altos.

Quando for citado nó vizinho molhado baixo, deve-se entender que a altitude da superfície da água do nó vizinho é inferior à altitude do solo do nó analisado, do mesmo modo, quando for citado nó vizinho seco alto, deve-se entender que a altitude do solo do nó vizinho é maior que a altitude da superfície da água do nó analisado.

3.6.1 *Nó seco vizinho de nós secos ou de nós molhados baixos*

O nó seco vizinho de apenas outros nós secos ou de nós molhados baixos não tem condições de receber volume de água em um ciclo de cálculo, já que a condição CFL garante que seja necessário mais de um ciclo para que um volume de água percorra um nó inteiro. Deste modo, não há necessidade de calcular as variáveis para este nó seco que não tem condições de se tornar molhado, como pode ser visto no nó i,j na Figura 13.

Figura 13 - Condição de nó seco vizinho de nós secos ou molhados baixos.



Fonte: Elaborado pelo autor.

Então, para diminuir os processos de cálculo, quando se é identificado que um nó seco é vizinho de outros nós secos ou nós molhados baixos, não é realizado, em tal caso, o processo de cálculo das variáveis para este nó. Este roteiro é feito como descrito a seguir.

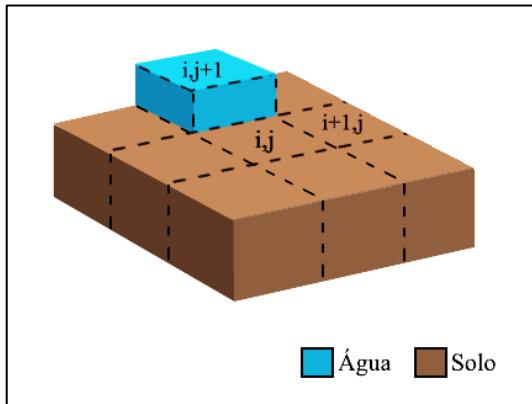
$$\begin{aligned}
 & Se \left(\begin{array}{l} (h_{i,j}^n = 0) \text{ e } ((z_{i,j}^n \geq z_{i-1,j}^n + h_{i-1,j}^n) \text{ ou } (h_{i-1,j}^n = 0)) \text{ e } \\ ((z_{i,j}^n \geq z_{i+1,j}^n + h_{i+1,j}^n) \text{ ou } (h_{i+1,j}^n = 0)) \text{ e } \\ ((z_{i,j}^n \geq z_{i,j-1}^n + h_{i,j-1}^n) \text{ ou } (h_{i,j-1}^n = 0)) \text{ e } \\ ((z_{i,j}^n \geq z_{i,j+1}^n + h_{i,j+1}^n) \text{ ou } (h_{i,j+1}^n = 0)) \end{array} \right) \\
 & \text{então } \left\{ \begin{array}{l} h_{i,j}^{n+1} = 0 \\ u_{i+1/2,j}^{n+1} = 0 \\ v_{i,j+1/2}^{n+1} = 0 \\ \text{Omitir cálculos} \end{array} \right.
 \end{aligned}$$

3.6.2 Nô seco vizinho de nós molhados altos e nós secos ou nós molhados baixos

Quando um nó seco é vizinho de outros nós secos ou de nós molhados baixos e de pelo menos um nó molhado alto, então é cogitado que este nó tenha condição de se tornar molhado no próximo passo de tempo. Entretanto, se este nó vizinho molhado alto não for vizinho de uma das faces nas quais são calculadas as velocidades, então pode ser desconsiderado o cálculo da velocidade nesta face.

Para uma melhor compreensão, é ilustrado na Figura 14, o nó i,j , no qual o nó $i,j+1$ se trata de um nó vizinho molhado alto, enquanto o nó $i+1,j$ se trata de um nó vizinho seco. Neste caso, será desconsiderado o cálculo da velocidade u entre os nós i,j e $i+1,j$, já que não haverá nenhum fluxo entre estes dois nós. Porém, é necessário realizar o cálculo da velocidade v entre os nós i,j e $i,j+1$ e também o cálculo da profundidade h para o nó i,j , pois ele poderá receber volume de água do nó $i,j+1$.

Figura 14 – Nô seco vizinho de nô molhado alto e nô secos.



Fonte: Elaborado pelo autor.

Logo, para analisar e eventualmente desconsiderar os cálculos nas faces que não terão fluxo de fluido, é realizado o roteiro conforme descrito a seguir.

$$\text{Se } (h_{i,j} = 0) \text{ e } (h_{i+1,j} = 0) \text{ ou } (h_{i+1,j} + z_{i+1,j} \leq z_{i,j})$$

$$\text{então } \begin{cases} u_{i+1/2,j} = 0 \\ \text{Omitir cálculos de } u_{i+1/2,j} \end{cases}$$

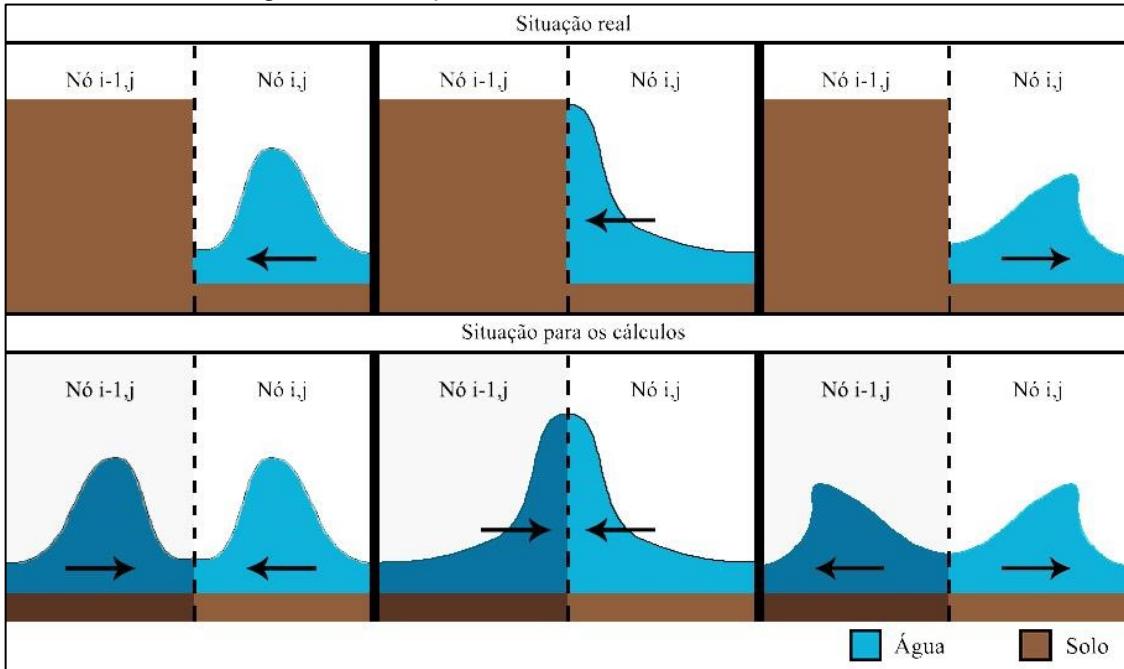
$$\text{Se } (h_{i,j} = 0) \text{ e } (h_{i,j+1} = 0) \text{ ou } (h_{i,j+1} + z_{i,j+1} \leq z_{i,j})$$

$$\text{então } \begin{cases} v_{i,j+1/2} = 0 \\ \text{Omitir cálculos de } v_{i,j+1/2} \end{cases}$$

3.6.3 Nô molhado vizinho de nô secos altos

Quando um nô molhado é vizinho de um nô seco alto, como exemplo uma ilha, é necessário que a água que esteja indo de encontro com este solo seja rebatida. Para que os cálculos simulem este rebatimento, durante os cálculos, além de determinar que a velocidade nesta face seja nula, é simulado um volume fictício idêntico de água vindo de encontro com o volume a ser rebatido, como exemplificado na Figura 15, onde se destaca o contato entre os nós i,j e $i-1,j$.

Figura 15 - Condição de nó molhado vizinho de nó seco alto.



Fonte: Elaborado pelo autor.

Esta simulação deste volume fictício vindo de encontro com o volume a ser rebatido é feita alterando momentaneamente as variáveis dos nós vizinhos. Estas alterações estão descritas a seguir, sendo reproduzidas, em vermelho, as variáveis dos nós fantasmas que são alteradas momentaneamente.

$$\left\{
 \begin{array}{l}
 u_{i+1/2,j}^n = 0 \\
 h_{i+1/2,j}^n = h_{i,j}^n \\
 h_{i+1,j}^n = h_{i,j}^n \\
 v_{i+1,j+1/2}^n = -v_{i,j+1/2}^n \\
 v_{i+1/2,j+1/2}^n = 0 \\
 u_{i+3/2,j}^n = -u_{i-1/2,j}^n \\
 u_{i+1,j}^n = -(u_{i-1/2,j}^n)/2 \\
 v_{i+1/2,j}^n = 0
 \end{array}
 \right.$$

Omitir cálculos de u

$$\left\{
 \begin{array}{l}
 h_{i-1/2,j}^n = h_{i,j}^n \\
 h_{i-1,j}^n = h_{i,j}^n \\
 u_{i-1/2,j}^n = 0 \\
 v_{i-1/2,j+1/2}^n = 0 \\
 v_{i-1,j+1/2}^n = -v_{i,j+1/2}^n
 \end{array}
 \right.$$

Se $(h_{i,j+1}^n = 0 \text{ e } z_{i,j+1} \geq h_{i,j}^n + z_{i,j})$ então:

$$\left\{ \begin{array}{l} h_{i,j+1}^n = h_{i,j}^n \\ v_{i,j+1/2}^n = 0 \\ h_{i,j+1/2}^n = h_{i,j}^n \\ u_{i+1/2,j+1}^n = -u_{i+1/2,j}^n \\ u_{i+1/2,j+1/2}^n = 0 \\ v_{i,j+3/2}^n = -v_{i,j-1/2}^n \\ u_{i,j+1/2}^n = 0 \\ v_{i,j+1}^n = -(v_{i,j-1/2}^n)/2 \\ \text{Omitir cálculos de } v \end{array} \right.$$

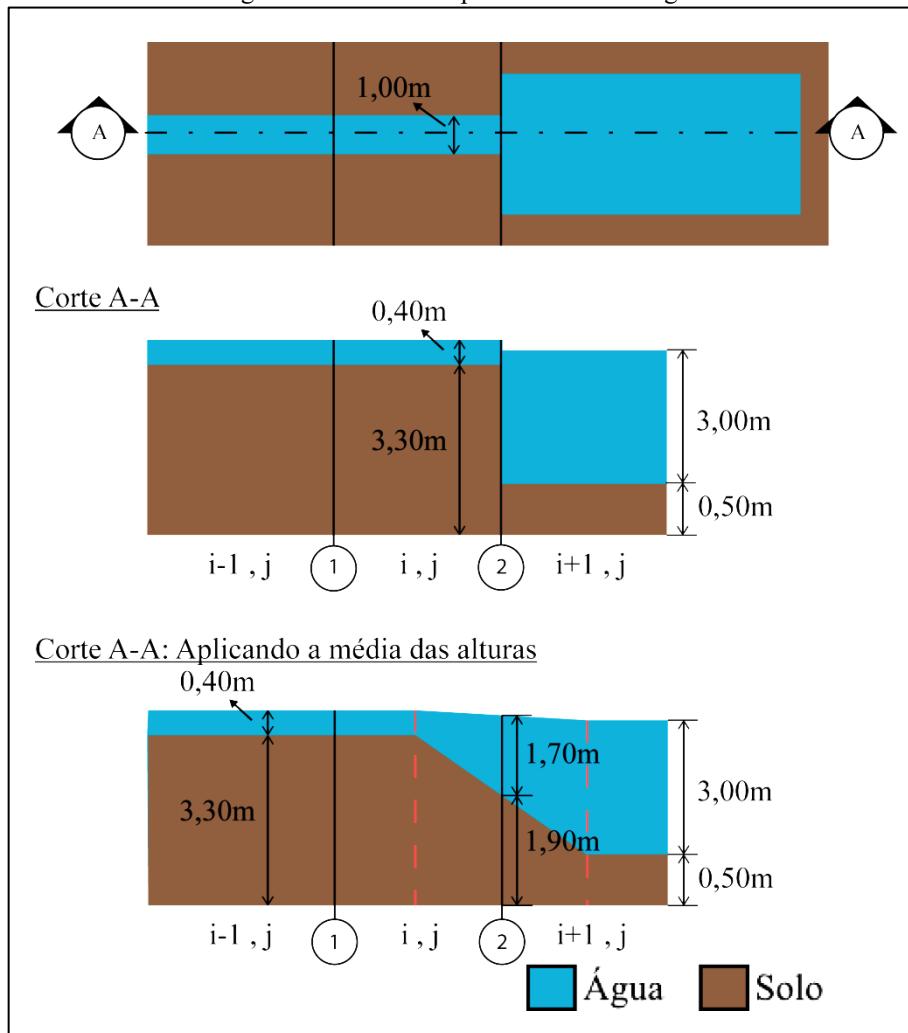
Se $(h_{i,j-1}^n = 0 \text{ e } z_{i,j-1} \geq h_{i,j}^n + z_{i,j})$ então:

$$\left\{ \begin{array}{l} h_{i,j-1/2}^n = h_{i,j}^n \\ h_{i,j-1}^n = h_{i,j}^n \\ u_{i+1/2,j-1/2}^n = 0 \\ u_{i+1/2,j-1}^n = -u_{i+1/2,j}^n \\ v_{i,j-1/2}^n = 0 \end{array} \right.$$

3.7 Esquema de cálculo para a profundidade de água e altitude do solo nas faces das células

O cálculo de h dos nós é feito através da Equação 30, a qual utiliza as velocidades e profundidades de água das faces do nó analisado. Entretanto, as profundidades de água só são conhecidas nos centros de cada nó. Uma possível solução seria utilizar a média da profundidade de água dos centros dos nós vizinhos e, também, a média da altitude dos solos dos nós vizinhos, como ilustrado na Figura 16.

Figura 16 - Média das profundidades de água.



Fonte: Elaborado pelo autor.

O exemplo hipotético aqui ilustrado é de um rio que corre no sentido do nó $i-1,j$ para o nó i,j , que tem 1 metro de largura, 40 centímetros de profundidade, velocidade de 1m/s e que desagua em um lago de 3 metros de profundidade.

Com estes dados, têm-se então os cálculos para as vazões das faces 1 e 2, em que é feita a média das profundidades de água e das altitudes dos solos dos centros das células vizinhas.

$$\text{Face 1} - Q = V \times A = 1 \times 0,4 \times 1 = 0,4 \text{ m}^3/\text{s}$$

$$\text{Face 2} - Q = V \times A = 1 \times 1,7 \times 1 = 1,7 \text{ m}^3/\text{s}$$

$$\Delta Vol_{i,j} = \text{Face 1} - \text{Face 2} = 0,4 - 1,7 = -1,3 \text{ m}^3/\text{s}$$

Pelos resultados, pode-se notar que, utilizando a média das profundidades, a vazão aumenta bruscamente gerando resultados incoerentes, em alguns casos, chegando a resultar em uma profundidade de água negativa.

Para solucionar este problema, foram utilizadas as configurações do Quadro 3, para se obter a profundidade da água e altitude do solo nas faces dos nós. Neste caso, supõe-se que, para o valor positivo da velocidade, a água se desloca da esquerda para a direita.

Quadro 3 - Obtenção da profundidade de água e altitude do solo nas faces das células.

Se velocidade > 0	Utilizar valores de profundidade de água e altitude do solo do centro da célula da esquerda.
Se velocidade < 0	Utilizar valores de profundidade de água e altitude do solo do centro da célula da direita.
Se velocidade = 0	Utilizar valores de profundidade de água e altitude do solo do centro da célula que tiver a maior altitude da superfície de água.

Fonte: Elaborado pelo autor.

Refazendo os cálculos da situação da Figura 16, utilizando este novo esquema, tem-se os seguintes resultados:

$$\text{Face 1} - Q = V \times A = 1 \times 0,4 \times 1 = 0,4 \text{ m}^3/\text{s}$$

$$\text{Face 2} - Q = V \times A = 1 \times 0,4 \times 1 = 0,4 \text{ m}^3/\text{s}$$

$$\Delta Vol_{i,j} = \text{Face 1} - \text{Face 2} = 0,4 - 0,4 = \mathbf{0} \text{ m}^3/\text{s}$$

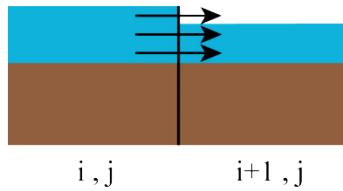
Este simples exemplo numérico mostra que este artifício preserva melhor a continuidade dos volumes quando comparado com o método que utiliza a média das profundidades.

3.8 Esquema de cálculo de profundidades relativas no cálculo de h

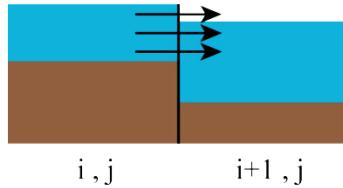
O método utilizado no subitem 3.7 é válido somente quando a altitude do solo, de onde o volume de água está saindo, é maior ou igual à altitude do solo na qual a água está chegando, como ilustrado nas situações *a*) e *b*) da Figura 17.

Figura 17 - Uso da profundidade relativa para o cálculo de h .

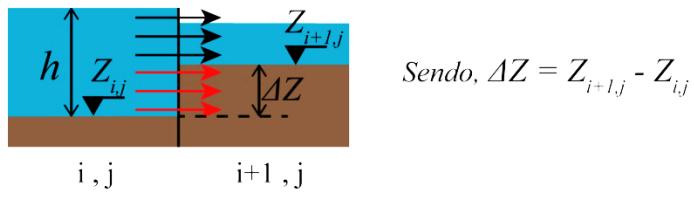
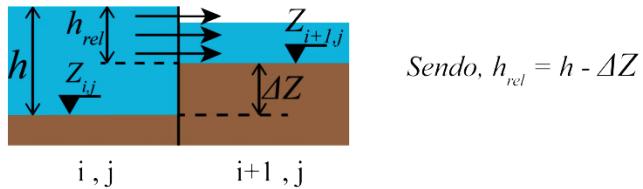
a) Altitude do solo da célula de origem igual à da célula de destino



b) Altitude do solo da célula de origem maior que a da célula de destino



c) Altitude do solo da célula de origem menor que a da célula de destino

d) Altitude do solo da célula de origem menor que a da célula de destino, utilizando alturas relativas para o cálculo de h 

Fonte: Elaborado pelo autor.

Entretanto, quando a altitude da célula de origem é menor que a altitude da célula de destino, tem-se uma parede de altura ΔZ que impede o fluxo total, como mostrado na situação c) da Figura 17. Para que os cálculos sejam mais fiéis à realidade, é então utilizada durante os cálculos de h , a profundidade relativa de água (h_{rel}), que é a profundidade de água da célula de origem (h) subtraído a altura do obstáculo (ΔZ).

3.9 Esquema de cálculo de profundidades relativas no termo de suavização de h

O termo de suavização de h , descrito por Zhendong, Junqiang e Zhenghui (2015), foi utilizado devido à dissipação causada por este termo nos erros residuais presentes em cada iteração conferindo, desta forma, uma maior estabilidade à solução numérica. Este termo é descrito na Equação 32.

$$\gamma \left(\frac{h_{i+1,j}^n - 2h_{i,j}^n + h_{i-1,j}^n}{\Delta x^2} + \frac{h_{i,j+1}^n - 2h_{i,j}^n + h_{i,j-1}^n}{\Delta y^2} \right) \quad (32)$$

Observando este termo de suavização, pode-se notar que ele utiliza diferenças de profundidades entre o nó analisado e os nós vizinhos, como mostrado na Equação 33.

$$\gamma \left(\frac{h_{i+1,j}^n - 2h_{i,j}^n + h_{i-1,j}^n}{\Delta x^2} + \frac{h_{i,j+1}^n - 2h_{i,j}^n + h_{i,j-1}^n}{\Delta y^2} \right) = \gamma \left(\frac{\Delta h_{i+1,j}^n + \Delta h_{i-1,j}^n}{\Delta x^2} + \frac{\Delta h_{i,j+1}^n + \Delta h_{i,j-1}^n}{\Delta y^2} \right) \quad (33)$$

Sendo, $\Delta h_{posx,posy}^n = h_{posx,posy}^n - h_{i,j}^n$, em m; $posx$ a numeração do nó ao longo do eixo X e $posy$ a numeração do nó ao longo do eixo Y.

Logo, foi utilizado um esquema de profundidades relativas semelhante ao utilizado no cálculo de h , explicado no subitem 3.8, feito conforme o Quadro 4, e aplicado para os 4 nós vizinhos.

Quadro 4 - Esquema de profundidades relativas para o termo de suavização de h .

Se $h_{i+1,j}^n > 0$	$dfh1 = h_{i+1,j}^n + z_{i+1,j}^n - h_{i,j}^n - z_{i,j}^n$
Senão	$dfh1 = 0$

Fonte: Elaborado pelo autor.

Aplicando, então, este esquema de profundidades relativas no termo de suavização de h , tem-se então o termo final utilizado descrito pela Equação 34.

$$D(\mathbf{1}) = \gamma \left(\frac{dfh1 + dfh2}{\Delta x^2} + \frac{dfh3 + dfh4}{\Delta y^2} \right) \quad (34)$$

Sendo, $dfh1$ a diferença de cotas altimétricas entre as superfícies líquidas das células $i+1,j$ e i,j , em m; $dfh2$ a diferença de cotas altimétricas entre as superfícies líquidas das células $i-1,j$ e i,j , em m; $dfh3$ a diferença de cotas altimétricas entre as superfícies líquidas das células $i,j+1$ e i,j , em m; $dfh4$ a diferença de cotas altimétricas entre as superfícies líquidas das células $i,j-1$ e i,j , em m.

O coeficiente γ varia entre os valores 0 a 1, sendo que, quanto mais próximo do valor 1, maior é a suavização causada entre as diferenças de profundidades das células e quanto mais próximo do valor 0, menor é a suavização.

3.10 Esquema de cálculo dos raios hidráulicos

O cálculo do raio hidráulico é realizado conforme a Equação 35.

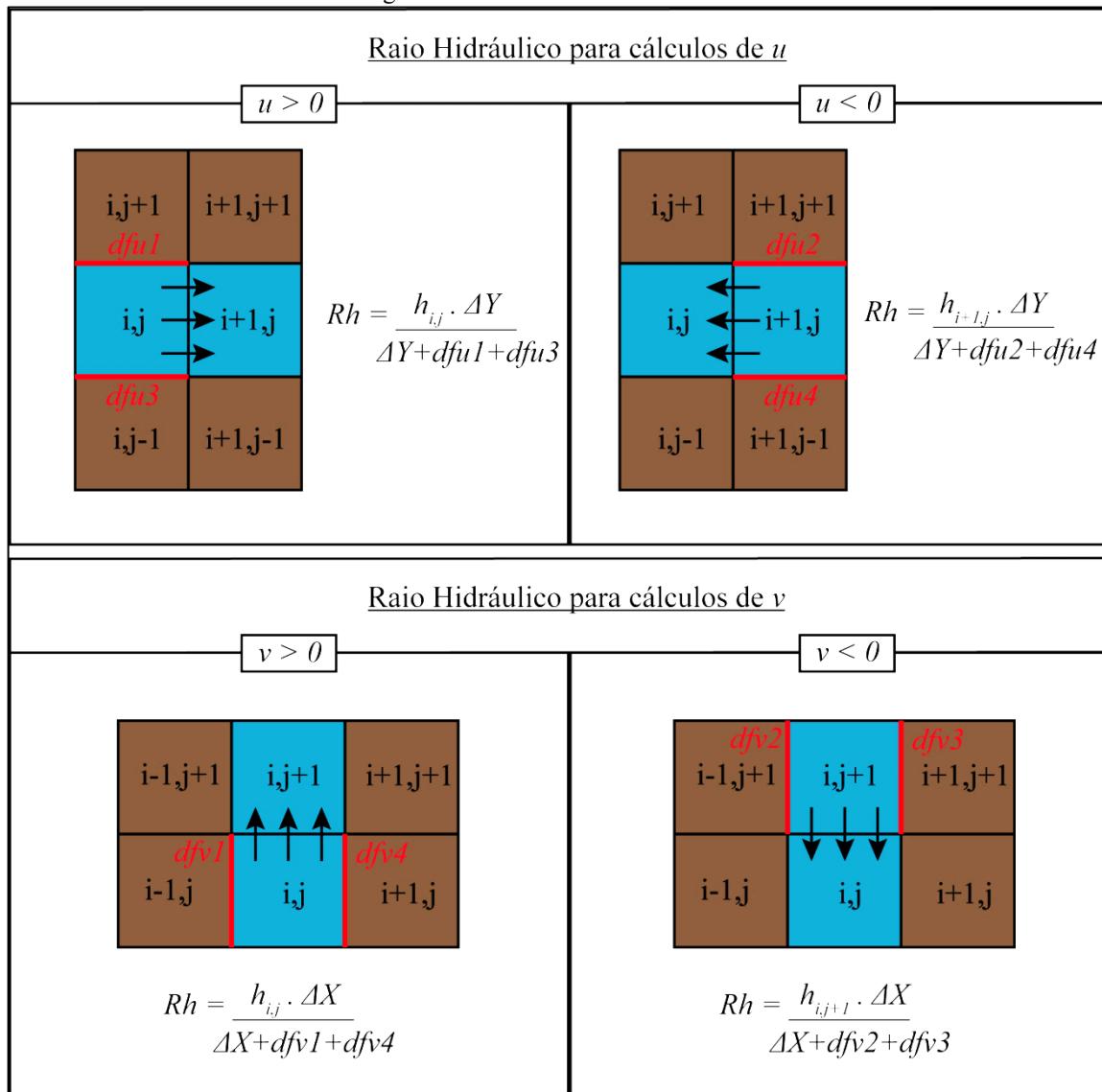
$$R_h = \frac{A_m}{P_m} \quad (35)$$

Sendo, A_m a área molhada, em m^2 e P_m o perímetro molhado, em m.

Entretanto, como as velocidades são calculadas nas faces das células, tem-se então o mesmo problema descrito no subitem 3.7, já que os dados de profundidade de água de altitude do solo são conhecidos somente no centro das células.

Neste caso, foi adotado o mesmo esquema descrito no item 3.7, no qual são utilizados os dados de profundidade de água e altitude do solo do centro da célula de origem do escoamento. Desta forma, o cálculo do raio hidráulico é feito conforme ilustrado na Figura 18.

Figura 18 - Cálculo de raio hidráulico.



Fonte: Elaborado pelo autor.

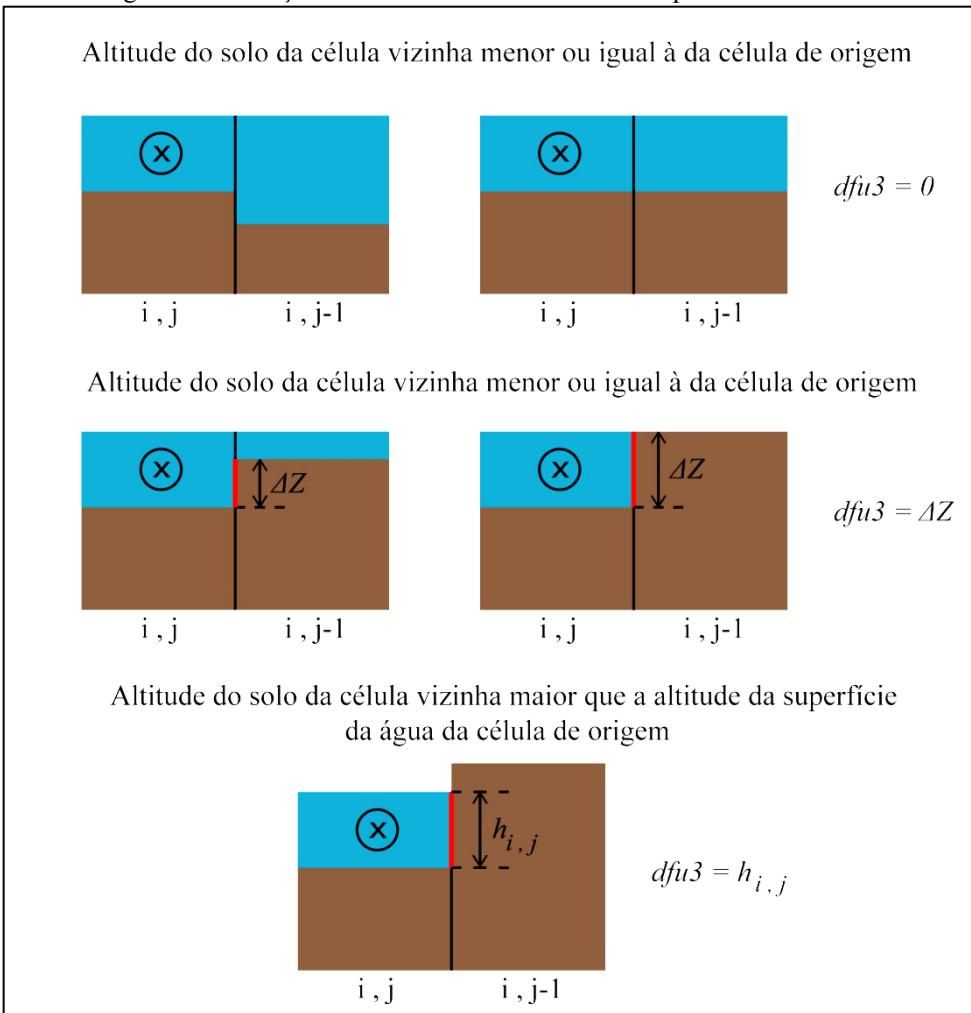
Sendo, $dfu1$ o comprimento de parede em contato com a água na face entre os nós i,j e $i,j+1$, em m; $dfu2$ o comprimento de parede em contato com a água na face entre os nós $i+1,j$ e $i+1,j+1$, em m; $dfu3$ o comprimento de parede em contato com a água na face entre os nós i,j e $i,j-1$, em m; $dfu4$ o comprimento de parede em contato com a água na face entre os nós $i+1,j$ e $i+1,j-1$, em m; $dfv1$ o comprimento de parede em contato com a água na face entre os nós i,j e $i-1,j$, em m; $dfv2$ o comprimento de parede em contato com a água na face entre os nós $i,j+1$ e $i-1,j+1$, em m; $dfv3$ o comprimento de parede em contato com a água na face entre os nós $i,j+1$ e $i+1,j+1$, em m e $dfv4$ o comprimento de parede em contato com a água na face entre os nós i,j e $i+1,j$, em m.

Para o cálculo do perímetro molhado existem três situações de relevo, que são:

- Altitude do solo da célula vizinha menor ou igual à da célula de origem;
- Altitude do solo da célula vizinha menor ou igual à profundidade de água da célula de origem;
- Altitude do solo da célula vizinha maior do que a altitude da superfície da água da célula de origem.

Estas três configurações são ilustradas na Figura 19.

Figura 19 - Situações de relevos durante o cálculo de perímetro molhado.



Fonte: Elaborado pelo autor.

São calculados os perímetros molhados de acordo com cada situação apresentada para, posteriormente, calcular o raio hidráulico de cada célula para cada direção de velocidade.

3.11 Esquema de cálculo de profundidades relativas e alterações no Método de Lax-Friedrichs

Aplicando o Método de Lax-Friedrichs, descrito no subitem 2.5.1, no cálculo de h , as equações governantes ficam conforme as matrizes da Equação 36. Neste caso, as mudanças ocorreram apenas na matriz P e estão marcadas em vermelho.

$$\mathbf{P} - \mathbf{D} - \mathbf{E} - \mathbf{F} - \mathbf{K} - \mathbf{S} = 0 \quad (36)$$

Sendo,

$$\mathbf{P} = \begin{pmatrix} \frac{h_{i,j}^{n+1} - \frac{1}{4}[h_{i+1,j}^n + h_{i-1,j}^n + h_{i,j+1}^n + h_{i,j-1}^n]}{\Delta t} \\ \frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}^n}{\Delta t} \\ \frac{v_{i,j+1/2}^{n+1} - v_{i,j+1/2}^n}{\Delta t} \end{pmatrix}$$

Entretanto, utilizando a equação desta forma não é levada em conta a diferença entre as altitudes dos solos dos nós analisados. Para resolver isto, foi utilizado um esquema de profundidades relativas semelhante ao descrito no subitem 3.8. Além disto, foi também adicionado a profundidade de água do nó da célula nesta média de profundidades, para que o cálculo da profundidade de água do nó analisado sofresse menos influência dos nós vizinhos. Após estas alterações, as equações governantes ficam conforme as matrizes da Equação 37, onde as mudanças ocorreram apenas na matriz \mathbf{P} e estão indicadas em vermelho.

$$\mathbf{P} - \mathbf{D} - \mathbf{E} - \mathbf{F} - \mathbf{K} - \mathbf{S} = 0 \quad (37)$$

Sendo,

$$\mathbf{P} = \begin{pmatrix} \frac{h_{i,j}^{n+1} - \frac{1}{(TT+4)}[hT1 + hT2 + hT3 + hT4 + 4h_{i,j}^n]}{\Delta t} \\ \frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}^n}{\Delta t} \\ \frac{v_{i,j+1/2}^{n+1} - v_{i,j+1/2}^n}{\Delta t} \end{pmatrix}$$

Em que,

$$\text{Se } h_{i+1,j}^n > 0 \text{ e } h_{i+1,j}^n + Z_{i+1,j} > Z_{i,j} \quad \begin{cases} hT1 = h_{i+1,j}^n + Z_{i+1,j} - Z_{i,j} \\ TT = TT + 1 \end{cases}$$

$$\text{Senão } \{hT1 = 0$$

$$\text{Se } h_{i-1,j}^n > 0 \text{ e } h_{i-1,j}^n + Z_{i-1,j} > Z_{i,j} \quad \begin{cases} hT2 = h_{i-1,j}^n + Z_{i-1,j} - Z_{i,j} \\ TT = TT + 1 \end{cases}$$

$$\text{Senão } \{hT2 = 0$$

$$\text{Se } h_{i,j+1}^n > 0 \text{ e } h_{i,j+1}^n + Z_{i,j+1} > Z_{i,j} \quad \begin{cases} hT3 = h_{i,j+1}^n + Z_{i,j+1} - Z_{i,j} \\ TT = TT + 1 \end{cases}$$

$$\text{Senão } \{hT3 = 0$$

$$\text{Se } h_{i,j-1}^n > 0 \text{ e } h_{i,j-1}^n + Z_{i,j-1} > Z_{i,j} \quad \begin{cases} hT4 = h_{i,j-1}^n + Z_{i,j-1} - Z_{i,j} \\ TT = TT + 1 \end{cases}$$

Senão $\{hT4 = 0$

CAPÍTULO 4

R E S U L T A D O S E D I S C U S S Õ E S

Este capítulo apresenta os resultados obtidos neste trabalho, a partir da aplicação da metodologia apresentada no capítulo anterior, no desenvolvimento da ferramenta SimHidro2D.

No início deste capítulo são apresentadas as interfaces do SimHidro2D, além de três testes realizados para avaliar a rotina de cálculo embutida no programa.

Ao final do capítulo, são apresentados os resultados gerados pelas simulações do Reservatório Bom Jardim, realizadas nas ferramentas computacionais SimHidro2D e HEC-RAS, e em seguida, são comparados os resultados obtidos entre essas duas ferramentas. Espera-se que após a leitura deste capítulo o leitor compreenda todo o funcionamento e as possíveis aplicações da rotina computacional criada no presente trabalho.

A título de informação, todos os testes e simulações citadas neste Capítulo foram calculados no computador do autor, o qual tem sistema operacional *Windows 10*, processador *Intel Core i5-4200U* e *8GB de memória RAM*.

4.1 Interfaces de configuração, entrada de dados e saída dos resultados do SimHidro2D

Como descrito no subitem 3.1, foi utilizado o programa computacional *Microsoft Office Excel 2010* para desenvolver a rotina de cálculos, implementada em linguagem VBA.

O programa desenvolvido se baseia principalmente em três abas, sendo a primeira para inserir as configurações desejadas para a simulação; a segunda para a entrada de dados de relevo e profundidades e velocidades de água e, a terceira, para a saída dos resultados.

Conforme as prerrogativas iniciais, a interface do programa é simples, contendo somente o necessário para a configuração e entrada de dados, como se pode notar na Figura 20.

Figura 20 - Aba de configuração da versão de testes do SimHidro2D.

Fonte: Elaborado pelo autor.

A planilha apresentada na Figura 20 trata-se da versão de testes do SimHidro2D, onde foram configuradas e calculadas as simulações piloto a serem apresentadas no subitem 4.2. A versão final do SimHidro2D, utilizada para simular o Reservatório Bom Jardim, e apresentada no subitem 4.3, foi simplificada, pois não era mais necessária a realização de testes, como se pode observar na Figura 21.

Figura 21 - Aba de configuração do SimHidro2D.

SIMULAÇÃO - REPRESA BOM JARDIM																		
1	A	B	C	D	E	F	G	H	I	J	K							
2	Números de células na direção X: 141						X mín [m]: 786700											
3	Números de células na direção Y: 440						X máx [m]: 788100											
4	ΔX [m]: 10						Y mín [m] : 7892100											
5	ΔY [m]: 10						t inicial [s] : 0											
6							t final [s] : 10801											
7																		
8	CFL: 0.1																	
9	D adotado [Eddy Viscosity]: 0.6						Coeficiente de Manning : 0.04											
10																		
11	Gamma [suavização de h]: 0.9						Configurações das paredes de contorno											
12																		
13																		
14																		
15																		
16																		
17																		
18																		
19																		
20																		
21																		
22																		
23																		
24																		
25																		
26																		
27																		
28																		
29																		
30																		
31																		
32																		
33																		
34																		
35																		
36																		
37																		
38																		
39																		
40																		
41																		
42																		
43																		
44																		
45																		
46																		
47																		
48																		
49																		
50																		
51																		
52																		
53																		
54																		
55																		
56																		
57																		
58																		
59																		
60																		
61																		
62																		
63																		
64																		
65																		
66																		
67																		
68																		
69																		
70																		
71																		
72																		
73																		
74																		
75																		
76																		
77																		
78																		
79																		
80																		
81																		
82																		
83																		
84																		
85																		
86																		
87																		
88																		
89																		
90																		
91																		
92																		
93																		
94																		
95																		
96																		
97																		
98																		
99																		
100																		
101																		
102																		
103																		
104																		

Fonte: Elaborado pelo autor.

A entrada dos valores de geometria, profundidades de água e velocidades de escoamento é feita através da aba *Entrada*, na qual são inseridos os valores de X , Y , Z , h , u , v e W de toda a área de estudo. Estes valores inseridos correspondem aos dados da área de estudo no instante do início da simulação. A aba *Entrada* é ilustrada na Figura 22.

Figura 22 - Aba de entrada de dados do SimHidro2D.

	A	B	C	D	E	F	G	H	I	J	K	L
1	X [m]	Y [m]	Z [m]	h [m]	u [m/s]	v [m/s]	W (Z+h) [m]		Legenda			
2	786700	7892100	811.51	0	0	0	811.51		Z = Altitude do solo			
3	786700	7892110	811.46	0	0	0	811.46		h = Altura da água			
4	786700	7892120	811.41	0	0	0	811.41		u = Velocidade na direção X			
5	786700	7892130	811.34	0	0	0	811.34		v = Velocidade na direção Y			
6	786700	7892140	811.27	0	0	0	811.27		W = Altitude da superfície da água			
7	786700	7892150	811.19	0	0	0	811.19					
8	786700	7892160	811.11	0	0	0	811.11					
9	786700	7892170	811.02	0	0	0	811.02					
10	786700	7892180	810.93	0	0	0	810.93					
11	786700	7892190	810.83	0	0	0	810.83					
12	786700	7892200	810.73	0	0	0	810.73					
13	786700	7892210	810.63	0	0	0	810.63					
14	786700	7892220	810.53	0	0	0	810.53					
15	786700	7892230	810.42	0	0	0	810.42					
16	786700	7892240	810.32	0	0	0	810.32					

Fonte: Elaborado pelo autor.

Os dados a serem inseridos na aba *Entrada* podem ser importados a partir de modelos digitais de terrenos e/ou levantamentos realizados em campo, sendo que o ideal seria importar estes dados a partir de batimetrias realizadas na área de estudo.

Na aba *Saída*, está contido o botão para início da rotina de cálculos. Nesta planilha também se pode acompanhar o processo dos cálculos, através dos valores de *Passo*, *t* e Δt , como ilustrado na Figura 23. Este valor de *Passo* equivale à quantidade de instantes de tempo que já foram calculados na simulação em andamento. Finalmente, após o término dos cálculos, os resultados dos campos hidrodinâmicos são dispostos na planilha.

Figura 23 - Aba de saída de resultados do SimHidro2D.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	X [m]	Y [m]	Z [m]	hf [m]	uf [m/s]	vf [m/s]	Wf (Z+h) [m]	Vel [m/s]					
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													

Fonte: Elaborado pelo autor.

4.2 Testes preliminares com o código

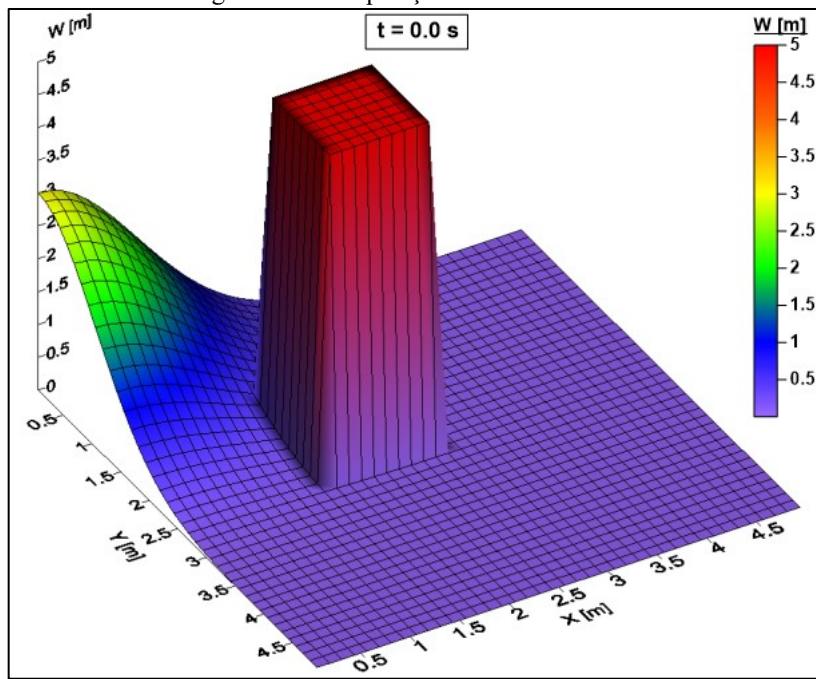
Para verificar a confiança dos resultados gerados pelo algoritmo elaborado, foram realizados testes mais simples, de processamento mais expedito. Nesta etapa o intuito era observar o comportamento do código, se o programa era capaz de gerar resultados satisfatórios. Somente após isto partiria-se para simulações de reservatório, onde naturalmente demanda-se grande volume de cálculos.

Ao longo deste item, serão apresentados apenas os testes que foram mais importantes durante a fase de verificação do programa.

4.2.1 Teste 1 - Obstáculo

Este teste foi realizado para observar como um volume de água, propagando-se na forma de onda, iria reagir ao encontrar um obstáculo alto, como por exemplo, os pilares de uma ponte. Para isto, foi posicionado um volume de água em um dos cantos da área de estudo e elevada a altitude do solo de uma região próxima do centro, para simular um obstáculo. Esta composição inicial está ilustrada na Figura 24.

Figura 24 - Composição inicial do Teste 1.



Fonte: Elaborado pelo autor.

Neste teste foram utilizadas as configurações descritas na Tabela 2.

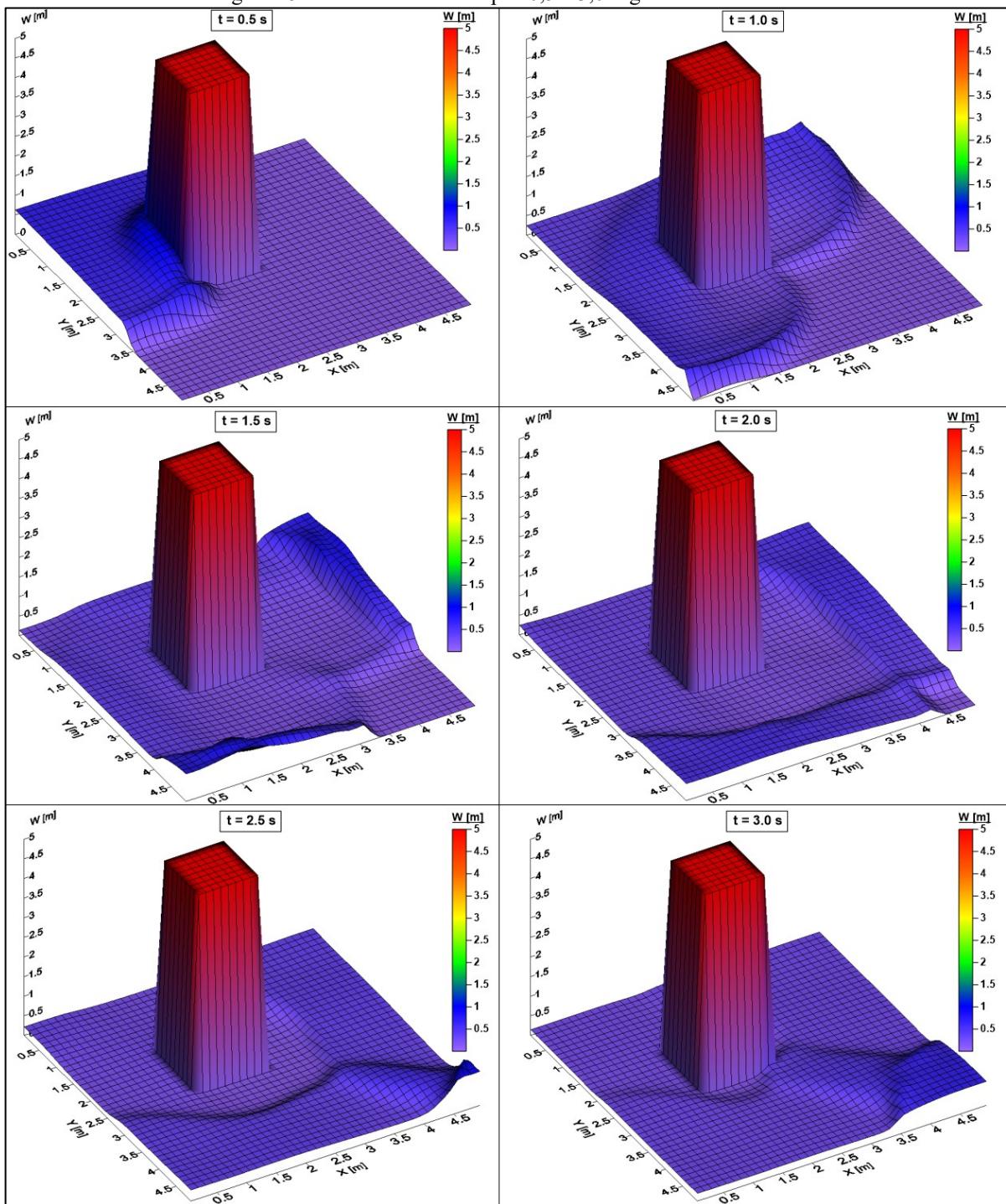
Tabela 2 - Configurações para o teste 1.

Nós em X	40
Nós em Y	40
Total de nós	1600
Tempo final [s]	5
Coef. de Manning	0,013
Número de CFL	0,4
D	0,6
γ	0,6
Condições de contorno	
Pare de X = 0	Total reflexão
Pare de X = Xmáx	Total reflexão
Pare de Y = 0	Total reflexão
Pare de Y = Ymáx	Total reflexão

Fonte: Elaborado pelo autor.

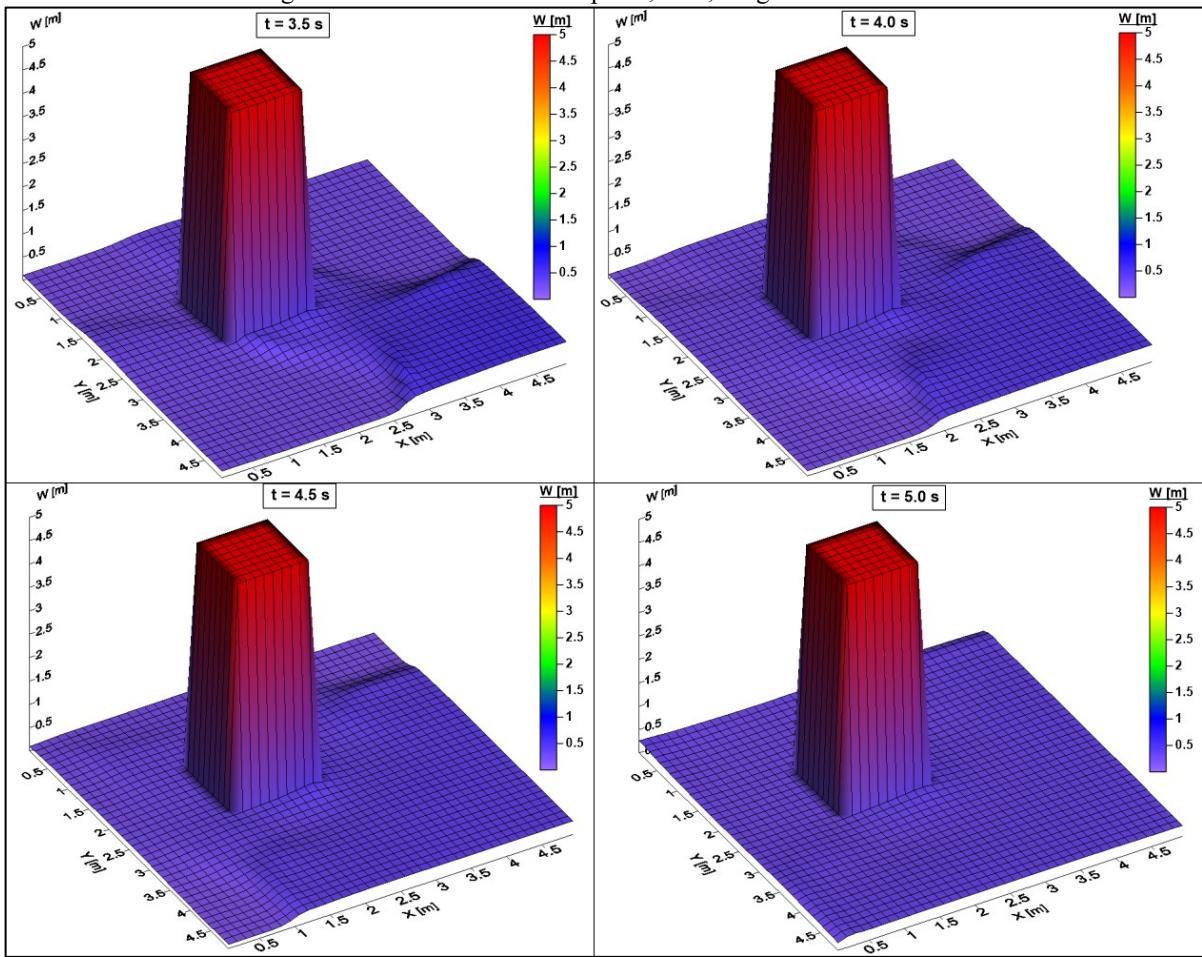
Os resultados do teste 1 estão ilustrados nas Figuras 25 e 26.

Figura 25 - Resultados nos tempos 0,5 a 3,0 segundos do Teste 1.



Fonte: Elaborado pelo autor.

Figura 26 - Resultados nos tempos 3,5 a 5,0 segundos do Teste 1



Fonte: Elaborado pelo autor.

Nestas figuras, o termo W , indicado na legenda, refere-se à soma da profundidade do escoamento com a altitude do solo, em metros.

Este teste foi processado em torno de 46 segundos, no computador com as configurações citadas no início deste capítulo.

Como informado na Tabela 2, utilizaram-se condições de contorno de total reflexão nos quatro limites, para que o volume de água não escapasse da área de estudo.

Partindo da Figura 24 e observando os resultados nos tempos 0,5 e 1 segundo, é possível notar que a condição interna de total reflexão para solos altos funcionou corretamente, já que uma porção do volume é refletida. O restante do volume de água, o qual não foi refletido pelo obstáculo, continua o seu caminho até ser rebatido nas paredes do limite da área de estudo durante o tempo entre 1,5 e 3,0 segundos, o que comprova o funcionamento da condição de contorno de total reflexão. Em suma, nesta simulação, foram reproduzidos padrões de

escoamento esperados, como a colisão da onda com o obstáculo e sua dissociação em duas partes que contornam o pilar. A jusante do obstáculo, formam-se ondas menores que tornam a colidir com as paredes que delimitam externamente o domínio. Após estas colisões, as massas de água são refletidas para o interior do domínio e se fundem em uma única onda com altura ainda menor que torna a atacar o pilar. Nota-se, ao longo dos diversos instantes, que há dissipação das perturbações na zona de estudo. Este efeito dissipativo decorre principalmente da introdução de uma viscosidade turbulenta no modelo matemático expresso pelas equações bidimensionais de Saint Venant.

Portanto, utilizando o teste 1, foram verificadas as condições de total reflexão nos contornos internos e externos do domínio.

Para uma melhor compreensão e visualização dos resultados gerados neste teste, foi disponibilizado no *YouTube* um vídeo com os resultados deste teste durante os tempos de 0 a 10 segundos, o qual pode ser visualizado através do seguinte link:
<https://www.youtube.com/watch?v=uPENgn2WYVQ>

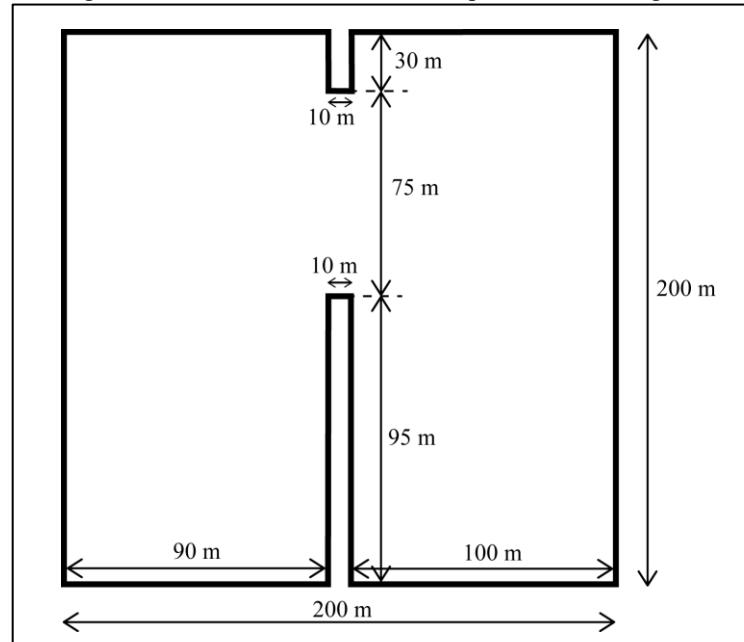
4.2.2 Teste 2 – Rompimento de barragem

O teste 2 trata-se da simulação do rompimento instantâneo de um trecho de uma barragem.

Este teste é utilizado por vários pesquisadores (CANNATA e MARZOCCHI, 2012; CHAUDHRY, 2008; ERDURAN, 2013; LI e FAN, 2017; ZHENDONG e JUNQIANG, 2015), que trabalham com as equações bidimensionais de Saint-Venant, para analisar a confiança das suas simulações, pois as grandes descontinuidades iniciais geram resultados incoerentes em alguns esquemas numéricos utilizados.

A configuração inicial deste teste consiste de uma área de estudo de 200 metros de largura por 200 metros de comprimento; uma barragem de 10 metros de largura no meio da área de estudo; uma abertura na parede da barragem não simétrica de 75 metros de largura que se situa a 30 metros do limite da área de estudo; um lado da barragem com 10 metros de profundidade de água e o outro lado com 5 metros de profundidade de água. Os resultados são analisados 7,2 segundos após o início da simulação. A geometria deste teste está ilustrada na Figura 27.

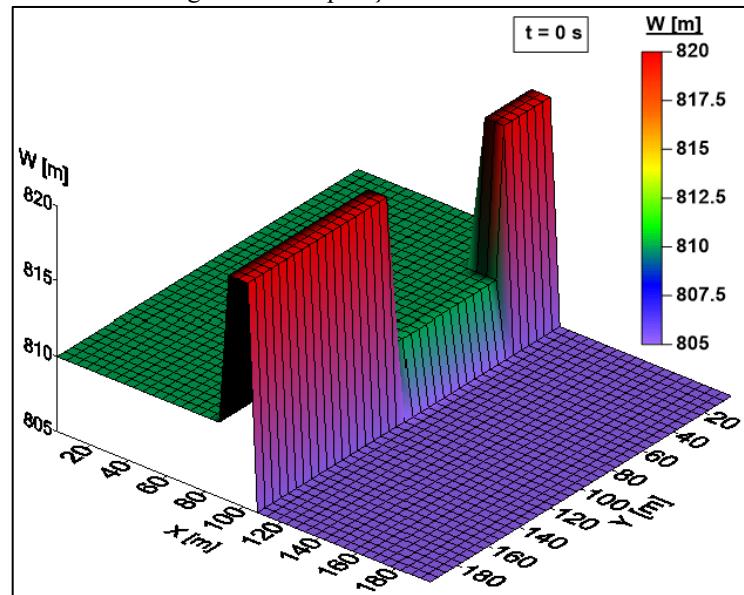
Figura 27 - Geometria do teste de rompimento de barragem.



Fonte: Elaborado pelo autor.

A disposição inicial deste teste está ilustrada pela Figura 28.

Figura 28 - Disposição inicial do teste 2.



Fonte: Elaborado pelo autor.

Assim como na simulação anterior, o termo W , cuja magnitude é indicada na legenda, equivale à soma da profundidade de água com a altitude do solo.

Neste teste foram utilizadas as configurações descritas na Tabela 3.

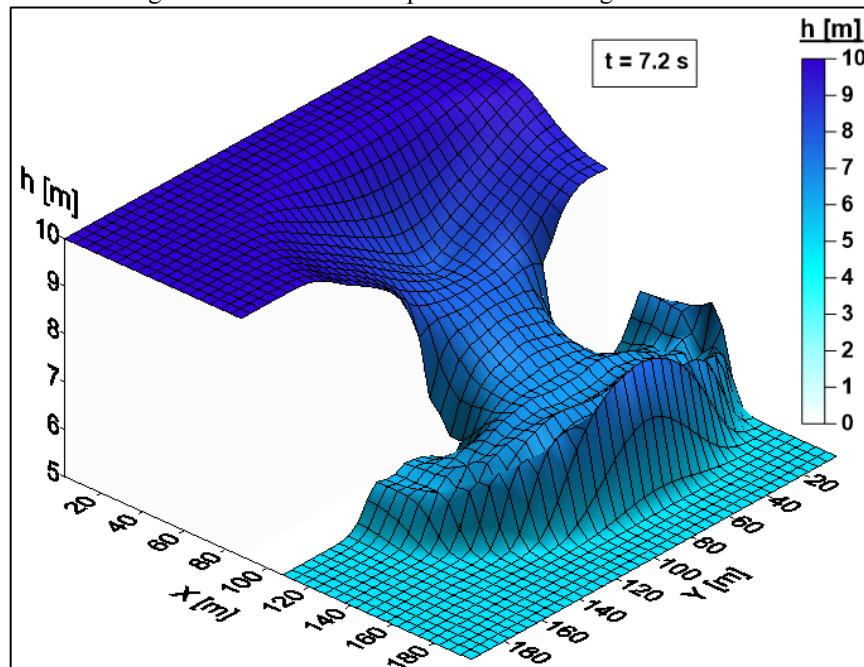
Tabela 3 - Configurações para o teste 2.

Nós em X	40
Nós em Y	40
Total de nós	1600
Tempo final [s]	7,2
Coef. de Manning	0,013
Número de CFL	0,1
D	0,6
γ	0,9
Condições de contorno	
Parede X = 0	Total reflexão
Parede X = Xmáx	Total reflexão
Parede Y = 0	Total reflexão
Parede Y = Ymáx	Total reflexão

Fonte: Elaborado pelo autor.

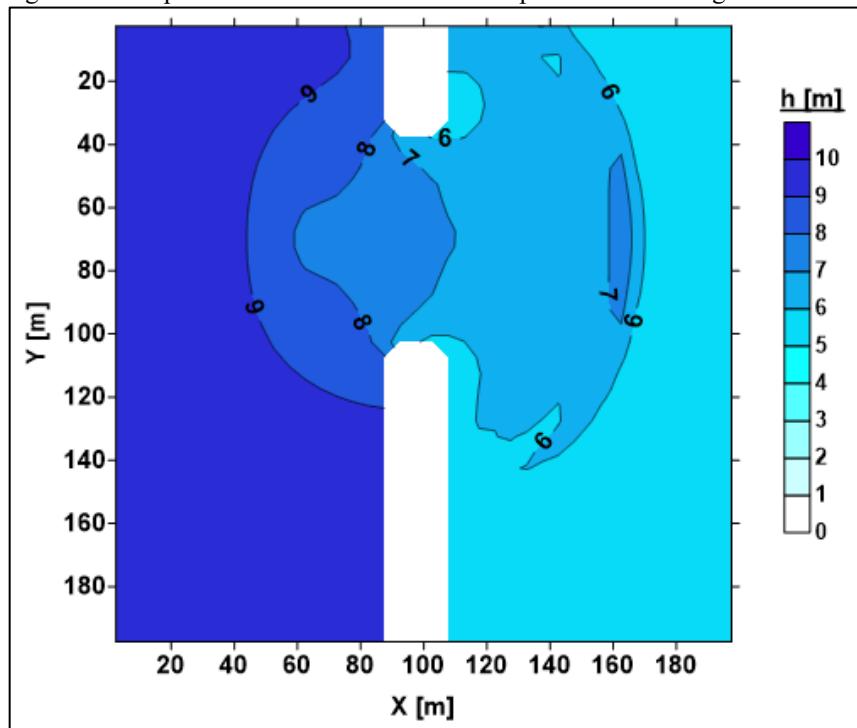
Os resultados de profundidade de água e velocidades obtidos após 7,2 segundos de simulação estão representados pelas Figuras 29, 30 e 31.

Figura 29 - Resultados de profundidade de água do teste 2.



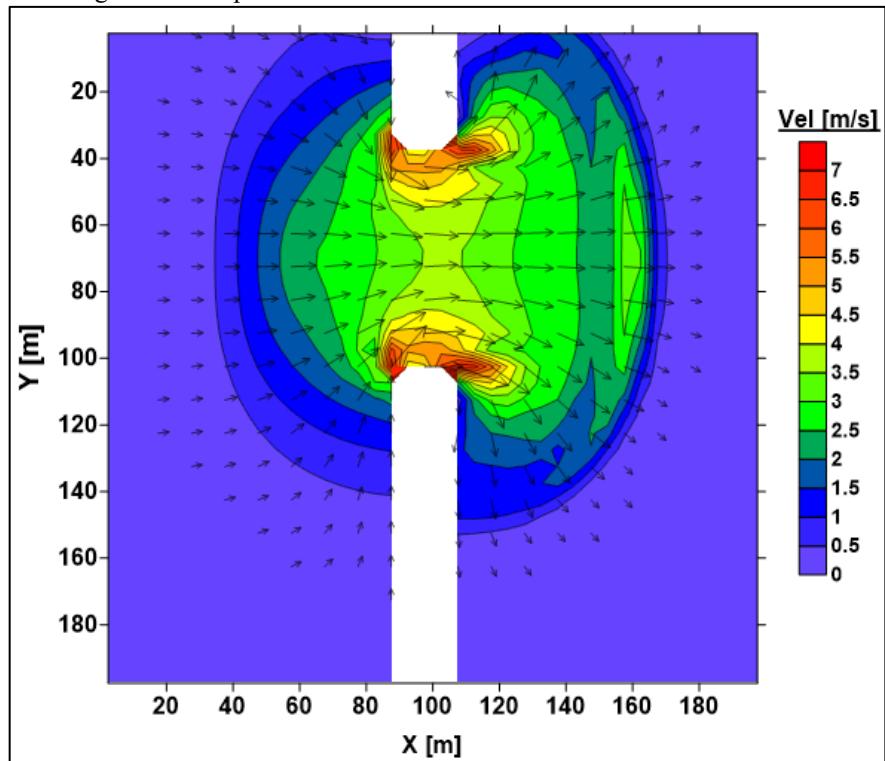
Fonte: Elaborado pelo autor.

Figura 30 - Mapa de contorno dos resultados de profundidade de água do teste 2.



Fonte: Elaborado pelo autor.

Figura 31 - Mapa de contorno dos resultados de velocidade do teste 2.



Fonte: Elaborado pelo autor.

Este teste demorou em torno de 30 segundos para ser processado no computador com as especificações citadas no início deste capítulo.

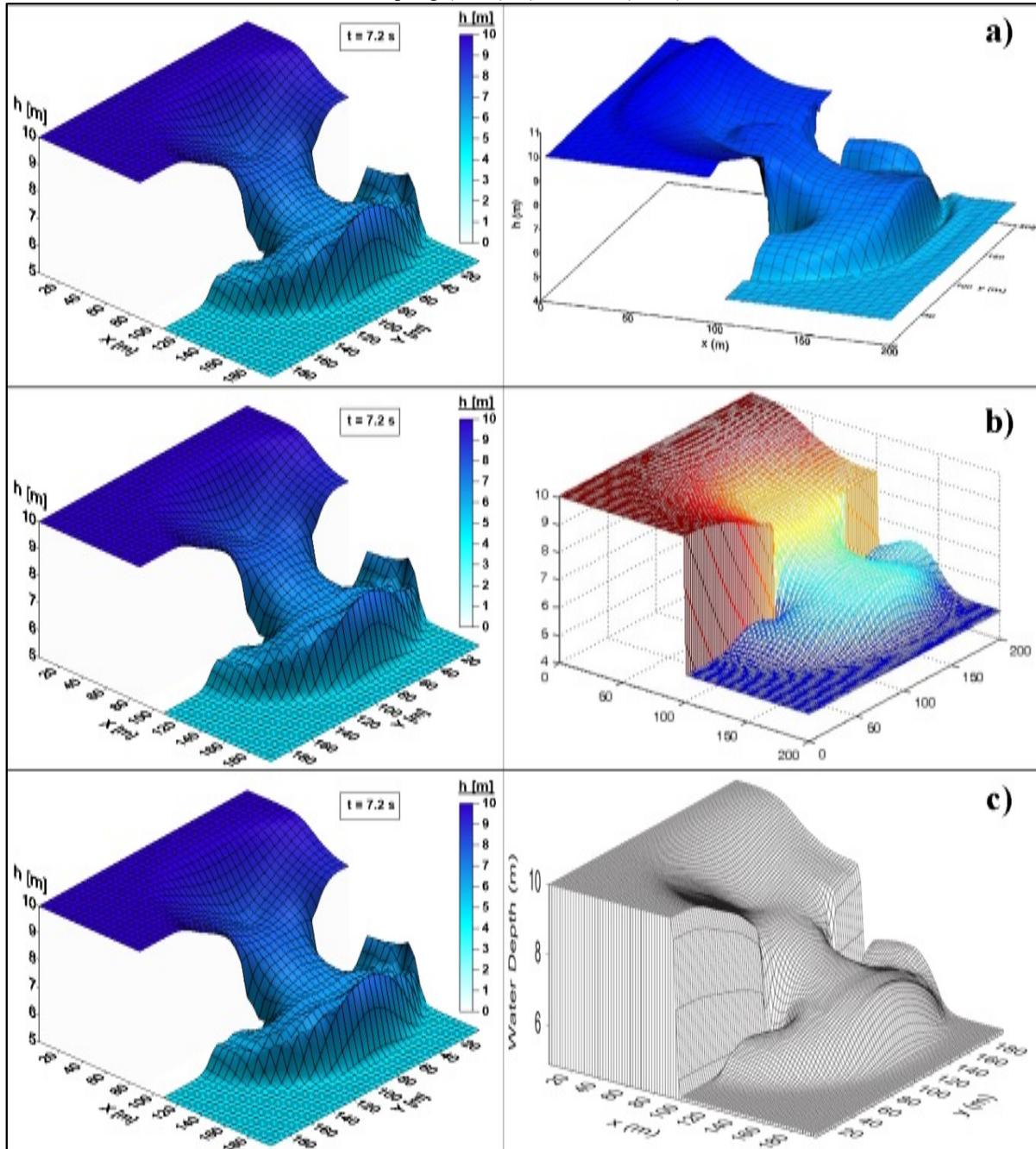
Observando estes resultados, nota-se que a superfície de água foi contínua em toda a área de estudo, não apresentando picos ou valores anormais, os quais significariam possíveis erros durante os cálculos. Estes valores espúrios seriam decorrentes de instabilidades numéricas, as quais foram inibidas pela adoção de passos de tempo curtos, regulados por condições de estabilidade descritas no capítulo anterior. Destaca-se que, em problemas onde as frentes de onda são abruptas, conforme o caso de rupturas de barragens, aumentam-se as possibilidades da solução numérica ser destruída por instabilidades. Por isto que a adoção de passos de tempo pequenos contribui para a continuidade da solução.

Esta simulação evidenciou velocidades mais altas próximas às laterais da abertura, onde se notam grandes gradientes de profundidade (dh/dx).

Para verificar se estes resultados gerados foram satisfatórios, fez-se comparação com os comportamentos das superfícies livres calculadas por Erduran (2013), Li e Fan (2017) e Zhendong e Junqiang (2015), que utilizaram os métodos dos volumes finitos, diferenças finitas generalizado e método das diferenças finitas, respectivamente.

Neste paralelo entre as simulações, notou-se que o resultado obtido teve similaridade com todos os resultados apresentados, sendo observada uma maior proximidade com os testes realizados por Erduran (2013) e por Zhendong e Junqiang (2015). Os resultados destes trabalhos citados estão ilustrados na Figura 32, onde foram colocados ao lado do resultado gerado pelo SimHidro2D para uma melhor visualização.

Figura 32 - Comparação dos resultados do teste 2 com os resultados de: a) Li e Fan (2017); b) Zhendong e Junqiang (2015); c) Erduran (2013).



Fonte: Li e Fan (2017), Zhendong e Junqiang (2015) e Erduran (2013), adaptado pelo autor.

Com estes resultados, foi então verificado que a rotina de cálculo elaborada estava gerando resultados coerentes para este tipo de situação.

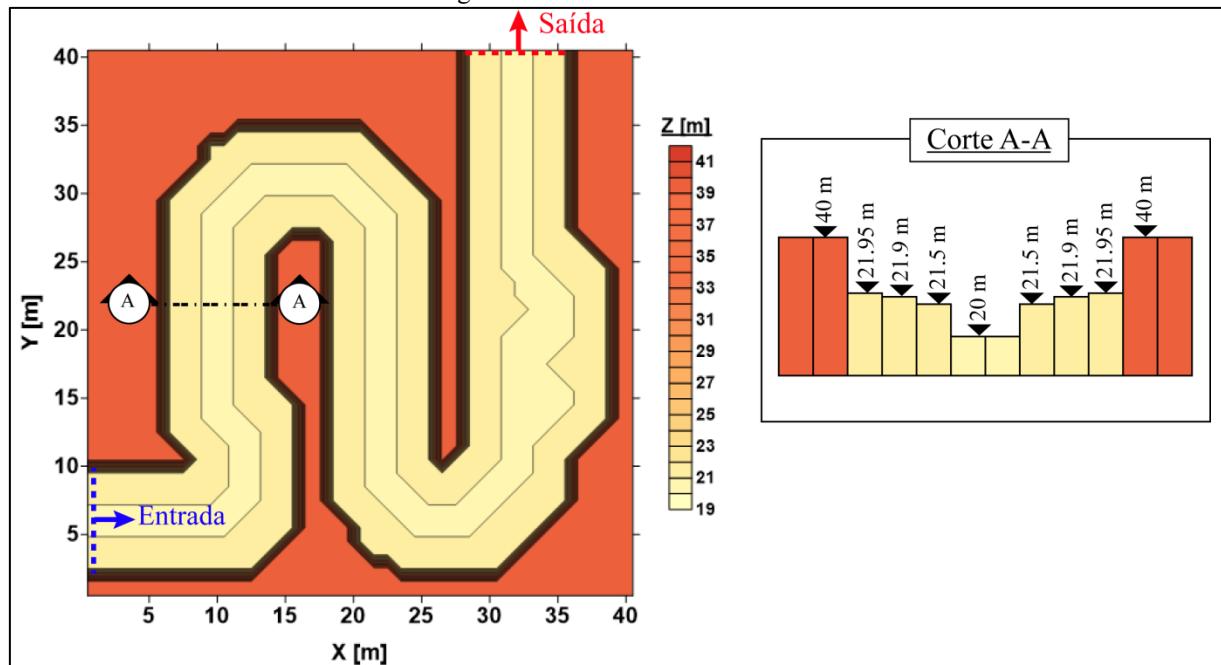
Semelhante ao teste anterior, para uma melhor compreensão e visualização dos resultados gerados, também foi disponibilizado no *YouTube* um vídeo com os resultados produzidos entre os tempos de 0 a 200 segundos, o qual pode ser visualizado através do seguinte link: <https://www.youtube.com/watch?v=GW6W17bRhf8>.

4.2.3 Teste 3 - Rio

Este teste foi feito com o intuito de analisar a rotina de cálculo quando sujeita à terrenos com diferentes altitudes.

Foi produzido um terreno com vales e traçado meandrado, simulando o canal de um rio. O domínio total é constituído por um quadrado de 41 por 41 metros, o qual é atravessado por um vale. Esse domínio de cálculo é ilustrado na Figura 33.

Figura 33 - Geometria do teste 3.



Fonte: Elaborado pelo autor.

As configurações utilizadas neste teste estão descritas na Tabela 4.

Tabela 4 - Configurações utilizadas no teste 3.

Nós em X	41
Nós em Y	41
Total de nós	1681
Tempo final [s]	100
Coef. de Manning	0,04
Número de CFL	0,1
D	0,6
γ	0,4
Condições de contorno	
Pare de X = 0	Total reflexão
Pare de X = Xmáx	Total reflexão
Pare de Y = 0	Total reflexão
Pare de Y = Ymáx	Sem reflexão

Fonte: Elaborado pelo autor.

Foi utilizado o valor de Manning de 0,04, pois, segundo Porto (2006), este valor representa rios retilíneos com vegetações e pedras.

Os valores utilizados para a entrada contínua de água estão descritos na Tabela 5.

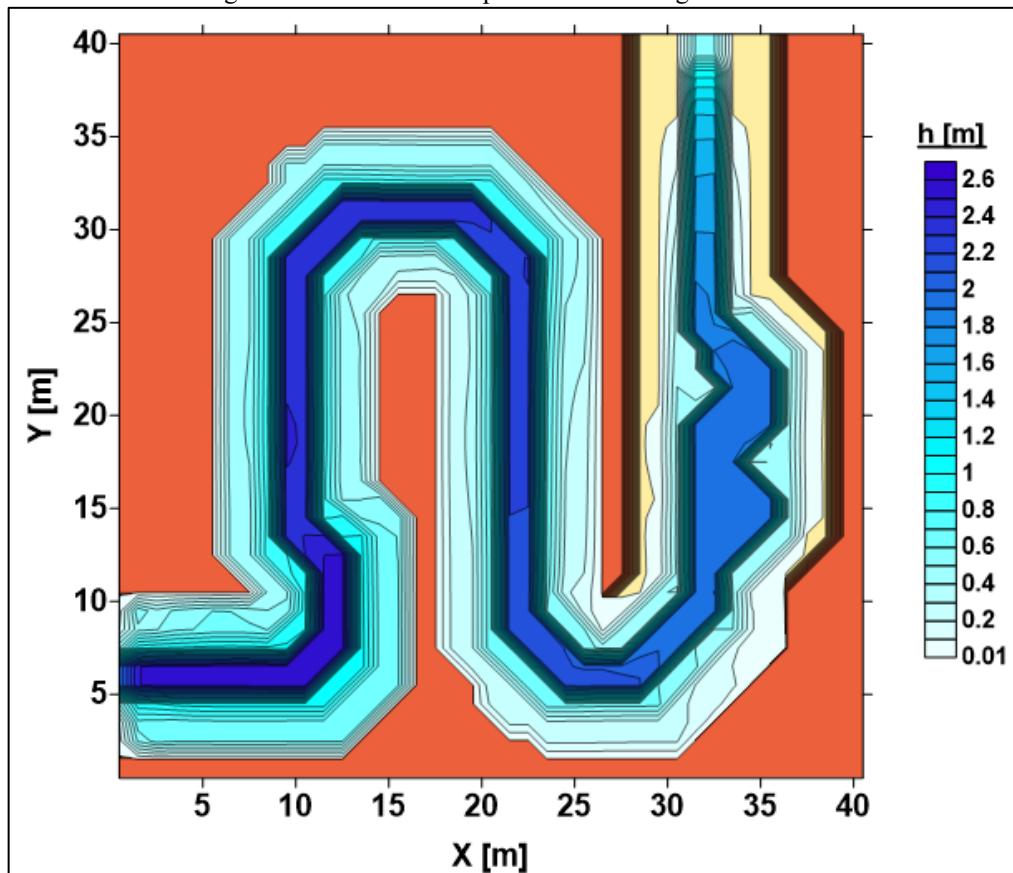
Tabela 5 - Valores utilizados para a entrada contínua de água.

Nó		h [m]	u [m/s]	v [m/s]
i	j			
1	3	0,05	2	0
1	4	0,1	2	0
1	5	0,5	2	0
1	6	2	2	0
1	7	2	2	0
1	8	0,5	2	0
1	9	0,1	2	0
1	10	0,05	2	0

Fonte: Elaborado pelo autor.

Os mapas de contorno com os resultados de profundidades de água e velocidades de escoamento, após 100 segundos de simulação, estão ilustrados nas Figuras 34 e 35.

Figura 34 - Resultados de profundidade de água do teste 3.

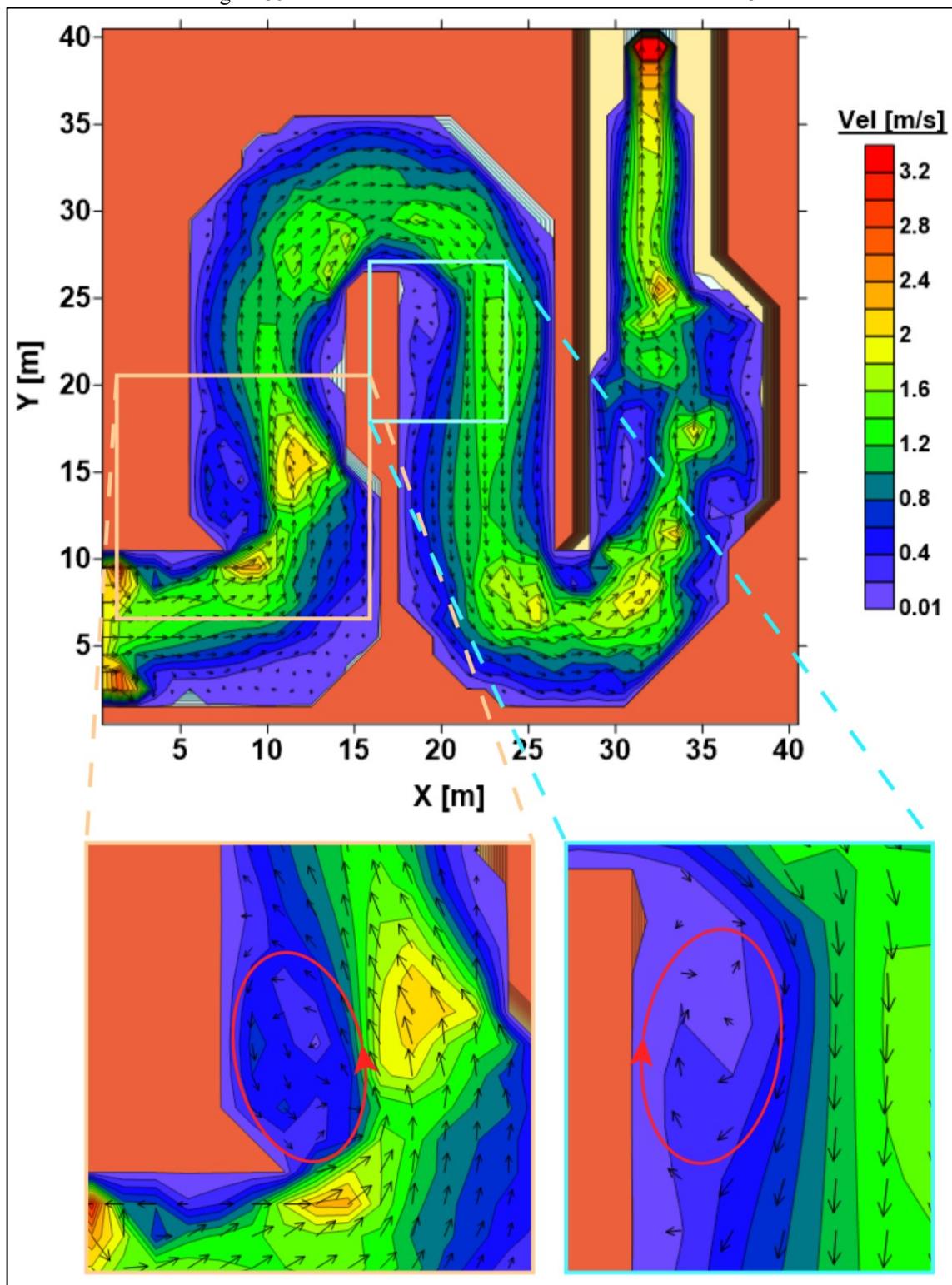


Fonte: Elaborado pelo autor.

O computador com as configurações descritas no início deste capítulo demandou cerca de 10 minutos para processar este teste.

Observando na Figura 34 os resultados de profundidade de água, nota-se que a rotina de cálculos gerou resultados coerentes, já que, as maiores profundidades surgiram justamente onde havia as cotas mais baixas do terreno. Os resultados desta simulação também não evidenciaram picos anormais de profundidades de água.

Figura 35 - Resultados de velocidade de escoamento do teste 3.



Fonte: Elaborado pelo autor.

Nota-se através da Figura 35 que, assim como nos resultados de profundidade de água, também foram computadas as velocidades mais altas nas regiões de maiores profundidades e as velocidades mais baixas nas regiões de menores profundidades. Estes resultados eram

esperados, pois, quanto menor a profundidade de água em uma célula, menor será o seu raio hidráulico, e, consequentemente, maior a perda de velocidade de escoamento neste volume.

Padrões interessantes capturados pela simulação foram as regiões de recirculação de água, como indicado na Figura 35. Numericamente isto sugere que os termos difusivos da equação governante têm contribuição decisiva nos resultados gerados. Nas equações de Saint-Venant que desprezam os termos difusivos interligados à viscosidade turbulenta, dificilmente recirculações tão nítidas como estas são capturadas.

É importante comentar que, ao capturar regiões de recirculação a jusante das curvas acentuadas, o código aqui construído demonstra potencial de simular trechos de rios meandrados.

4.3 Simulação – Reservatório Bom Jardim

Para uma avaliação de como a rotina de cálculos se comportaria quando fosse aplicada em geometrias reais, foi realizada a simulação do escoamento no Reservatório Bom Jardim, situado no município de Uberlândia-MG.

O Reservatório Bom Jardim tem grande importância para a cidade de Uberlândia, pois é uma das principais fontes de abastecimento de água da cidade.

A área de estudo tem como limite inferior a coordenada S 7.892.100, E 786.700, e como limite superior a coordenada S 7.896.490, E 788.100, ambas em coordenadas UTM na zona 22S. A sua localização está sinalizada em vermelho na Figura 36.

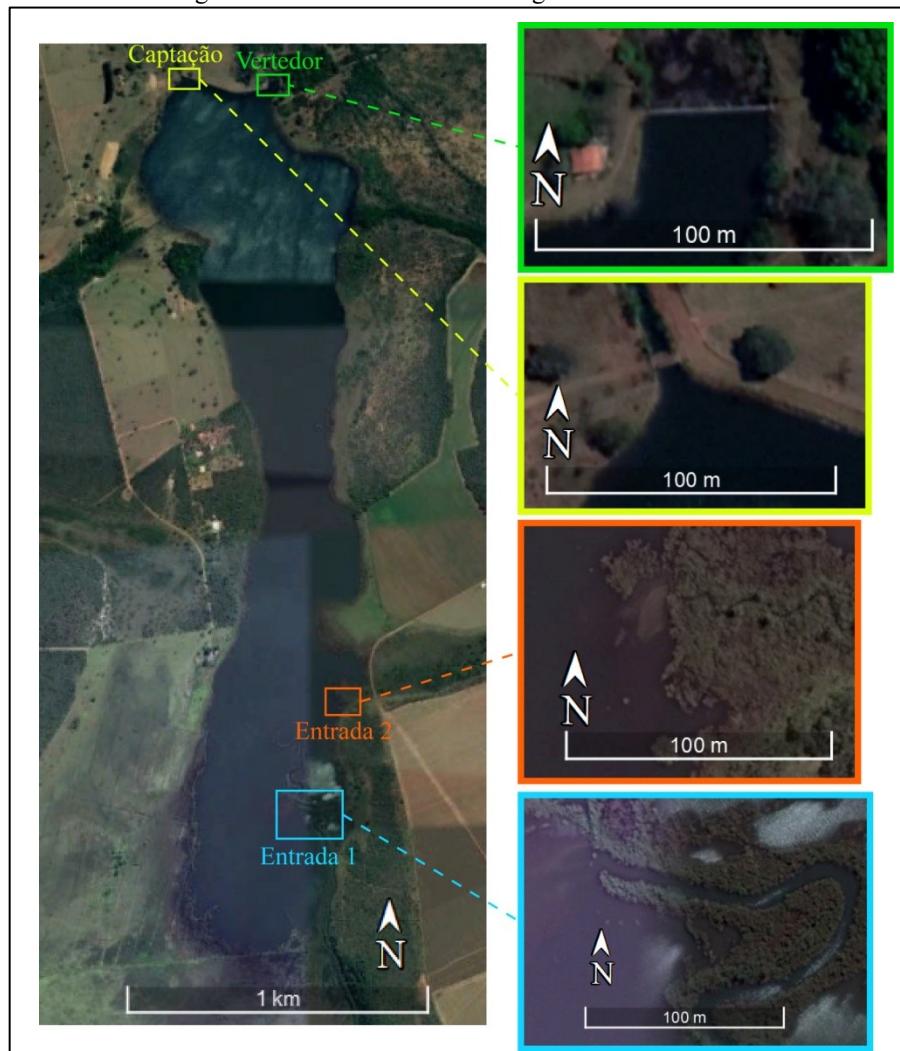
Figura 36 - Localização do Reservatório Bom Jardim.



Fonte: Google (2018).

O reservatório apresenta duas entradas de água principais, sendo uma de aproximadamente 9,80 metros de largura e a outra de aproximadamente 1 metro de largura. Estas entradas principais situam-se na porção sul desse corpo de água. Além disto, há um canal de captação de água para a adutora da Estação de Tratamento de Água do município de Uberlândia, de aproximadamente 6 metros de largura e um vertedor, cuja soleira tem cerca de 38 metros de comprimento. Ambos os dispositivos de saída de água são adjacentes à barragem, na porção norte do reservatório. Estas entradas e saídas de água estão indicadas e ilustradas na Figura 37.

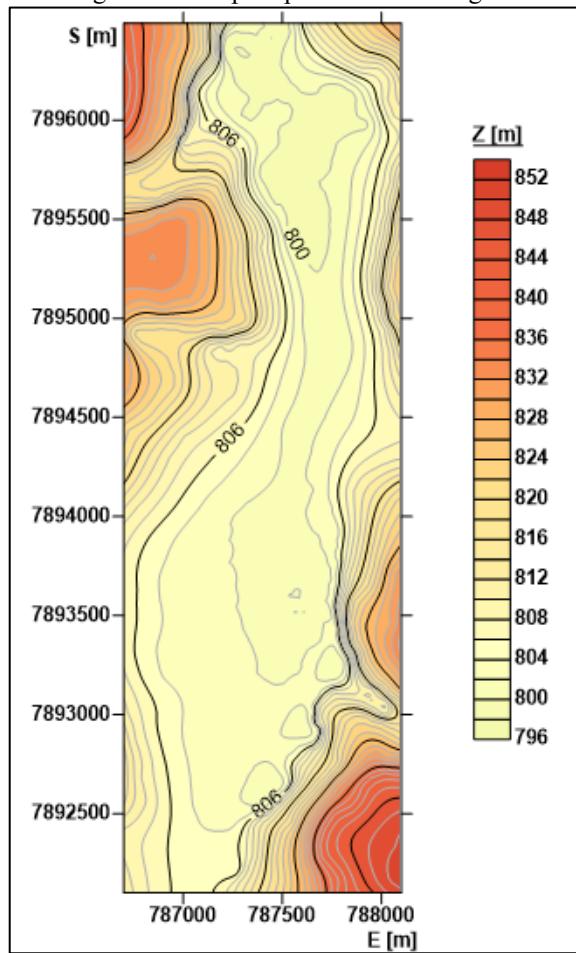
Figura 37 - Entradas e saídas de água do reservatório.



Fonte: Google (2018) adaptado pelo autor.

Neste trabalho, utilizou-se mapa topo-batimétrico digital produzido por Martins (2018), o qual foi elaborado com base nas cartas planialtimétricas levantadas pelo Exército Brasileiro no ano de 1983. Tal mapa é apresentado na Figura 38.

Figura 38 - Mapa topo-batimétrico digital.

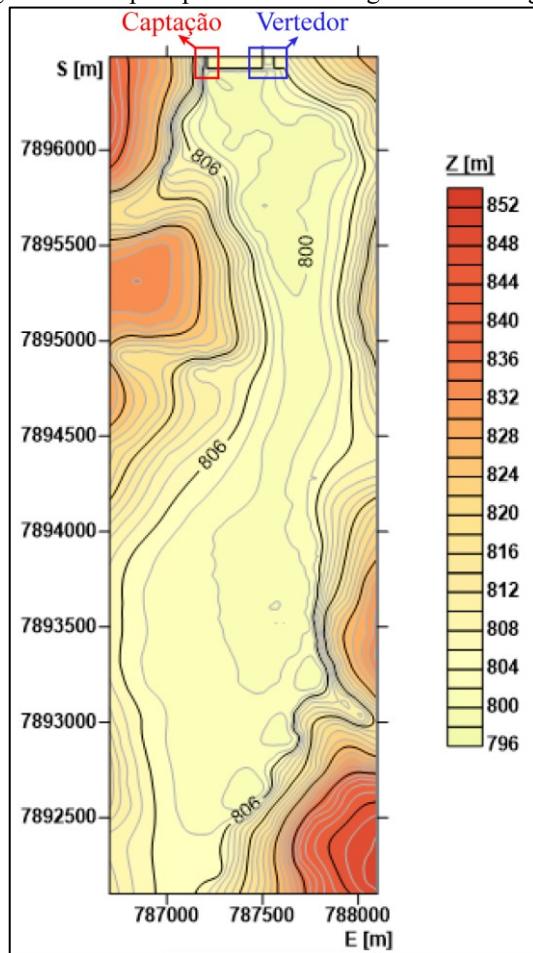


Fonte: Elaborado pelo autor.

As cartas planialtimétricas que originaram o modelo topo-batimétrico foram produzidas em 1983, ano em que ainda não existia o reservatório, já que o mesmo foi formado apenas em Dezembro de 1992, segundo Martins (2018). Ainda segundo Martins (2018), as cotas altimétricas da entrada do vertedor e do fundo do canal de captação são 804 e 801,5 metros, respectivamente.

Como o mapa topo-batimétrico digital original não apresentava a barragem, esta foi então adicionada manualmente, com o auxílio do programa computacional *Surfer*, a barragem juntamente com o canal de captação e vertedor. Para isto, foram obtidas as coordenadas e larguras da barragem e das entradas do canal de captação e do vertedor através do *Google Earth*. O mapa modificado está ilustrado na Figura 39.

Figura 39 - Mapa topo-batimétrico digital com barragem.



Fonte: Elaborado pelo autor.

Os valores das vazões fixas de entrada e saída utilizadas foram próximos das medições informadas por Martins (2018), as quais estão indicadas na Tabela 6.

Tabela 6 – Vazões fixas de entrada e saída utilizadas.

Localização	Vazão [m³/s]
Entrada 1	4
Entrada 2	1
Captação	-3,5

Fonte: Elaborado pelo autor.

4.3.1 Simulação utilizando o SimHidro2D

Para esta simulação, adotaram-se dimensões das células da área de estudo de uma maneira que elas não ficassem demasiadamente grandes, a ponto de ter uma resolução ruim dos resultados. Por outro lado, atentou-se para que as mesmas células de discretização também não ficassem pequenas demais, para não ter uma quantidade elevada de células que ocasionariam em uma grande quantidade de cálculos com aumento excessivo do custo

computacional. Salienta-se que todo o modelo topo-batimétrico foi discretizado e não apenas a área interna do reservatório. Assim, o domínio simulado torna-se retangular, fator que facilita o uso de células retangulares e malhas estruturadas regulares, típicas do Método de Diferenças Finitas. Neste caso, dependendo da cota da superfície livre, que deve ser calculada a cada passo de tempo, as células tornam-se secas ou molhadas, indicando se elas estão fora ou dentro do reservatório. Ou seja, a delimitação dos limites do reservatório é identificada pela própria rotina de cálculo.

As configurações utilizadas nesta simulação estão descritas na Tabela 7.

Tabela 7 - Configurações utilizadas na simulação do reservatório.

Nós em X	141
Nós em Y	440
Total de nós	62040
Tempo final [s]	3600
Coeff. de Manning	0,04
Número de CFL	0,1
D	0,6
γ	0,9
Condições de contorno	
Parede X = 0	Total reflexão
Parede X = Xmáx	Total reflexão
Parede Y = 0	Total reflexão
Parede Y = Ymáx	Sem reflexão

Fonte: Elaborado pelo autor.

Foi utilizado o valor de Manning de 0,04, pois este valor é indicado para corpos de água retilíneos com vegetações e pedras, segundo Porto (2006).

Além destas configurações, foi utilizado, para solução numérica das equações de Saint-Venant, o método de *Lax-Friedrichs*, já descrito no subitem 2.5.1.

Para a inserção das vazões fixas de entrada descritas na Tabela 6, foram adotadas as profundidades nas células de entrada e, em seguida, calculadas as velocidades para resultar na vazão desejada. Os valores de profundidades e velocidades utilizados para as células de entrada de vazão estão indicados na Tabela 8.

Tabela 8 - Valores de profundidades e velocidades utilizados para as células de entrada de vazão.

Localização	h [m]	ΔY [m]	A = h * ΔY [m²]	Q [m³]	V = Q/A [m/s]
Entrada 1	0.4	10	4	4	1
Entrada 2	0.4	10	4	1	0.25

Fonte: Elaborado pelo autor.

Para a aplicação da vazão de saída, na tomada de água pelo canal de derivação, a velocidade foi calculada a cada passo de tempo, de acordo com a profundidade instantânea da face da célula.

Para evitar velocidades elevadas durante o início da simulação, quando a profundidade de água no canal de captação ainda é pequena, foi iniciado o processo de retirada de volume de água apenas quando a profundidade de água na captação fosse superior a 1 metro. Para aplicar isto, foram feitas as seguintes condições.

Se $h_{cap} < 1 \{ vel_{cap} = 0$

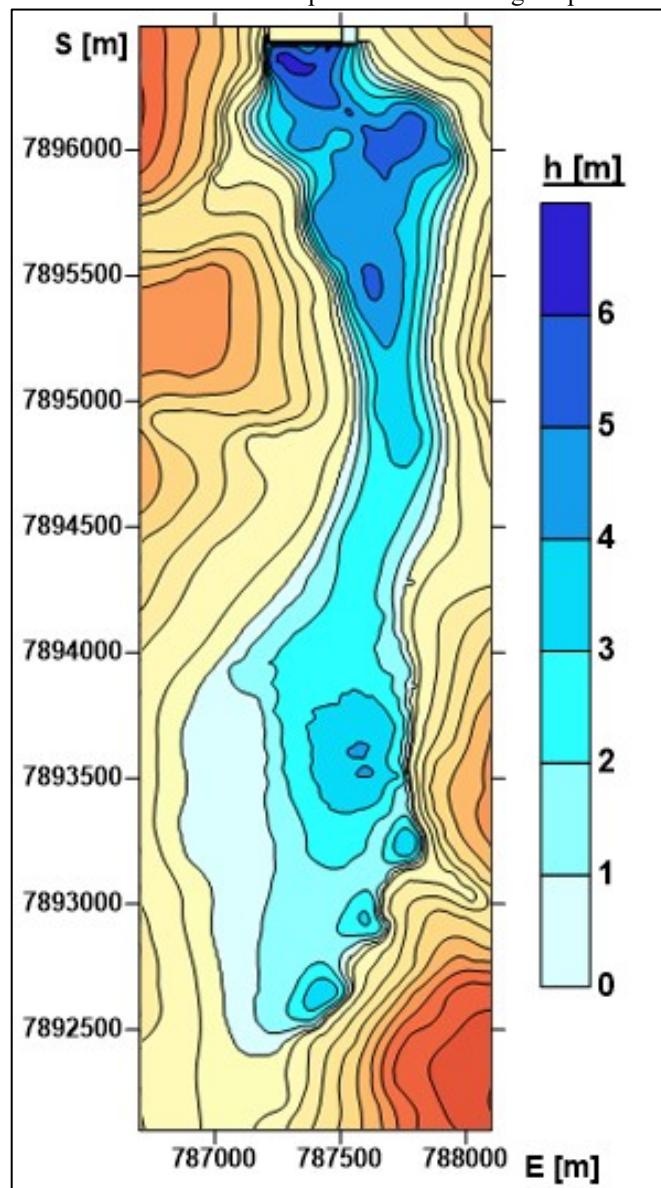
$$\text{Senão } \left\{ vel_{cap} = \frac{Q_{cap}}{\Delta x \times h_{cap}} \quad \text{ou seja,} \quad vel_{cap} = \frac{3.5}{10 \times h_{cap}}$$

Sendo, h_{cap} a profundidade de água instantânea na captação de água, em m; vel_{cap} a velocidade de escoamento calculada na captação, em m/s; Q_{cap} a vazão adotada de captação, em m^3/s .

Para o extravasor, não foi fixada nenhuma vazão, já que todo o volume de água que passasse pelo vertedouro deveria ser eliminado da área de estudo. Para isto, foi configurada, para a parede de $Y_{máx}$, a condição de contorno Sem Reflexão, como já descrito na Tabela 7.

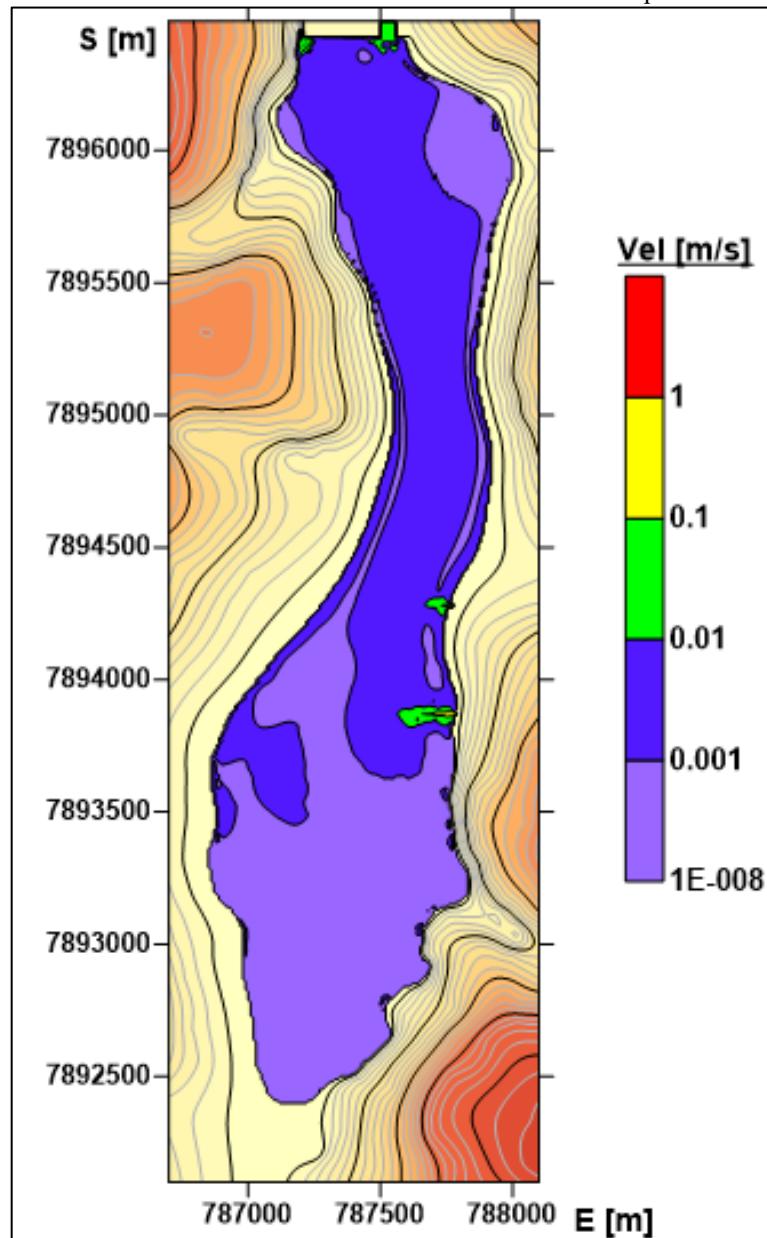
Os resultados após 1 hora de simulação de profundidades de água e velocidades de escoamento estão ilustrados nas Figuras 40 e 41, respectivamente.

Figura 40 – Resultados do SimHidro2D de profundidades de água após 1 hora de simulação.



Fonte: Elaborado pelo autor.

Figura 41 – Resultados do SimHidro2D de velocidades de escoamento após 1 hora de simulação.



Fonte: Elaborado pelo autor.

Para processar esta simulação, o computador, cujas configurações foram descritas no início deste capítulo, demandou cerca de 13 horas. Nota-se que este intervalo é visivelmente maior do que aqueles utilizados nas simulações teste. Isto decorre do maior tempo de simulação e da elevada quantidade de células necessárias para discretizar o modelo topo-batimétrico que representa a área de estudo. Além disto, este refinamento foi necessário para que o código pudesse representar adequadamente os contornos do reservatório.

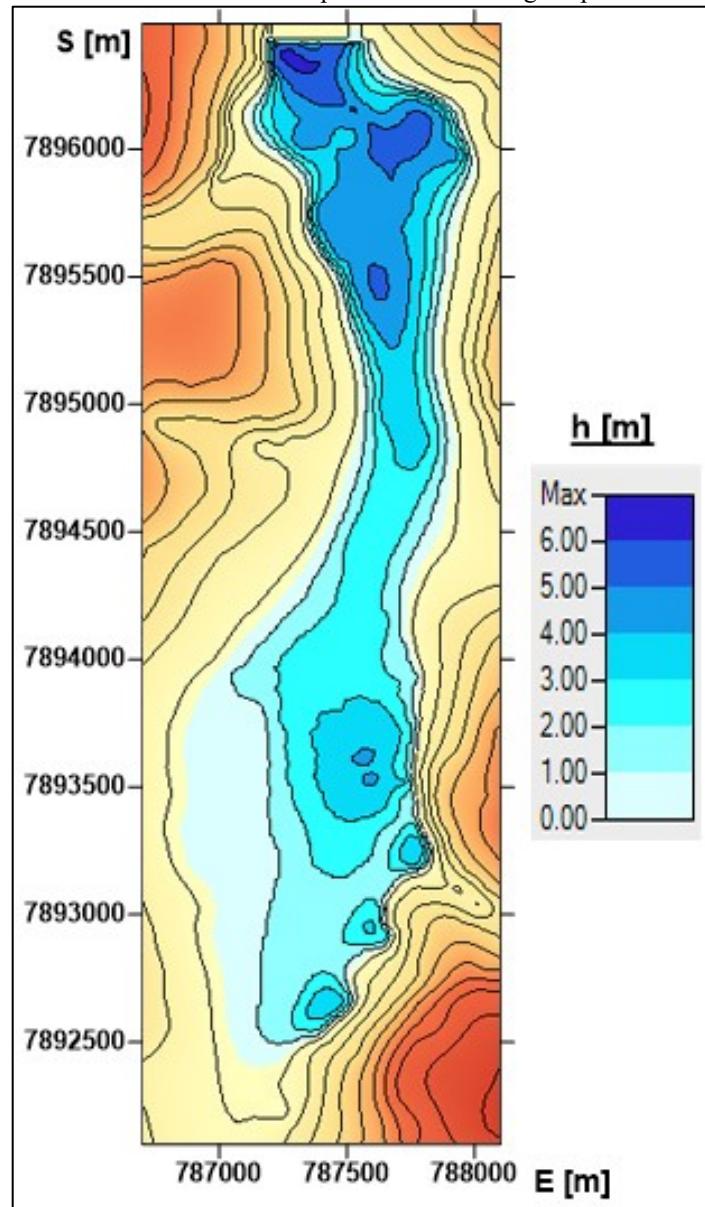
Estes resultados serão analisados e comparados no subitem 4.3.3, assim como os mapas de velocidades de escoamento com as suas direções sinalizadas com vetores.

4.3.2 Simulação utilizando o HEC-RAS

Para que a comparação de resultados fosse válida, todos os dados empregados para a simulação no HEC-RAS foram os mesmos utilizados na simulação feita no SimHidro2D.

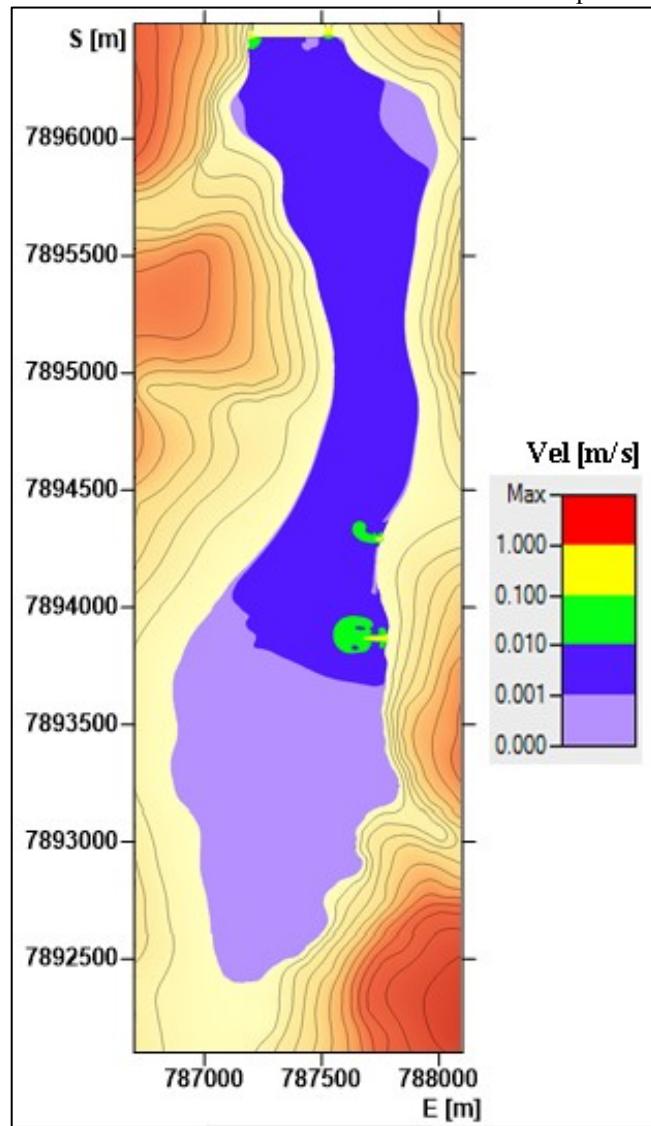
Os resultados de profundidades de água e velocidades de escoamento após 1 hora de simulação estão ilustrados nas Figuras 42 e 43, respectivamente.

Figura 42 - Resultados do HEC-RAS de profundidades de água após 1 hora de simulação.



Fonte: Elaborado pelo autor.

Figura 43 - Resultados do HEC-RAS de velocidades de escoamento após 1 hora de simulação.



Fonte: Elaborado pelo autor.

4.3.3 Análise comparativa dos resultados

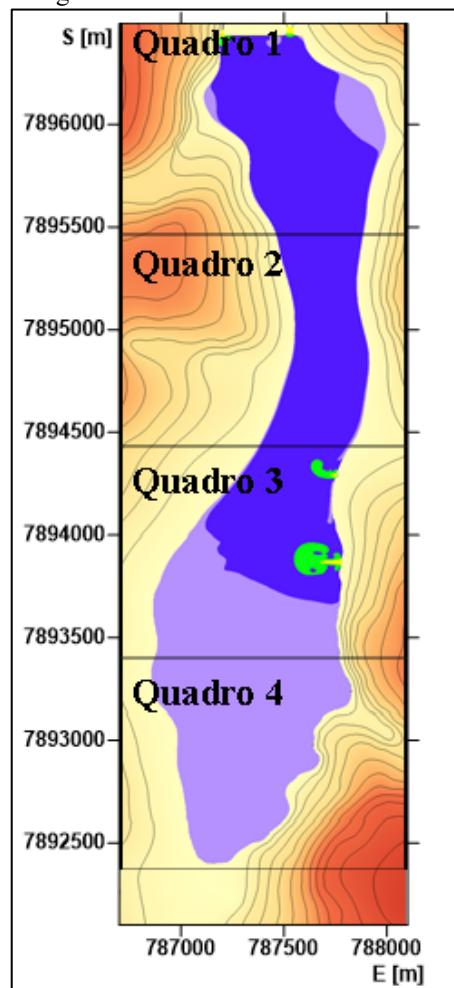
Como a superfície líquida tem cota praticamente constante ao longo de todo o reservatório, salvo nas regiões de entradas e saídas de água, já era previsto que ambas as simulações gerassem resultados de profundidades de água semelhantes. No caso da simulação de ambientes aquáticos lênticos e de grandes dimensões, como reservatórios e lagos, em situações que não representem a passagem de ondas de cheia ou de ondas de impacto (decorrentes de desmoronamento de encostas adjacentes), é comum que a cota da superfície livre sofra apenas ligeiras modificações, em singularidades específicas como nas entradas e saídas de água.

Neste item serão analisados os mapas de campos de velocidade. Inicialmente procurou-se exportar os resultados numéricos gerados no HEC-RAS, para que ambas as simulações fossem plotadas no mesmo programa, para uma melhor análise. Isto possibilitaria a quantificação das diferenças relativas entre as velocidades estimadas pelo SimHidro2D e o HEC-RAS. Entretanto, na atual versão 5.0.5 do HEC-RAS ainda não é possível exportar os resultados das simulações bidimensionais em forma de planilhas ou arquivo texto. Por esta razão, foram realizadas as comparações dos resultados do SimHidro2D plotados na ferramenta computacional *Surfer* com os resultados plotados no próprio HEC-RAS.

Devido às velocidades muito baixas encontradas no reservatório, o mapa de vetores de velocidades gerado pelo HEC-RAS não foi visível. Por esta razão, para as seguintes comparações foi utilizado o mapa de velocidades de escoamentos com traçadores de partículas produzido pelo HEC-RAS.

Para possibilitar a visualização dos vetores com uma melhor resolução, a área de estudo foi dividida em quatro quadros. A disposição desta divisão está ilustrada na Figura 44.

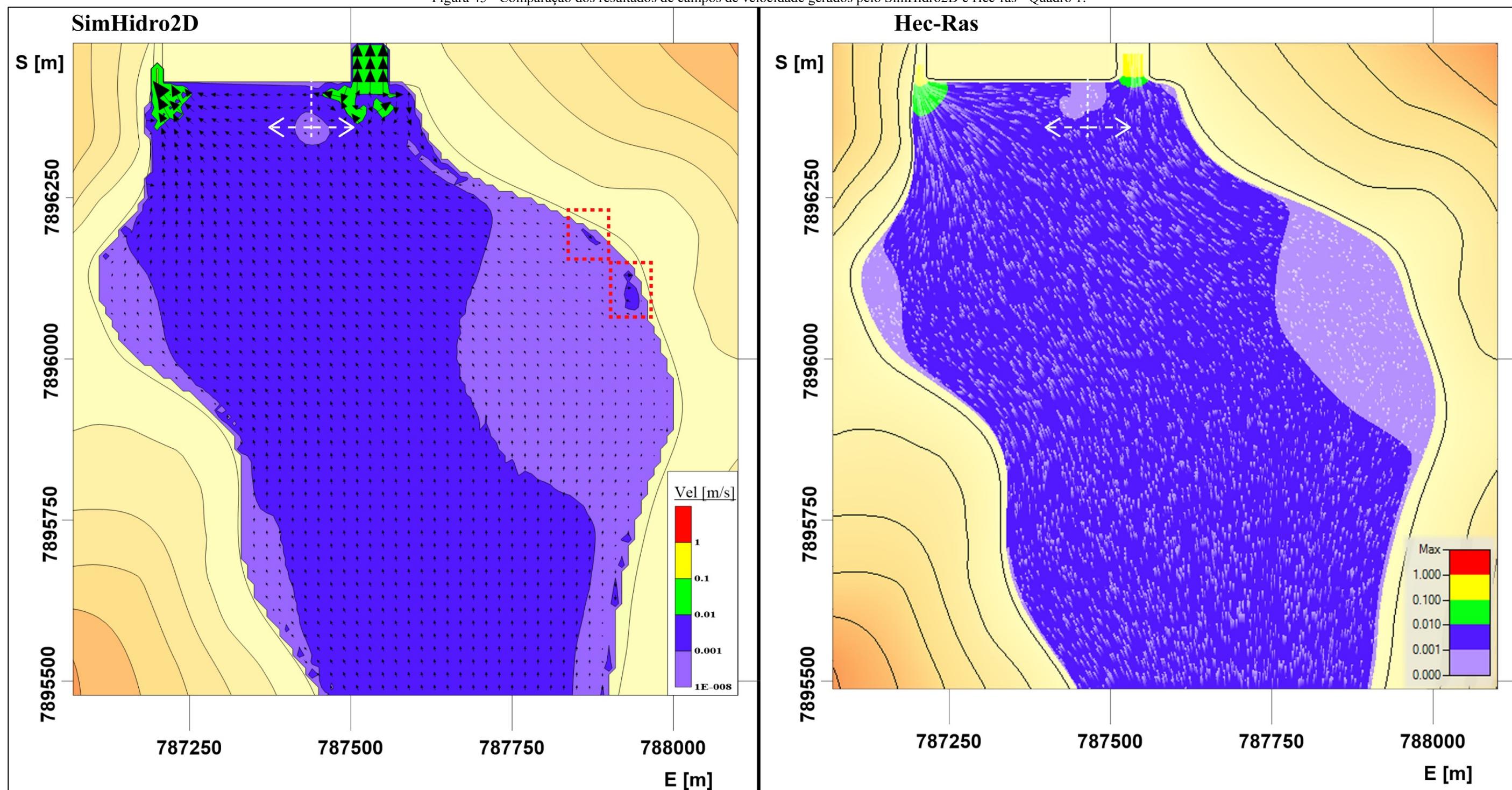
Figura 44 - Divisão da área de estudo.



Fonte: Elaborado pelo autor.

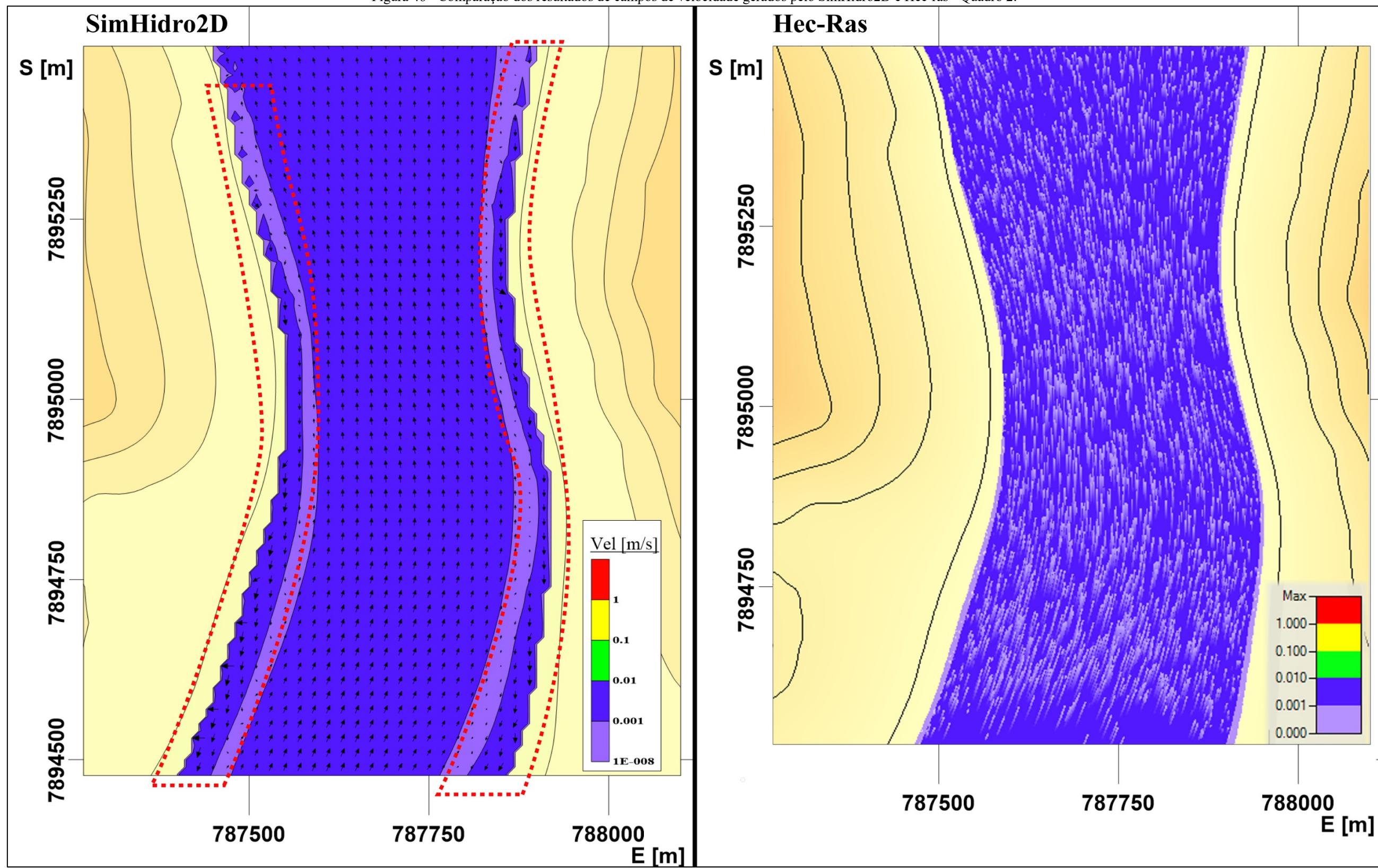
Para sinalizar as principais regiões de resultados semelhantes foram utilizadas linhas tracejadas brancas, enquanto que, para sinalizar as regiões de resultados divergentes, foram utilizadas linhas tracejadas vermelhas. Os mapas para as comparações dos resultados de campos de velocidade estão ilustrados nas Figuras 45 a 48.

Figura 45 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 1.



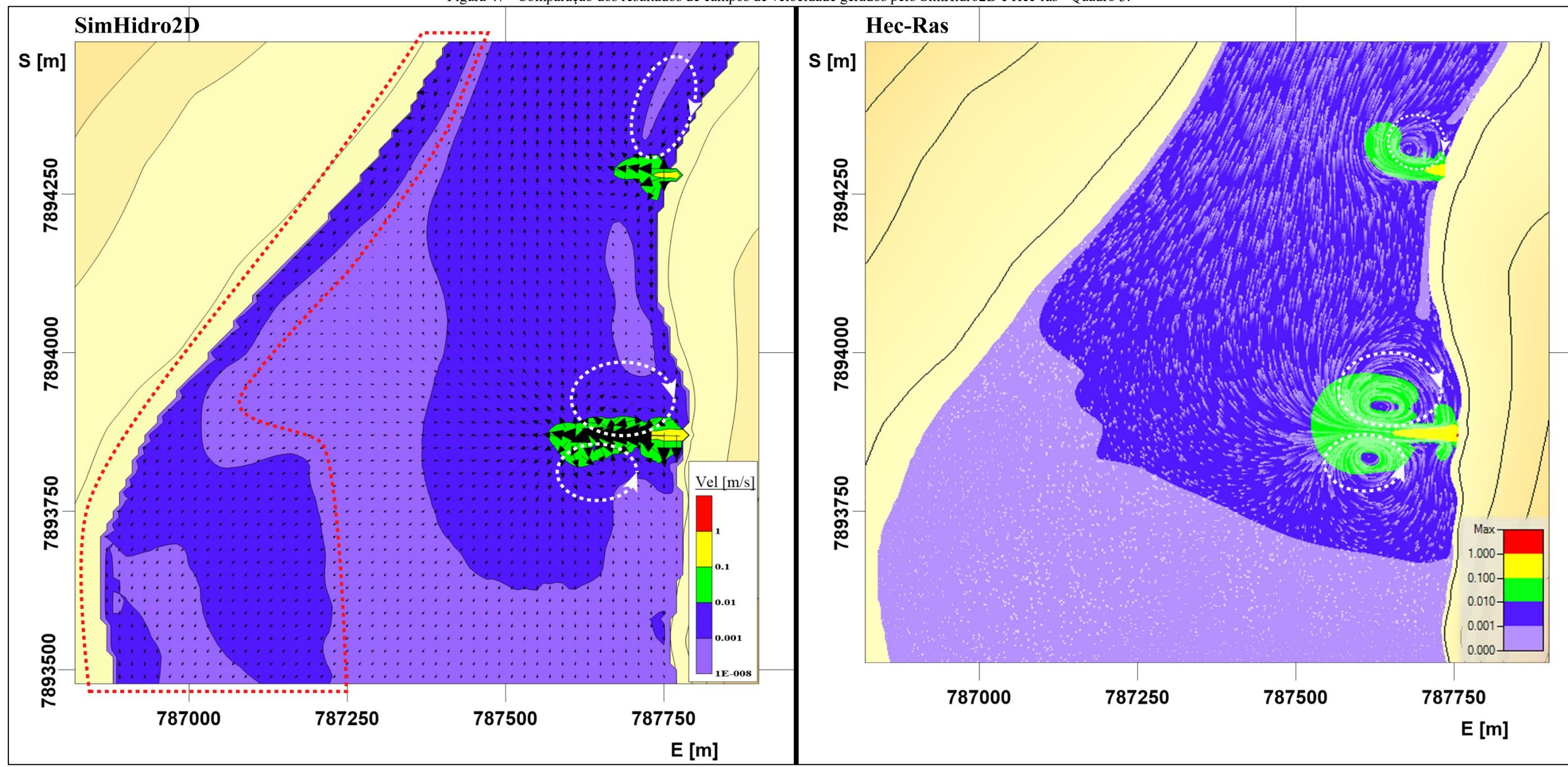
Fonte: Elaborado pelo autor.

Figura 46 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 2.



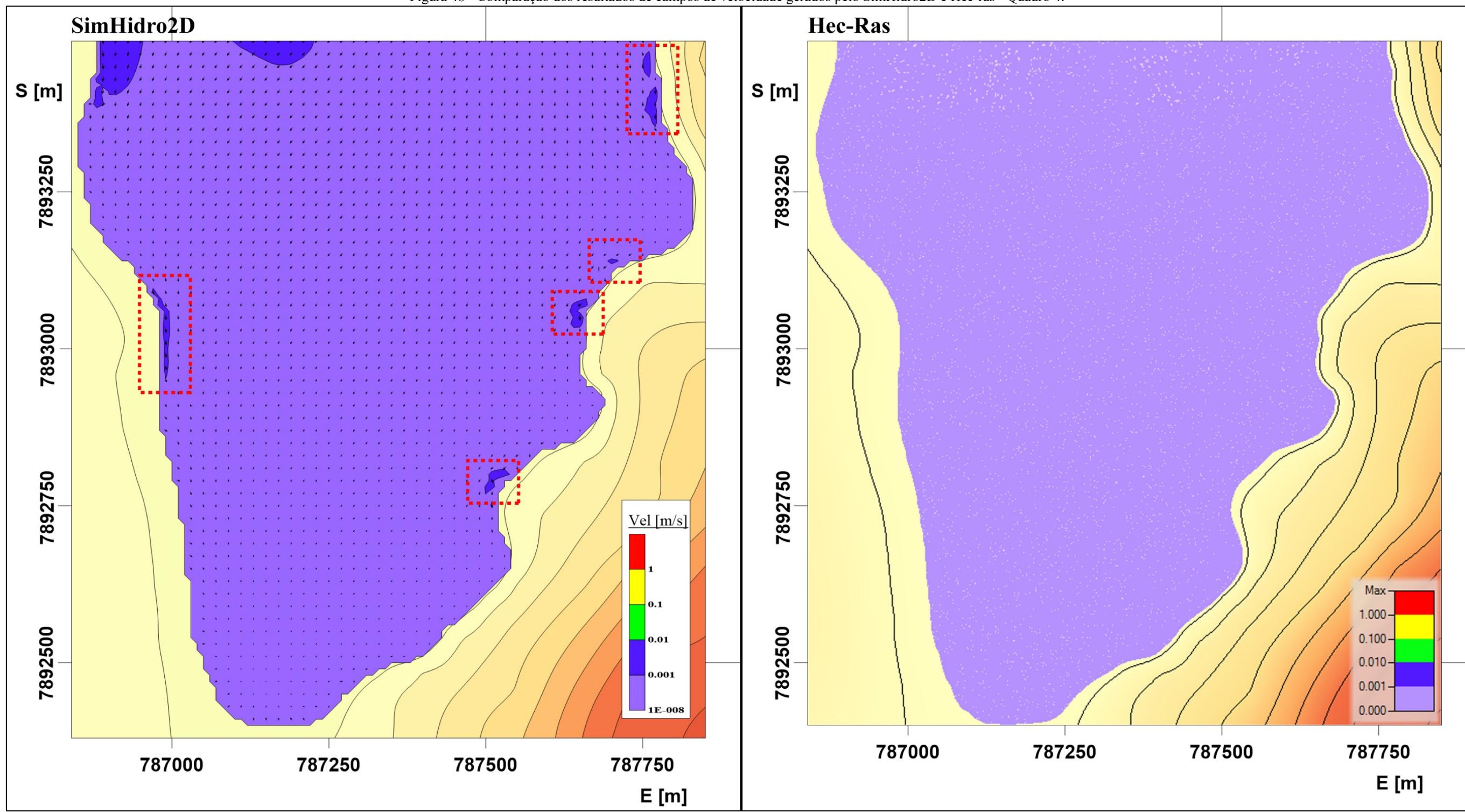
Fonte: Elaborado pelo autor.

Figura 47 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 3.



Fonte: Elaborado pelo autor.

Figura 48 - Comparação dos resultados de campos de velocidade gerados pelo SimHidro2D e Hec-ras - Quadro 4.



Fonte: Elaborado pelo autor.

Na região do Quadro 1, ilustrado pela Figura 45, foram observadas algumas pequenas regiões de velocidades ligeiramente mais altas, duas delas sinalizadas pelos retângulos vermelhos e outras menores sem sinalização, esta diferença pode ter ocorrido devido à algum volume de água que chegou na borda do reservatório e, ao ser rebatido, ocasionou este aumento na velocidade de escoamento.

Ainda sobre o Quadro 1, foi observado que os volumes de água de ambas as simulações tiveram um ponto em comum de divisão de direção de escoamento, no qual à direita deste ponto, os escoamentos tendem para o vertedouro, enquanto que à esquerda, os escoamentos tendem para a tomada de água do canal de derivação, como sinalizado em linhas tracejadas brancas.

Sobre a região do Quadro 2, ilustrada pela Figura 46, nos resultados do SimHidro2D surgiram regiões próximas às bordas do reservatório, sinalizadas por retângulos vermelhos, onde o sentido do escoamento foram contrários aos sentidos dos resultados do HEC-RAS. Esta diferença entre os resultados das simulações será abordada no final deste subitem.

Na região do Quadro 3, ilustrada na Figura 47, os resultados da simulação do SimHidro2D apresentaram uma região de escoamento com o sentido inverso dos resultados do HEC-RAS, assim como aconteceu no Quadro 2, esta diferença entre os resultados das simulações será abordada no final deste subitem.

Ainda sobre o Quadro 3, a maior área em azul, que representa o volume de água se deslocando no sentido em direção à barragem, foram similares em ambos os resultados e, além disto, ocorrem três regiões de recirculação de água em ambas as simulações, sendo duas delas na Entrada 1, e a terceira na Entrada 2. Ocorreram também, grandes regiões de baixas velocidades de escoamento, ilustradas pelas áreas em roxo claro, nas quais tendem à ocorrer uma maior sedimentação de sólidos. Nas entradas de água, ambos os modelos capturaram padrões de escoamento similares aos de jatos intensos. No caso específico da entrada 2, que tem baixa vazão, um melhor refinamento da malha nessa região, com a utilização de células menores, poderia capturar com maior detalhes este tipo de comportamento.

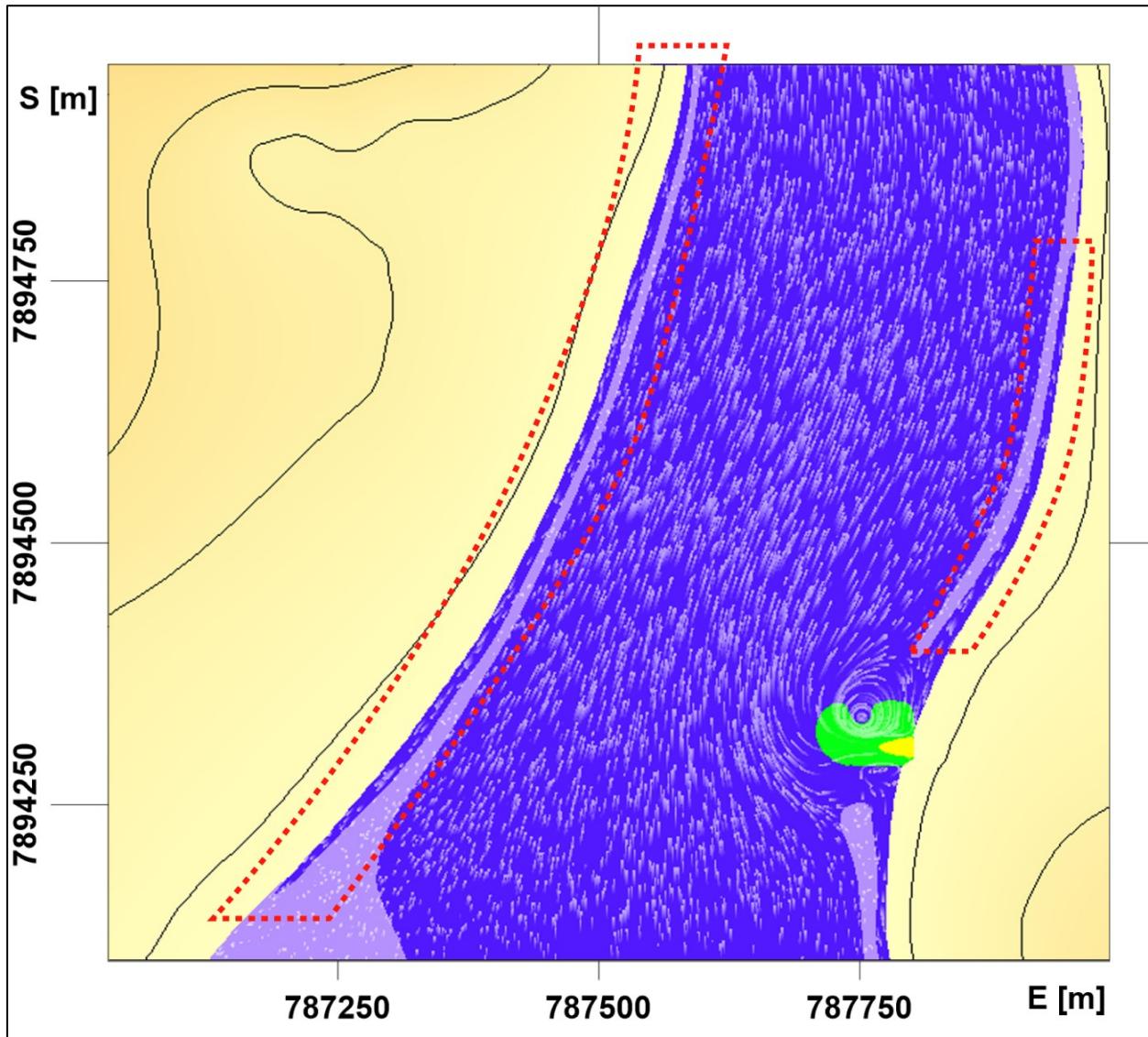
No Quadro 4, ilustrado pela Figura 48, a porção sul do reservatório apresentou predominância de velocidades de escoamento abaixo de 1mm/s, ou seja, uma região de baixas velocidades na qual há uma grande tendência de sedimentação de sólidos. O reservatório tem, nesses setores, os pontos com as menores velocidades, as quais coincidem com vastas zonas de águas rasas.

Nota-se que as entradas de água situam-se distantes e isto contribui para a estagnação das águas nessa região.

Assim como ocorreu no Quadro 1, no Quadro 4 os resultados da simulação do SimHidro2D também apresentaram pequenas regiões com aumento de velocidades, possivelmente ocasionadas pela reflexão sofrida por volumes de água com as bordas do reservatório.

Como já descrito, nos resultados gerados pelo SimHidro2D foram encontradas regiões em que os escoamentos tiveram sentidos contrários aos dos resultados do HEC-RAS. A princípio estas discrepâncias aparentaram terem sido geradas por erros na rotina de cálculo desenvolvida para o SimHidro2D. Entretanto, quando se observaram os resultados da simulação no HEC-RAS em tempos diferentes de 1 hora, identificou-se que, no tempo de 30 minutos por exemplo, também foram produzidos escoamentos com sentidos inversos ao do escoamento principal, como pode ser visualizado na Figura 49.

Figura 49 - Escoamentos com sentido inverso ao sentido do escoamento principal, na simulação de 30 minutos do HEC-RAS.



Fonte: Elaborado pelo autor.

Além disto, também foram encontradas regiões com acréscimos de velocidades semelhantes às regiões sinalizadas na Figura 48, descartando, desta maneira, a ideia de que estas discrepâncias de resultados fossem geradas devido a erros na rotina de cálculo do SimHidro2D.

Foi disponibilizado um vídeo no *YouTube* para uma melhor compreensão dos resultados da simulação de 1 hora feita no HEC-RAS, e também para a visualização dos escoamentos com sentido inverso e das regiões de acréscimo de velocidades na simulação de 30 minutos. Este vídeo pode ser acessado através do seguinte link:

<https://www.youtube.com/watch?v=8qdJTQ8BfZs>

Os resultados gerados pelas simulações feitas no SimHidro2D ou no HEC-RAS devem ser tratados como aproximações do escoamento no reservatório, uma vez que alterações nas vazões de entrada, devido a períodos secos ou chuvosos, sedimentações, mudanças na vazão vertida e retirada na tomada de água para o canal de derivação, em função de aberturas ou fechamentos de comportas, podem alterar as configurações localizadas dos campos de velocidade. Com aumento nas vazões de entrada, as zonas de influência dos jatos internos deverão ser mais extensas. Da mesma forma, com o aumento das vazões captadas e vertidas, as zonas de convergência das linhas de corrente também deverão ser mais amplas.

Todavia, de uma forma geral e independente das vazões de entrada, os padrões de escoamento não devem se alterar na grande massa líquida do reservatório. Assim, as zonas mais propícias a baixas velocidades deverão continuar com escoamento quase estagnado, tornando-as mais propícias ao assoreamento. Ainda nessas zonas, o eventual acúmulo de nutrientes, em decorrência basicamente de atividades agrícolas na bacia, pode facilitar a proliferação de algas. No estrangulamento do reservatório, detalhado no Quadro 2, deverão observar-se velocidades intermediárias, com sentido basicamente sul-norte.

CAPÍTULO 5

CONCLUSÕES

A ferramenta computacional SimHidro2D, através dos testes realizados, mostrou ser útil e eficiente principalmente quando aplicada a pequenas áreas de estudo, já que tem uma fácil entrada e saída de dados na forma de planilha. O programa pode ser alimentado por dados numéricos de coordenadas e cotas de um modelo digital topo-batimétrico, de forma que o simulador identifica as células contidas dentro do corpo de água. As condições de contorno entre células secas e molhadas limitam o escoamento dentro do reservatório. O uso do arranjo deslocado de malha, com velocidades armazenadas nas faces e profundidades no centro, facilita a aplicação dessas condições de contorno.

Percebeu-se que, à medida que foram aumentados os números de nós analisados e/ou o tempo total de simulação e diminuído o valor de CFL adotado, aumentava-se proporcionalmente o tempo necessário para o computador processar os cálculos. Portanto, para otimizar o tempo de processamento, é necessário utilizar o menor número possível de células, a ponto de não atrapalhar a qualidade e resolução dos resultados, e adotar o maior número possível de CFL, de forma que os erros residuais gerados não comprometam a simulação.

Foi possível observar a dificuldade que é produzir um programa computacional para simular escoamentos hidráulicos bidimensionais, uma vez que existem muitas variáveis, formas de relevos e interações terreno-água. Na modelagem de escoamentos em ambientes lênticos, como reservatórios, recomenda-se o uso de termos difusivos nas equações de Saint-Venant. Isto deve ser destacado, pois tratam-se de ambientes aquáticos onde não há efeito dominante da advecção. Esses termos, associados ao refinamento da malha, permitem que as simulações capturem vórtices, como aqueles verificados nos jatos internos originados da injeção de água no reservatório.

Apesar da facilidade em se utilizar o Excel como entrada e saída de dados, executando o programa em linguagem computacional VBA, o tempo de processamento do SimHidro2D é um fator negativo quando comparado a outras programas computacionais, como o próprio

HEC-RAS, que necessitou de um tempo muito menor de processamento para realizar a simulação do Reservatório Bom Jardim.

É interessante destacar a similaridade que foi atingida entre os padrões de escoamento gerados pelo SimHidro2D quando comparados com os resultados gerados pelo HEC-RAS na simulação realizada. Foram capturadas pelo SimHidro2D todas as recirculações de água que foram geradas pelos resultados do HEC-RAS. Também foi verificado, quando comparados ambos os resultados, a semelhança nos formatos e valores dos campos de velocidades gerados.

Com todos os resultados dos testes e da simulação realizados apresentados, o programa aqui desenvolvido provou-se ser capaz de capturar o comportamento de corpos hídricos, produzindo resultados semelhantes aos resultados gerados pelo HEC-RAS, que se trata de um programa difundido no meio técnico e científico.

Algumas recomendações para trabalhos futuros e melhorias para o SimHidro2D:

- Criar um programa computacional em outra linguagem de programação para se trabalhar exclusivamente com a rotina de cálculo. Ação necessária pois, ao se utilizar o algoritmo de cálculo dentro do Excel, o computador gasta mais recursos do que se fosse utilizado um programa exclusivo para isto.
- Implementar um método numérico implícito, tornando-se possível aumentar os incrementos de tempo durante os cálculos das simulações sem prejudicar a estabilidade das soluções numéricas. Desta forma, necessitando calcular menos instantes de tempo durante a simulação, ocasionando em menores tempos de processamento.
- Introduzir termos relativos às ações de ventos nas equações governantes para melhorar a qualidade dos resultados em situações com presença de ventos.
- Introduzir termos relativos à concentração de sedimentos, para realizações de simulações de transporte de sedimentos.

REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, V. S. Análise de Hidrodinâmica Ambiental e de Qualidade de Água na Baía de Guanabara via Modelagem Computacional. Rio de Janeiro – RJ. 122 p. Dissertação de Mestrado, COPPE, Universidade Federal do Rio de Janeiro, 2018.

BRUNNER, G. W. HEC-RAS, River Analysis System Hydraulic Reference Manual. US Army Corps Of Engineers, 2016. Disponível em: <http://www.hec.usace.army.mil/software/hec-ras/documentation/HEC-RAS%205.0%20Reference%20Manual.pdf>. Acesso em: 13 mai 2018.

BURDEN R. L.; FAIRES J. D. Numerical Analysis. 9 ed. CENGAGE Learning, 2011. 877p.

CANNATA, M.; MARZOCCHI, R. Two-dimensional dam break flooding simulation: a GIS embedded approach. **Nat Hazards**. v. 61, p. 1143-1159, 2012.
<https://doi.org/10.1007/s11069-011-9974-6>

CARVALHO, A. C. B. Trcas de sal entre a laguna de Araruama e o mar, utilizando um modelo hidrodinâmico (SiSBaHiA). Niterói – RJ. 129 p. Dissertação de Mestrado, Universidade Federal Fluminense, 2018.
<https://app.uff.br/riuff/handle/1/6046>

CHAUDHRY, M. H. Open Channel Flow. 2 ed. Columbia: Springer, 2008. 523 p.

DANG, T. A.; TRAN, T. H. Application of the Mike21C model to simulate flow in the lower Mekong river basin. **SpringerPlus**. v.5, 2016.
<https://doi.org/10.1186/s40064-016-3637-8>

DHI. MIKE 21C River hydraulics and morphology. [201-?]. Disponível em: <https://www.mikepoweredbydhi.com/-/media/shared%20content/mike%20by%20dhi/flyers%20and%20pdf/software%20flyers/water%20resources/mike21c-mikepoweredbydhi-productflyer-uk.pdf>. Acesso em: 2 out 2018.

DUMKWU, F. A.; ISLAM A. W.; CARLSON E. S. Review of well models and assessment of their impacts on numerical reservoir simulation performance. **Journal of Petroleum Science and Engineering**. v. 82, p.174-186, 2012.
<https://doi.org/10.1016/j.petrol.2011.12.005>

EL KACIMI, A. et al. Bernstein - Bézier based finite elements for efficient solution of short wave problems. **Comput. Methods Appl. Mech. Engrg.** v. 343, p. 166-185, 2019.
<https://doi.org/10.1016/j.cma.2018.07.040>

ERDURAN, K. S. Fourth order finite volume solution to shallow water equations and applications. **INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS.** v. 73, p. 637-659, 2013.
<https://doi.org/10.1002/fld.3816>

GAMVROUDIS, C. et al. Water and sediment transport modeling of a large temporary river basin in Greece. **Science of the Total Environment.** v. 508, p. 354-365, 2015.
<http://doi.org/10.1016/j.scitotenv.2014.12.005>

GOOGLE. **Google Maps.** 2018. Disponível em: <https://www.google.com.br/maps>. Acesso em: 20 out 2018.

GRILLI, S. T. et al. Modeling of SMF tsunami hazard along the upper US East Coast: detailed impact around Ocean City, MD. **Nat Hazards.** v. 76, p. 705-746, 2015.
<https://doi.org/10.1007/s11069-014-1522-8>

HE, Z. et al. Numerical simulation of dam-break flow and bed change considering the vegetation effects. **International Journal of Sediment Research.** v. 32, p. 105-120.
<https://doi.org/10.1016/j.ijsrc.2015.04.004>

HU, C.; SUEYOSHI, M. Numerical simulation and experiment on dam break problem. **Journal of Marine Science and Application.** v. 9, n. 2, p. 109-114, 2010.
<https://doi.org/10.1007/s11804-010-9075-z>

KAO, H. M.; CHANG, T. J. Numerical modeling of dambreak-induced flood and inundation using smoothed particle hydrodynamics. **Journal of Hydrology.** v. 448, p. 232-244, 2012.
<https://doi.org/10.1016/j.jhydrol.2012.05.004>

KESHTPOOR, M. et al. 3D numerical simulation of turbulence and sediment transport within a tidal inlet. **Coastal Engineering.** v. 96, p.13-26, 2015.
<http://doi.org/10.1016/j.coastaleng.2014.10.009>

KILINC, I.; HAYIR, A.; CIGIZOGLU, H. K. Wave dispersion study for tsunami propagation in the Sea of Marmara. **Coastal Engineering.** v. 56, n. 9, p. 982-991, 2009.
<https://doi.org/10.1016/j.coastaleng.2009.06.006>

KWAN, S. A two dimensional hydrodynamic river morphology and gravel transport model. Vancouver. 113 p. Master's Degree Dissertation, Department of Civil Engineering, University of British Columbia, 2009.
<https://doi.org/10.14288/1.0063132>

LI, P.; FAN, C. Generalized finite difference method for two-dimensional shallow water equations. **Engineering Analysis With Boundary Elements**. v. 80, p.58-71, 2017.
<https://doi.org/10.1016/j.enganabound.2017.03.012>

MARTINS, C. S. Influence of land use and occupation on the silting of Bom Jardim Reservoir, Uberlândia – MG. 155 p. Master's Degree Dissertation, School of Civil Engineering, Federal University of Uberlândia, 2018.
<http://dx.doi.org/10.14393/ufu.di.2018.1122>

MORIANOU, G. G. et al. Hydraulic and sediment transport simulation of Koiliaris River using the MIKE 21C model. **Procedia Engineering**. v. 162, p. 463-470, 2016.
<https://doi.org/10.1016/j.proeng.2016.11.089>

OÑATE, E. et al. A stabilized finite point method for analysis of fluid mechanics problems. **Comput. Methods Appl. Mech. Engrg.** v.139, p.315-346, 1996.
[https://doi.org/10.1016/S0045-7825\(96\)01088-2](https://doi.org/10.1016/S0045-7825(96)01088-2)

ORTEGA, E. An adaptive finite point method for the shallow water equations. **International Journal for Numerical Methods in Engineering**. v.88, p.180-204, 2011.
<https://doi.org/10.1002/nme.3171>

PECINI, B. **Aplicação do modelo river 2D em estudos de desvios de rios.** Campinas – SP. 105p. Dissertação de Mestrado, Faculdade de Engenharia Civil, Arquitetura e Urbanismo, Universidade Estadual de Campinas, 2016.
<http://repositorio.unicamp.br/jspui/handle/REPOSIP/305380>

PORTE, R. M. **Hidráulica Básica.** 4 ed. São Carlos: EESC-USP, 2006. 519p.
 REZZOLA, L. **Numerical Methods for the Solution of Partial Differential Equations.** Albert Einstein Institute. 82 p. 2011. Disponível em:
http://www.aei.mpg.de/~rezzolla/lnotes/Evolution_Pdes/evolution_pdes_lnotes.pdf. Acesso em: 15 jan 2018.

SAMPAIO, M. **Manual do Usuário do SisBaHiA.** Coordenação de Programas de Pós Graduação em Engenharia, Universidade Federal do Rio de Janeiro, 2017. Disponível em:
<http://www.sisbahia.coppe.ufrj.br/ManualSisBAHIA/A1Index.htm>. Acesso em: 25 set 2018.

SATRIA, M. T. et al. GPU Acceleration of Tsunami Propagation Model. **IEEE journal of selected topics in applied earth observations and remote sensing**. v. 5, n. 3, p. 1014-1023, 2012.

<https://doi.org/10.1109/JSTARS.2012.2199468>

SINGH, J.; ALTINAKAR, M. S.; DING, Y. Two-dimensional numerical modeling of dam-break flows over natural terrain using a central explicit scheme. **Advances in Water Resources**. v. 34, p. 1366-1375, 2011.

<https://doi.org/10.1016/j.advwatres.2011.07.007>

STEFFLER, P.; BLACKBURN, J. **River2D, Introduction to Depth Averaged Modeling and User's Manual**. University of Alberta, 2002. Disponível em: <http://www.river2d.ualberta.ca/Downloads/documentation/River2D.pdf>. Acesso em: 24 set 2018.

SUN C. P. et al. Application of localized meshless methods to 2D shallow water equation problems. **Engineering Analysis with Boundary Elements**. v. 37, p. 1339-1350, 2013.

<https://doi.org/10.1016/j.enganabound.2013.06.010>

TIWARI, S.; KLAR, A.; HARDT, S. Numerical simulation of wetting phenomena by a meshfree particle method. **Journal of Computational and Applied Mathematics**. v. 292, p. 469-485, 2016.

<https://doi.org/10.1016/j.cam.2015.07.021>

USACE. **HEC-RAS**. US Army Corps Of Engineers, [2016?]. Disponível em: <http://www.hec.usace.army.mil/software/hec-ras/>. Acesso em: 25 set 2018.

VACONDIO, R. Simulation of the January 2014 flood on the Secchia River using a fast and high-resolution 2D parallel shallow-water numerical scheme. **Nat Hazards**. v. 80, p. 103-125, 2016.

<https://doi.org/10.1007/s11069-015-1959-4>

VIANINI NETO, L. **Estudo de ruptura da barragem da Pampulha, em Belo Horizonte: Retroanálise da brecha do acidente de 1953 e ruptura hipotética nas condições atuais**. Belo Horizonte – MG. 303 p. Dissertação de Mestrado, Escola de Engenharia, Universidade Federal de Minas Gerais, 2016.

<http://hdl.handle.net/1843/BUBD-AUAGVF>

YAKTI, B. P. et al. 2D Modeling of Flood Propagation due to te Failure of Way Ela Natural Dam. **MATEC Web of Conferences**. v. 147, 2018.

<https://doi.org/10.1051/matecconf/201814703009>

ZHENDONG, L.; JUNQIANG, G. The numerical simulations based on the NND finite difference scheme for shallow water wave equations including sediment concentration. **Comput. Methods Appl. Mech. Engrg.** v. 294, p. 245-258, 2015.
<https://doi.org/10.1016/j.jmaa.2015.04.024>

ZHENDONG, L.; JUNQIANG, G.; ZHENGHUI, X. Reduced-order finite difference extrapolation model based on proper orthogonal decomposition for two-dimensional shallow water equations including sediment concentration. **Journal of Mathematical Analysis and Applications.** v. 429, p. 901-923, 2015.
<https://doi.org/10.1016/j.jmaa.2015.06.011>

ZHOU, Z. Numerical simulation of dam-break flooding of Cascade reservoirs. **Trans. Tianjin Univ.** v. 23, p.570-581.
<https://doi.org/10.1007/s12209-017-0073-y>

APÊNDICE A

ROTINA DE CÁLCULOS COMPLETA PARA A SIMULAÇÃO DO RESERVATÓRIO BOM JARDIM

Sub Calcular()

```

Dim z(), h(), u(), v(), w() As Single
Dim ui(), vj() As Single
Dim hf(), uif(), vjf() As Single
Dim deltat, deltax, deltay As Single
Dim deltax2, deltay2 As Single
Dim tinicial, tfinal, t As Single
Dim n, g As Single
Dim passo As Single
Dim celx, cely As Single
Dim Au, Av As Single
Dim Dev As Single
Dim Ru, Rv As Single
Dim cont As Single
Dim gamma As Single
Dim hc, his1, his2, hii1, hii2, hji1, hji2, hjs1, hjs2 As Single
Dim hcis1, hcii1, hcjs1, hcji1 As Single
Dim uc, uis1, uis3, uii1, uis1js2, uis1ji2, uis2, uis1js1, uis1ji1, ujs1 As Single
Dim vc, vjs1, vji1, vis1, vis2js1, vii2js1, vjs3, vis1js1, vii1js1, vjs2 As Single
Dim zc, zis2, zjs2, zji2, zii2, zis2js2, zii2js2 As Single
Dim hcvf, uivf, vjvf As Single
Dim hm As Single
Dim fatumin, fatvmin, fatmin As Single
Dim P(3), D(3), E(3), F(3), K(3), S(3) As Single
Dim dfv1, dfv2, dfv3, dfv4 As Single
Dim dfv1f, dfv2f, dfv3f, dfv4f As Single
Dim dfu1, dfu2, dfu3, dfu4 As Single
Dim dfu1f, dfu2f, dfu3f, dfu4f As Single
Dim pularui As Boolean
Dim pularvj As Boolean
Dim pii, pis, pji, pjs As Integer

```

g = 9.81 '[m/s]

```

celx = Sheets("Config").Cells(3, 2).Value
cely = Sheets("Config").Cells(4, 2).Value
xmin = Sheets("Config").Cells(3, 5).Value
ymin = Sheets("Config").Cells(4, 5).Value
xmax = Sheets("Config").Cells(3, 8).Value
ymax = Sheets("Config").Cells(4, 8).Value
deltax = Sheets("Config").Cells(8, 2).Value
deltay = Sheets("Config").Cells(9, 2).Value
tinicial = Sheets("Config").Cells(8, 5).Value
tfinal = Sheets("Config").Cells(9, 5).Value
plotn = Sheets("Config").Cells(11, 6).Value
n = Sheets("Config").Cells(13, 6).Value
c = Sheets("Config").Cells(12, 2).Value
tipo = Sheets("Config").Cells(15, 5).Value
Dev = Sheets("Config").Cells(13, 2).Value

```

```

gamma = Sheets("Config").Cells(14, 2).Value
pii = Sheets("Config").Cells(17, 9).Value
pis = Sheets("Config").Cells(17, 13).Value
pji = Sheets("Config").Cells(19, 11).Value
pjs = Sheets("Config").Cells(15, 11).Value

deltax2 = deltax * deltax
deltay2 = deltay * deltay

ReDim z(celx, cely)
ReDim h(celx, cely)
ReDim v(celx, cely)
ReDim vj(celx, cely + 1)
ReDim u(celx, cely)
ReDim ui(celx + 1, cely)
ReDim hf(celx, cely)
ReDim uif(celx + 1, cely)
ReDim vjf(celx, cely + 1)
ReDim w(celx, cely)

Sheets("Entrada").Activate

cont = 1
For i = 1 To celx
    For j = 1 To cely
        z(i, j) = Sheets("Entrada").Cells(cont + 1, 3).Value
        h(i, j) = Sheets("Entrada").Cells(cont + 1, 4).Value
        u(i, j) = Sheets("Entrada").Cells(cont + 1, 5).Value
        v(i, j) = Sheets("Entrada").Cells(cont + 1, 6).Value
        w(i, j) = Sheets("Entrada").Cells(cont + 1, 7).Value
        cont = cont + 1
    Next j
Next i

For i = 2 To celx
    For j = 1 To cely
        ui(i, j) = (u(i, j) + u(i - 1, j)) / 2
    Next j
Next i

For j = 1 To cely
    ui(1, j) = 0
    ui(celx + 1, j) = 0
Next j

For i = 1 To celx
    For j = 2 To cely
        vj(i, j) = (v(i, j) + v(i, j - 1)) / 2
    Next j
Next i

For i = 1 To celx
    vj(i, 1) = 0
    vj(i, cely + 1) = 0
Next i

Sheets("Saida").Activate

passo = 1
t = tinicial

```

```

Do
    Sheets("Saida").Cells(3, 13).Value = passo
    Sheets("Saida").Cells(4, 13).Value = t
    Sheets("Saida").Cells(5, 13).Value = deltat

    fatmin = 10000
    If (passo < 5) Then
        deltat = 0.01
    Else
        For i = 1 To (celx - 1)
            For j = 1 To (cely - 1)
                If (h(i, j) < 0) Then
                    If (fatmin > deltax * deltay / ((Abs(ui(i + 1, j)) + (g * h(i, j)) ^ 0.5) * deltay + (Abs(vj(i, j + 1)) + (g * h(i, j)) ^ 0.5) * deltax)) Then
                        fatmin = deltax * deltay / ((Abs(ui(i + 1, j)) + (g * h(i, j)) ^ 0.5) * deltay + (Abs(vj(i, j + 1)) + (g * h(i, j)) ^ 0.5) * deltax)
                    End If
                End If
                Next j
            Next i
            deltat = fatmin * c
        End If

        For i = 1 To celx
            For j = 1 To cely
                pularui = False
                pularvj = False

                zc = z(i, j)
                hc = h(i, j)
                uc = (ui(i, j) + ui(i + 1, j)) / 2
                vc = (vj(i, j) + vj(i, j + 1)) / 2

                If (i = 110 And j = 178) Then
                    h(i, j) = 0.4
                    ui(i + 1, j) = 0
                    vj(i, j + 1) = 0
                    hc = 0.4
                    uivf = 0
                    vjvf = 0
                    GoTo 3
                ElseIf (i = 109 And j = 178) Then
                    ui(i + 1, j) = -1
                    uivf = -1
                    pularui = True
                ElseIf (i = 110 And j = 177) Then
                    hc = 0
                    uivf = 0
                    vjvf = 0
                    GoTo 3
                End If

                If (i = 109 And j = 219) Then
                    h(i, j) = 0.4
                    ui(i + 1, j) = 0
                    vj(i, j + 1) = 0
                    hc = 0.4
                    uivf = 0
                    vjvf = 0
                End If
            Next j
        Next i
    End If
End Sub

```

```

GoTo 3
ElseIf (i = 108 And j = 219) Then
    ui(i + 1, j) = -0.25
    uivf = -0.25
    pularui = True
ElseIf (i = 109 And j = 218) Then
    hcvf = 0
    uivf = 0
    vjvf = 0
    GoTo 3
End If

If (i = 51 And j = 437) Then
    hcvf = 0
    uivf = 0
    vjvf = 0
    GoTo 3
ElseIf (i = 51 And j = 436) Then
    If hc < 1 Then
        vj(i, j + 1) = 0
        vjvf = 0
        pularvj = True
    Else
        vj(i, j + 1) = 3.5 / (10 * hc)
        vjvf = 3.5 / (10 * hc)
        pularvj = True
    End If
End If

If j = 1 Then
    If pji = 2 Then
        hji1 = h(i, j)
        hji2 = h(i, j)
        uis1ji1 = ui(i + 1, j)
        uis1ji2 = ui(i + 1, j)
        vji1 = vj(i, j + 1)
        zji2 = z(i, j)
        If i < celx Then
            zis2ji2 = z(i + 1, j)
        Else
            zis2ji2 = z(i, j)
        End If

    Else
        hji1 = h(i, j)
        hji2 = h(i, j)
        uis1ji1 = 0
        uis1ji2 = -ui(i + 1, j)
        vji1 = 0
        zji2 = z(i, j)
        If i < celx Then
            zis2ji2 = z(i + 1, j)
        Else
            zis2ji2 = z(i, j)
        End If
    End If

Else
    If vj(i, j) > 0 Then
        hji1 = h(i, j - 1)

```

```

ElseIf vj(i,j) < 0 Then
    hji1 = h(i,j)
ElseIf h(i,j) > h(i,j - 1) Then
    hji1 = h(i,j)
Else
    hji1 = h(i,j - 1)
End If
hji2 = h(i,j - 1)
uis1ji1 = (ui(i + 1,j) + ui(i + 1,j - 1)) / 2
uis1ji2 = ui(i + 1,j - 1)
vji1 = vj(i,j)
zji2 = z(i,j - 1)
If i < celx Then
    zis2ji2 = z(i + 1,j - 1)
Else
    zis2ji2 = z(i,j - 1)
End If
End If

If i = 1 Then
    If pii = 2 Then
        hii1 = h(i,j)
        hii2 = h(i,j)
        uii1 = ui(i + 1,j)
        vii1js1 = vj(i,j + 1)
        vii2js1 = vj(i,j + 1)
        zii2 = z(i,j)
        If j < cely Then
            zii2js2 = z(i,j + 1)
        Else
            zii2js2 = z(i,j)
        End If

        Else
            hii1 = h(i,j)
            hii2 = h(i,j)
            uii1 = 0
            vii1js1 = 0
            vii2js1 = -vj(i,j + 1)
            zii2 = z(i,j)
            If j < cely Then
                zii2js2 = z(i,j + 1)
            Else
                zii2js2 = z(i,j)
            End If
        End If

    Else
        If ui(i,j) > 0 Then
            hii1 = h(i - 1,j)
        ElseIf ui(i,j) < 0 Then
            hii1 = h(i,j)
        ElseIf h(i,j) > h(i - 1,j) Then
            hii1 = h(i,j)
        Else
            hii1 = h(i - 1,j)
        End If
        hii2 = h(i - 1,j)
        uii1 = ui(i,j)
        vii1js1 = (vj(i - 1,j + 1) + vj(i,j + 1)) / 2

```

```

vii2js1 = vj(i - 1, j + 1)
zii2 = z(i - 1, j)
If j < cely Then
    zii2js2 = z(i - 1, j + 1)
Else
    zii2js2 = z(i - 1, j)
End If
End If

If j = cely Then
    If pjs = 2 Then
        hjs2 = h(i, j)
        vjs1 = vj(i, j)
        hjs1 = h(i, j)
        uis1js2 = ui(i + 1, j)
        uis1js1 = ui(i + 1, j)
        zjs2 = z(i, j)
        vjs3 = vj(i, j)
        ujs1 = uc
        vjs2 = (vj(i, j) / 2)
        If i < celx Then
            zis2js2 = z(i + 1, j)
        Else
            zis2js2 = z(i, j)
        End If
        pularvj = True
    Else
        hjs2 = h(i, j)
        vjs1 = 0
        hjs1 = h(i, j)
        uis1js2 = -ui(i + 1, j)
        uis1js1 = 0
        zjs2 = z(i, j)
        vjs3 = -vj(i, j)
        ujs1 = 0
        vjs2 = -(vj(i, j) / 2)
        If i < celx Then
            zis2js2 = z(i + 1, j)
        Else
            zis2js2 = z(i, j)
        End If
        pularvj = True
    End If
Else
    If vj(i, j + 1) > 0 Then
        hjs1 = h(i, j)
    ElseIf vj(i, j + 1) < 0 Then
        hjs1 = h(i, j + 1)
    ElseIf h(i, j) >= h(i, j + 1) Then
        hjs1 = h(i, j)
    Else
        hjs1 = h(i, j + 1)
    End If
    hjs2 = h(i, j + 1)
    vjs1 = vj(i, j + 1)
    uis1js2 = ui(i + 1, j + 1)
    uis1js1 = (ui(i + 1, j) + ui(i + 1, j + 1)) / 2
    zjs2 = z(i, j + 1)
    vjs3 = vj(i, j + 2)
    ujs1 = (ui(i, j) + ui(i + 1, j) + ui(i, j + 1) + ui(i + 1, j + 1)) / 4

```

```

vjs2 = (vj(i, j + 1) + vj(i, j + 2)) / 2
If i < celx Then
    zis2js2 = z(i + 1, j + 1)
Else
    zis2js2 = z(i, j + 1)
End If
End If

If i = celx Then
    If pis = 2 Then
        uis1 = ui(i, j)
        his1 = h(i, j)
        his2 = h(i, j)
        vis2js1 = vj(i, j + 1)
        vis1js1 = vj(i, j + 1)
        zis2 = z(i, j)
        uis3 = ui(i, j)
        uis2 = ui(i, j)
        vis1 = vc
        pularui = True
    Else
        uis1 = 0
        his1 = h(i, j)
        his2 = h(i, j)
        vis2js1 = -vj(i, j + 1)
        vis1js1 = 0
        zis2 = z(i, j)
        uis3 = -ui(i, j)
        uis2 = -(ui(i, j) / 2)
        vis1 = 0
        pularui = True
    End If
End If

Else
    If ui(i + 1, j) > 0 Then
        his1 = h(i, j)
    ElseIf ui(i + 1, j) < 0 Then
        his1 = h(i + 1, j)
    ElseIf h(i, j) >= h(i + 1, j) Then
        his1 = h(i, j)
    Else
        his1 = h(i + 1, j)
    End If
    uis1 = ui(i + 1, j)
    his2 = h(i + 1, j)
    vis2js1 = vj(i + 1, j + 1)
    vis1js1 = (vj(i, j + 1) + vj(i + 1, j + 1)) / 2
    zis2 = z(i + 1, j)
    uis3 = ui(i + 2, j)
    uis2 = (ui(i + 1, j) + ui(i + 2, j)) / 2
    vis1 = (vj(i, j) + vj(i + 1, j) + vj(i, j + 1) + vj(i + 1, j + 1)) / 4
End If

If (hc = 0) And ((zc >= hii2 + zii2) Or (hii2 = 0)) And ((zc >= his2 + zis2) Or (his2 = 0)) And ((zc >= hji2 + zji2) Or (hji2 = 0)) And ((zc >= hjs2 + zjs2) Or (hjs2 = 0)) Then
    hcvf = 0
    uivf = 0
    vjvf = 0
    GoTo 3

```

End If

If ($hc = 0$) And (($his2 = 0$) Or ($his2 + zis2 \leq zc$)) Then
 pularui = True
 End If

If ($hc = 0$) And (($hjs2 = 0$) Or ($hjs2 + zjs2 \leq zc$)) Then
 pularvji = True
 End If

If $j > 1$ Then
 If ($h(i, j - 1) = 0$) And ($hc + zc \leq z(i, j - 1)$) Then
 hji1 = $h(i, j)$
 hji2 = $h(i, j)$
 uis1ji1 = 0
 uis1ji2 = $-ui(i + 1, j)$
 vji1 = 0
 End If
 End If

If $i > 1$ Then
 If ($h(i - 1, j) = 0$) And ($hc + zc \leq z(i - 1, j)$) Then
 hii1 = $h(i, j)$
 hii2 = $h(i, j)$
 uii1 = 0
 vii1js1 = 0
 vii2js1 = $-vj(i, j + 1)$
 End If
 End If

If $j < cely$ Then
 If ($h(i, j + 1) = 0$) And ($hc + zc \leq z(i, j + 1)$) Then
 hjs2 = $h(i, j)$
 vjs1 = 0
 hjs1 = $h(i, j)$
 uis1js2 = $-ui(i + 1, j)$
 uis1js1 = 0
 vjs3 = $-vj(i, j)$
 ujs1 = 0
 vjs2 = $-(vj(i, j) / 2)$
 pularvji = True
 End If
 End If

If $i < celx$ Then
 If ($h(i + 1, j) = 0$) And ($hc + zc \leq z(i + 1, j)$) Then
 uis1 = 0
 his1 = $h(i, j)$
 his2 = $h(i, j)$
 vis2js1 = $-vj(i, j + 1)$
 vis1js1 = 0
 uis3 = $-ui(i, j)$
 uis2 = $-(ui(i, j) / 2)$
 vis1 = 0
 pularui = True
 End If
 End If

If $zis2 > zc$ And $uis1 > 0$ Then
 hcis1 = his1 - ($zis2 - zc$)

```

ElseIf zc > zis2 And uis1 < 0 Then
    hcis1 = his1 - (zc - zis2)
Else
    hcis1 = his1
End If

If zii2 > zc And uii1 < 0 Then
    hcii1 = hii1 - (zii2 - zc)
ElseIf zc > zii2 And uii1 > 0 Then
    hcii1 = hii1 - (zc - zii2)
Else
    hcii1 = hii1
End If

If zjs2 > zc And vjs1 > 0 Then
    hcjs1 = hjs1 - (zjs2 - zc)
ElseIf zc > zjs2 And vjs1 < 0 Then
    hcjs1 = hjs1 - (zc - zjs2)
Else
    hcjs1 = hjs1
End If

If zji2 > zc And vji1 < 0 Then
    hcji1 = hji1 - (zji2 - zc)
ElseIf zc > zji2 And vji1 > 0 Then
    hcji1 = hji1 - (zc - zji2)
Else
    hcji1 = hji1
End If

If h(i, j) > 0 Then
    If i < celx Then
        If h(i + 1, j) > 0 And i < celx Then
            dfh1 = h(i + 1, j) + z(i + 1, j) - h(i, j) - z(i, j)
        Else
            dfh1 = 0
        End If
    Else
        dfh1 = 0
    End If

    If i > 1 Then
        If h(i - 1, j) > 0 Then
            dfh2 = h(i - 1, j) + z(i - 1, j) - h(i, j) - z(i, j)
        Else
            dfh2 = 0
        End If
    Else
        dfh2 = 0
    End If

    If j < cely Then
        If h(i, j + 1) > 0 Then
            dfh3 = h(i, j + 1) + z(i, j + 1) - h(i, j) - z(i, j)
        Else
            dfh3 = 0
        End If
    Else
        dfh3 = 0
    End If

```

```

End If

If j > 1 Then
  If h(i, j - 1) > 0 And j > 1 Then
    dfh4 = h(i, j - 1) + z(i, j - 1) - h(i, j) - z(i, j)
  Else
    dfh4 = 0
  End If
Else
  dfh4 = 0
End If

D(1) = gamma * ((dfh1 + dfh2) / deltax2 + (dfh3 + dfh4) / deltay2)
Else
  D(1) = 0
End If

E(1) = -((uis1 * hcis1) - (uii1 * hcii1)) / deltax
F(1) = -((vjs1 * hcjs1) - (vji1 * hcji1)) / deltay

K(1) = 0
S(1) = 0

If i > 1 And i < celx And j > 1 And j < cely Then
  D(1) = 0
  TT = 0
  If (h(i + 1, j) > 0 And (h(i + 1, j) + z(i + 1, j)) > z(i, j)) Then
    T1 = h(i + 1, j) + z(i + 1, j) - z(i, j)
    TT = TT + 1
  Else
    T1 = 0
  End If

  If (h(i - 1, j) > 0 And (h(i - 1, j) + z(i - 1, j)) > z(i, j)) Then
    T2 = h(i - 1, j) + z(i - 1, j) - z(i, j)
    TT = TT + 1
  Else
    T2 = 0
  End If

  If (h(i, j + 1) > 0 And (h(i, j + 1) + z(i, j + 1)) > z(i, j)) Then
    T3 = h(i, j + 1) + z(i, j + 1) - z(i, j)
    TT = TT + 1
  Else
    T3 = 0
  End If

  If (h(i, j - 1) > 0 And (h(i, j - 1) + z(i, j - 1)) > z(i, j)) Then
    T4 = h(i, j - 1) + z(i, j - 1) - z(i, j)
    TT = TT + 1
  Else
    T4 = 0
  End If

  hm = (T1 + T2 + T3 + T4 + hc * 4) / (TT + 4)

Else
  hm = hc
End If

```

```

P(1) = D(1) + E(1) + F(1) + K(1) + S(1)

hcvf = P(1) * deltat + hm

If (hcvf < 0.00001) Then
    hcvf = 0
End If

If pularui = True Then
    If pis = 2 And i = celx Then
        uivf = ui(i, j)
        GoTo 1
    ElseIf (i = 109 And j = 178) Then
        ui(i + 1, j) = -1
        uivf = -1
        GoTo 1
    ElseIf (i = 108 And j = 219) Then
        ui(i + 1, j) = -0.25
        uivf = -0.25
        GoTo 1
    Else
        uivf = 0
        GoTo 1
    End If

Else
    dfu1 = zjs2 - zc
    If (dfu1 > 0) And (dfu1 <= hc) Then
        dfu1f = dfu1
    ElseIf (dfu1 > 0) And (dfu1 > hc) Then
        dfu1f = hc
    Else
        dfu1f = 0
    End If

    dfu2 = zis2js2 - zis2
    If (dfu2 > 0) And (dfu2 <= his2) Then
        dfu2f = dfu2
    ElseIf (dfu2 > 0) And (dfu2 > his2) Then
        dfu2f = his2
    Else
        dfu2f = 0
    End If

    dfu3 = zji2 - zc
    If (dfu3 > 0) And (dfu3 <= hc) Then
        dfu3f = dfu3
    ElseIf (dfu3 > 0) And (dfu3 > hc) Then
        dfu3f = hc
    Else
        dfu3f = 0
    End If

    dfu4 = zis2ji2 - zis2
    If (dfu4 > 0) And (dfu4 <= his2) Then
        dfu4f = dfu4
    ElseIf (dfu4 > 0) And (dfu4 > his2) Then
        dfu4f = his2
    Else

```

```

        dfu4f = 0
    End If

    If uis1 >= 0 Then
        Ru = deltax * his1 / (deltay + dfu1f + dfu3f)
    Else
        Ru = deltax * his1 / (deltay + dfu2f + dfu4f)
    End If

    If Ru = 0 Then
        uivf = 0
        GoTo 1
    Else
        Au = Dev * his1 * n * (g ^ 0.5) * ((uis1 * uis1 + vis1 * vis1) ^ 0.5) / (Ru ^ (1 / 6))
    End If

    If uis1 = 0 Then
        D(2) = 0
    Else
        D(2) = Au * ((uis3 - 2 * uis1 + uii1) / deltax2 + (uis1js2 - 2 * uis1 + uis1ji2) / deltax2)
    End If

    E(2) = -(uis1 * (uis2 - uc) / deltax)
    F(2) = -(vis1 * (uis1js1 - uis1ji1) / deltax)
    K(2) = -g * ((his2 + zis2 - hc - xc) / deltax)
    S(2) = -(n * n * g * uis1 * ((uis1 * uis1 + vis1 * vis1) ^ 0.5)) / (Ru ^ (4 / 3))

    P(2) = D(2) + E(2) + F(2) + K(2) + S(2)
    uivf = P(2) * deltat + uis1

    If hcvf = 0 And uivf > 0 Then
        uivf = 0
        GoTo 1
    End If

    If his2 = 0 And uivf < 0 Then
        uivf = 0
        GoTo 1
    End If

    End If
1

    If pularvj = True Then
        If pjs = 2 And j = cely Then
            vjvf = vj(i, j)
            GoTo 2
        ElseIf (i = 51 And j = 436) Then
            If hc < 1 Then
                vj(i, j + 1) = 0
                vjvf = 0
                GoTo 2
            Else
                vj(i, j + 1) = 3.5 / (10 * hc)
                vjvf = 3.5 / (10 * hc)
                GoTo 2
            End If
        Else
            vjvf = 0
            GoTo 2
        End If
    End If

```

```

End If
Else
  dfv1 = zii2 - zc
  If (dfv1 > 0) And (dfv1 <= hc) Then
    dfv1f = dfv1
  ElseIf (dfv1 > 0) And (dfv1 > hc) Then
    dfv1f = hc
  Else
    dfv1f = 0
  End If

  dfv2 = zii2js2 - zjs2
  If (dfv2 > 0) And (dfv2 <= hjs2) Then
    dfv2f = dfv2
  ElseIf (dfv2 > 0) And (dfv2 > hjs2) Then
    dfv2f = hjs2
  Else
    dfv2f = 0
  End If

  dfv3 = zis2js2 - zjs2
  If (dfv3 > 0) And (dfv3 <= hjs2) Then
    dfv3f = dfv3
  ElseIf (dfv3 > 0) And (dfv3 > hjs2) Then
    dfv3f = hjs2
  Else
    dfv3f = 0
  End If

  dfv4 = zis2 - zc
  If (dfv4 > 0) And (dfv4 <= hc) Then
    dfv4f = dfv4
  ElseIf (dfv4 > 0) And (dfv4 > hc) Then
    dfv4f = hc
  Else
    dfv4f = 0
  End If

  If vjs1 >= 0 Then
    Rv = deltax * hjs1 / (deltax + dfv1f + dfv4f)
  Else
    Rv = deltax * hjs1 / (deltax + dfv2f + dfv3f)
  End If

  If Rv = 0 Then
    vjvf = 0
    GoTo 2
  Else
    Av = Dev * hjs1 * n * (g ^ 0.5) * ((ujs1 * ujs1 + vjs1 * vjs1) ^ 0.5) / (Rv ^ (1 / 6))
  End If

  If vjs1 = 0 Then
    D(3) = 0
  Else
    D(3) = Av * ((vis2js1 - 2 * yjs1 + vii2js1) / deltax2 + (vjs3 - 2 * vjs1 + vji1) / deltay2)
  End If
  E(3) = -(ujs1 * (vis1js1 - vii1js1) / deltax)
  F(3) = -(vjs1 * (vjs2 - vc) / deltay)
  K(3) = -g * (hjs2 + zjs2 - hc - zc) / deltay
  S(3) = -(n * n * g * vjs1 * ((ujs1 * ujs1 + vjs1 * vjs1) ^ 0.5)) / (Rv ^ (4 / 3))

```

```

P(3) = D(3) + E(3) + F(3) + K(3) + S(3)
vjvf = P(3) * deltat + vjs1

If hcvf = 0 And vjvf > 0 Then
    vjvf = 0
    GoTo 2
End If

If hjs2 = 0 And vjvf < 0 Then
    vjvf = 0
    GoTo 2
End If

End If

2
3
hf(i, j) = hcvf
uif(i + 1, j) = uivf
vjf(i, j + 1) = vjvf
Next j
Next i

For i = 1 To celx
    For j = 1 To cely
        h(i, j) = hf(i, j)
        ui(i, j) = uif(i, j)
        vj(i, j) = vjf(i, j)
        w(i, j) = hf(i, j) + z(i, j)
    Next j
Next i

If (t > 50 And t < 50.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 1).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 2).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 3).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 4).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 5).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 6).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 7).Value = w(i, j)
            cont = cont + 1
        Next j
    Next i
End If

If (t > 150 And t < 150.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 10).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 11).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 12).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 13).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 14).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 15).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 16).Value = w(i, j)
            cont = cont + 1
        Next j
    Next i
End If

```

```

    Next j
    Next i
End If

If (t > 900 And t < 900.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 19).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 20).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 21).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 22).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 23).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 24).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 25).Value = w(i, j)
            cont = cont + 1
        Next j
    Next i
End If

If (t > 1800 And t < 1800.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 28).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 29).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 30).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 31).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 32).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 33).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 34).Value = w(i, j)
            cont = cont + 1
        Next j
    Next i
End If

If (t > 2250 And t < 2250.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 37).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 38).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 39).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 40).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 41).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 42).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 43).Value = w(i, j)
            cont = cont + 1
        Next j
    Next i
End If

If (t > 2700 And t < 2700.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 46).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 47).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 48).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 49).Value = h(i, j)

```

```

Sheets("Grafico").Cells(cont + 2, 50).Value = ui(i + 1, j)
Sheets("Grafico").Cells(cont + 2, 51).Value = vj(i, j + 1)
Sheets("Grafico").Cells(cont + 2, 52).Value = w(i, j)
cont = cont + 1
    Next j
    Next i
End If

If (t > 3150 And t < 3150.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 55).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 56).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 57).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 58).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 59).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 60).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 61).Value = w(i, j)
            cont = cont + 1
        Next j
        Next i
    End If

If (t > 3600 And t < 3600.3) Then
    cont = 1
    For i = 1 To celx
        For j = 1 To cely
            Sheets("Grafico").Cells(cont + 2, 64).Value = xmin + (i - 1) * deltax
            Sheets("Grafico").Cells(cont + 2, 65).Value = ymin + (j - 1) * deltay
            Sheets("Grafico").Cells(cont + 2, 66).Value = z(i, j)
            Sheets("Grafico").Cells(cont + 2, 67).Value = h(i, j)
            Sheets("Grafico").Cells(cont + 2, 68).Value = ui(i + 1, j)
            Sheets("Grafico").Cells(cont + 2, 69).Value = vj(i, j + 1)
            Sheets("Grafico").Cells(cont + 2, 70).Value = w(i, j)
            cont = cont + 1
        Next j
        Next i
    End If

DoEvents
t = t + deltat
passo = passo + 1

Loop Until (t > tfinal)

cont = 1
For i = 1 To celx
    For j = 1 To cely
        Sheets("Saida").Cells(cont + 1, 1).Value = xmin + deltax * (i - 1)
        Sheets("Saida").Cells(cont + 1, 2).Value = ymin + deltay * (j - 1)
        Sheets("Saida").Cells(cont + 1, 3).Value = z(i, j)
        Sheets("Saida").Cells(cont + 1, 4).Value = h(i, j)
        Sheets("Saida").Cells(cont + 1, 5).Value = ui(i + 1, j)
        Sheets("Saida").Cells(cont + 1, 6).Value = vj(i, j + 1)
        Sheets("Saida").Cells(cont + 1, 7).Value = w(i, j)
        cont = cont + 1
    Next j
    Next i

```

```
MsgBox ("Cálculos finalizados!")
```

```
End Sub
```