



UNIVERSIDADE ESTADUAL DE
CAMPINAS

Instituto de Matemática, Estatística e
Computação Científica

ANA CAROLINA VIEIRA

**UM ESTUDO SOBRE A MUDANÇA DA
QUALIDADE DA ÁGUA DE UM RIO E
POSSIBILIDADE DE TRATAMENTO:
ABORDAGEM UTILIZANDO EQUAÇÕES DE
DIFUSÃO-ADVECÇÃO**

Campinas

2016

Ana Carolina Vieira

**UM ESTUDO SOBRE A MUDANÇA DA QUALIDADE
DA ÁGUA DE UM RIO E POSSIBILIDADE DE
TRATAMENTO: ABORDAGEM UTILIZANDO
EQUAÇÕES DE DIFUSÃO-ADVECÇÃO**

Dissertação apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestra em Matemática Aplicada.

Orientador: João Frederico da Costa Azevedo Meyer

Coorientador: Aurelio Ribeiro Leite de Oliveira

O arquivo digital corresponde à versão final da Dissertação defendida pelo aluno Ana Carolina Vieira e orientada pelo Prof. Dr. João Frederico da Costa Azevedo Meyer.

Campinas

2016

Agência(s) de fomento e nº(s) de processo(s): CAPES

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

V673e Vieira, Ana Carolina, 1989-
Um estudo sobre a mudança da qualidade da água de um rio e possibilidade de tratamento : abordagem utilizando equações de difusão- advecção / Ana Carolina Vieira. – Campinas, SP : [s.n.], 2016.

Orientador: João Frederico da Costa Azevedo Meyer.

Coorientador: Aurelio Ribeiro Leite de Oliveira.

Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Poluição - Aspectos ambientais - Modelos matemáticos. 2. Equação de difusão-advecção-reação. 3. Diferenças finitas. 4. Crank-Nicolson, Método de. I. Meyer, João Frederico da Costa Azevedo, 1947-. II. Oliveira, Aurelio Ribeiro Leite de, 1962-. III. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: A study on quality change of water from a river and treatment possibility : approach using diffusion-advection equations

Palavras-chave em inglês:

Pollution - Environmental aspects - Mathematical models

Diffusion-advection-reaction equation

Finite differences

Crank-Nicolson method

Área de concentração: Matemática Aplicada

Títuloção: Mestra em Matemática Aplicada

Banca examinadora:

João Frederico da Costa Azevedo Meyer [Orientador]

Luciana Takata Gomes

Laecio Carvalho de Barros

Data de defesa: 11-03-2016

Programa de Pós-Graduação: Matemática Aplicada

Dissertação de Mestrado defendida em 11 de março de 2016 e aprovada

Pela Banca Examinadora composta pelos Profs. Drs.

Prof.(a). Dr(a). JOÃO FREDERICO DA COSTA AZEVEDO MEYER

Prof.(a). Dr(a). LUCIANA TAKATA GOMES

Prof.(a). Dr(a). LAECIO CARVALHO DE BARROS

A Ata da defesa com as respectivas assinaturas dos membros encontra-se no processo de vida acadêmica do aluno.

A minha Mãe Vera Lúcia, ao meu Pai Silvio Roberto, meu irmão Daniel Luiz... e a um grande amigo Diego Ferreira Gomes.

Agradecimentos

Primeiramente gostaria de agradecer a Deus pela grande oportunidade e força dada de iniciar e concluir este trabalho.

A meus Pais que me incentivaram nos momentos mais difíceis dessa jornada. Ao meu irmão Daniel e minha cunhada Taniane que ficava horas conversando sobre o trabalho e pelas palavras de incentivo, e ao meu lindo e amado sobrinho Emanuel, pelo carinho de alguém que ainda não entende porque temos que ficar longe de casa por tanto tempo. A minha Prima, amiga e Cumadre Juliana e a minha linda afilhada Maria Fernanda, que também não entende o que é estar longe, mas, minha princezinha, a madrinha tem um amor muito grande por você. A minha família toda que direta ou indiretamente estão apoiando de longe.

Aos meus amigos de Uberlândia, que são especiais e sem eles eu não estaria aqui em Campinas e não seria tão esforçada como sou hoje. Obrigada Fernando, Mario, Deive, Laura, Douglas, Maycon e Pedro.

Aos amigos que fiz aqui em Campinas, Diego, Chryslaine, Anna Lígia, Carol, Tatiana, José, Augusto (que não está mais aqui em Campinas, mas está no coração). E não menos importante ao Isaac, pelas brigas e puxões de orelha que ele me deu, pelos fins de semana que passamos na universidade para arrumar os erros dos algoritmos. Valeu Isaac!

Agradeço também ao enorme aprendizado que tive com o Joni, que em diversas ocasiões foi mais que um orientador e professor, ele foi um amigo. Joni, foi uma honra trabalhar com você.

Ao Aurelio que nos momentos decisivos estava ali para ajudar, acompanhar e orientar.

Por fim, agradeço a Luciana e ao Laecio pelas ótimas considerações que fizeram para melhor o trabalho.

Agradeço a Capes pelo apoio financeiro.

*“Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2)*

Resumo

Este trabalho utiliza o modelo clássico da Equação Diferencial Parcial de Difusão-Advecção com uma opção específica de condições iniciais e de contorno em domínio bidimensional. Tal equação tem sido utilizada para modelar problemas de espalhamento de petróleo e outros poluentes, e embora seja apresentada uma aplicação específica relacionada ao rio Jordão, em Minas Gerais, o modelo torna-se adequado a qualquer problema em situação análoga. Para a resolução da equação, diante da impossibilidade de encontrar sua solução analítica, a proposta deste trabalho é a da abordagem numérica do problema, com discretização espacial através do método de diferenças finitas centradas de segunda ordem e a discretização temporal através do método de Crank-Nicolson, também de segunda ordem. As condições de contorno foram definidas a partir da hipótese de que, havendo poluição no rio, parte desta poluição é absorvida pelas margens, supostas não rochosas. O objetivo deste estudo é construir uma ferramenta matemática que analise qual a localização mais eficiente para fazer o tratamento de esgoto. Se seria melhor junto à fonte poluidora ainda no córrego, ou já no rio Jordão, onde o córrego deságua no rio. Desta forma, pretende-se estabelecer, a partir da aplicação da equação de difusão-advecção-reação, de métodos numéricos e de algoritmo em ambiente Matlab, simulações e geração de cenários que favoreçam o estudo de impacto das fontes poluidoras às margens do rio, sua mitigação, bem como ações emergenciais relacionadas aos locais mais poluídos para iniciar um processo de contingência e limpeza. Com os resultados obtidos, podemos futuramente trabalhar com políticas públicas que possam auxiliar na melhoria do índice de IQA da água na região.

Palavras-chave: Poluição. Equação de Difusão-Advecção-Reação. Diferenças Finitas. Crank-Nicolson.

Abstract

This work uses a classical model of the Partial Differential Equation as a specific option for initial and boundary conditions in a bi-dimensional domain. This equation has been used to model problems of oil spreading and other pollutants. Although it is presented a specific application related to Jordão River, in Minas Gerais State, this model becomes adaptable to any problem in a similar situation. To solve the equation, facing the impossibility of finding its analytical solution, this work proposes a numerical approach for the problem, with spatial discretization through the centered finite difference of second order and the time discretization by the Crank-Nicolson method, also of second order. The boundary conditions were set on the assumption that, having pollution in the river, part of this is absorbed by the border, supposedly not made by rocks. The main goal of this work is to build a mathematical tool that analyzes where is more efficient do the sewer treatment: the pollutant source, through the bourn or where it ends, in Jordão River. So we intend to establish, from the application of the equation of diffusion-advection-reaction, numerical methods and a Matlab algorithm, simulation and generating scenarios which encourage the study of the impact of pollution sources along the river, its mitigation and emergency actions related to the most polluted places to start then a process of contingency and cleaning. With the results, we can work in the future with public policies that can help improve water IQA rate in that place.

Keywords: Pollution. Equation of Diffusion-Advection-Reaction. Finite Difference. Crank-Nicolson.

Lista de ilustrações

Figura 1 – Identificação do córrego Brejo Alegre (afluente do rio Jordão) e do rio Jordão (SALLA et al., 2013).	19
Figura 2 – Parâmetro de fósforo elevado do rio Jordão (COELHO, 2015).	20
Figura 3 – Índice de Qualidade da Água do rio Jordão do primeiro trimestre de 2015 e relação ao mesmo período de 2014 (COELHO, 2015).	21
Figura 4 – Domínio retangular Ω_1	24
Figura 5 – Domínio Ω_1	24
Figura 6 – Perfil parabólico retirado de (FORTUNA, 2000).	24
Figura 7 – Perfil parabólico do domínio Ω_1	25
Figura 8 – Domínio Ω_2	26
Figura 9 – Malha de passo h	26
Figura 10 – Malha do domínio Ω_1	26
Figura 11 – Malha do domínio Ω_2	27
Figura 12 – Parábolas	28
Figura 13 – Amplitude da Parábola	28
Figura 14 – Amplitude da Parábola	29
Figura 15 – Domínio Ω_1	37
Figura 16 – Contagem da malha.	37
Figura 17 – Pontos do interior da malha.	38
Figura 18 – Condições de Contorno.	41
Figura 19 – Pontos da fronteira acima.	41
Figura 20 – Pontos da fronteira à direita.	42
Figura 21 – Pontos da fronteira à esquerda.	44
Figura 22 – Pontos da fronteira inferior.	44
Figura 23 – Novo domínio Ω_2	47
Figura 24 – Discretização do domínio Ω_2	47
Figura 25 – Gráficos Referentes à dispersão de poluente em Ω_1 (365 dias) com 3 fontes onde o córrego Brejo Alegre deságua no rio Jordão e 3 fontes rio acima, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.	54
Figura 26 – Gráficos Referentes à dispersão de poluente em Ω_1 por mais 1 ano, no qual o antecessor é condição inicial para a próxima simulação, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.	54
Figura 27 – Gráficos Referentes à dispersão de poluente em Ω_2 , com uma fonte no córrego Brejo Alegre, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.	55

Figura 28 – Gráficos Referentes à dispersão de poluente em Ω_2 (365 dias), agora incluindo a dispersão no córrego Brejo Alegre com 3 fontes localizadas a montante do Rio Jordão e a montante do Córrego, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.	55
Figura 29 – Resultado de Ω_1 e Ω_2 depois de 2 anos de simulação.	56
Figura 30 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_1 (primeiro ano).	57
Figura 31 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_1 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.	58
Figura 32 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_1 no terceiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	58
Figura 33 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_2 no primeiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	59
Figura 34 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_2 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	59
Figura 35 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_2 no Terceiro ano, nos tempos $t_1 = 181$, $t_2 = 92$	60
Figura 36 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 no primeiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	61
Figura 37 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	62
Figura 38 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 no terceiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	62
Figura 39 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 (1460 dias - 4 anos).	63
Figura 40 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no primeiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	63
Figura 41 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	64
Figura 42 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no terceiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	64
Figura 43 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no quarto ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$	65

Lista de tabelas

Tabela 1 – Velocidade superficial (comprimento/tempo) dos ventos dominantes predominantes de Noroeste e Sudeste usando a fórmula de Ekman (valores estimados)	53
---	----

Lista de abreviaturas e siglas

IQA	Índice de Qualidade da Água
ANA	Agência Nacional de Água
IGAM	Instituto Mineiro de Gestão de Águas

Lista de códigos

parabolicsnovo.m	70
corregoteste.m	77

Sumário

Introdução	17
1 O estudo de caso	19
1.1 O estudo de caso	19
2 O modelo matemático	22
2.1 O modelo geral	22
2.1.1 Domínio Ω_1	23
2.1.2 Domínio proposto Ω_2	25
3 Método das Diferenças Finitas, Método de Crank-Nicolson e Núcleo de Peclét	30
3.1 Método das Diferenças Finitas, Método de Crank-Nicolson e Núcleo de Péclét	30
3.1.1 Métodos das Diferenças Finitas	30
3.1.2 Métodos de Crank-Nicolson	33
3.1.3 Núcleo de Péclét	34
4 Processos Numéricos	35
4.1 Processos numéricos	35
4.1.1 Condições de contorno	35
4.1.2 Equação difusão-advecção-reação	35
4.1.3 Procedimento Numérico para Ω_1	37
4.1.4 Pontos da fronteira de Ω_1	40
4.1.4.1 Pontos da fronteira acima	40
4.1.4.2 Pontos da fronteira à direita	42
4.1.4.3 Pontos da fronteira à esquerda	44
4.1.4.4 Pontos da fronteira abaixo	44
4.1.5 Procedimento Numérico para Ω_2	46
4.1.5.1 Fronteira acima ($\partial\Omega_{2_1}$) de Ω_2 :	48
4.1.5.2 Fronteira à esquerda do córrego ($\partial\Omega_{2_2}$) de Ω_2 :	48
4.1.5.3 Fronteira acima do córrego ($\partial\Omega_{2_3}$) de Ω_2 :	48
4.1.5.4 Fronteira à direita do córrego ($\partial\Omega_{2_4}$) de Ω_2 :	49
4.1.5.5 Fronteira acima ($\partial\Omega_{2_5}$) de Ω_2 :	49
4.1.5.6 Fronteira à direita do rio ($\partial\Omega_{2_6}$) de Ω_2 :	50
5 Simulações e Análise dos Resultados	52
5.1 Simulações e Análise dos Resultados	52
5.1.1 Primeiro cenário de Ω_1 e Ω_2	53
5.1.2 Cenário Ω_1 e Ω_2 - Controle Urbano Parcial	56
5.1.3 Cenário Ω_1 e Ω_2 - Controle Total Urbano	61
6 Considerações Finais	66

REFERÊNCIAS	67
Anexos	69
ANEXO A Domínio Ω_1	70
ANEXO B Domínio Ω_2	77

Introdução

A preocupação com a proteção ambiental já se estabeleceu como um papel relevante no cenário mundial, ainda que mais em acordos do que em ações, uma forte contribuição vem da crescente urbanização. Esta preocupação se concentra em diversas áreas do ambiente incluindo, em muitas cidades, um cuidado com os corpos aquáticos, sujeitos a impactos ambientais em função das atividades antrópicas.

São estudos importantes aqueles que desenvolvem ferramentas que permitem simular situações de risco com vistas a estabelecer estratégias e políticas de prevenção, de contingência e até, em casos extremos, de recuperação de ambientes, a fim de construir estratégias para a resolução de problemas ambientais que podem ser evitados ou, pelo menos, amenizados. Hoje existem diversos trabalhos na área da matemática aplicada, como (ABREU, 2009), (GUACA, 2015), (ALVES, 2009), (CANTAO, 1998.), (DINIZ, 1994), (DINIZ, 2003), (INFORZATO, 2008), (KRINDGES, 2011), (MISTRO, 1992), (OLIVEIRA, 2003), (VASQUEZ, 2005), (WOLMUTH, 2009) que enfatizam diversas técnicas matemáticas e computacionais para um vasto campo de estudo. Este trabalho visa apresentar técnicas com uma visão do problema um pouco diferente das que encontramos nos trabalhos citados, mas que se adaptam a uma vasta gama de situações.

O local de estudo deste trabalho é o rio Jordão, localizado a jusante da cidade de Araguari-MG que localiza-se no médio Paranaíba. No seu distrito sede situa-se a Bacia Hidrográfica do Córrego Brejo Alegre e o médio curso do referido córrego está entre as coordenadas geográficas $18^{\circ}37'$ e $18^{\circ}39'$ de latitude sul e $48^{\circ}8'$ e $48^{\circ}11'$ de longitude Oeste. Sua nascente principal é localizada na área urbana de Araguari-MG, atravessando longitudinalmente o município, perpassando uma área rural até desaguar no rio Jordão, cuja coordenadas são $18^{\circ}35'30''$ de latitude sul e $48^{\circ}7'43,5''$ de longitude oeste.

Além disso, é importante ressaltar que, desde a década de 60, já se encontram registros da contaminação das águas no alto curso desse córrego. No médio curso, encontram-se concentradas indústrias contaminadoras, como o Frigorífico Mataboi e o Frigorífico Santa Lúcia (desde a década de 50), como também empreendimentos de elevada geração de poluição, como curtumes, charqueadas, processamento de ossos bovinos e equinos como a fábrica de botões e outras pequenas e grandes iniciativas empresariais. Todos esses efluentes *in natura* se direcionam ao médio e baixo curso do córrego Brejo Alegre, que deságua na Bacia Hidrográfica do Rio Jordão, o qual é afluente da margem esquerda do Rio Paranaíba na divisa do estado de Minas Gerais com Goiás.

Neste trabalho, pretendemos utilizar da modelagem matemática, que, segundo (BASSANEZI, 2004), consiste na arte de transformar problemas da realidade em problemas

matemáticos e resolvê-los interpretando suas soluções na linguagem do mundo real.

O capítulo 1 deste trabalho contém uma rápida apresentação da história da ameaça da poluição do rio Jordão, enfocando a medida de poluição que iremos adotar, o Índice de Qualidade da Água (IQA), quando foi criado e como será utilizado para o nosso trabalho.

No capítulo 2 apresentaremos a equação de difusão-advecção-reação para estudar a dispersão de poluente no rio Jordão. Além disso, dividiremos o capítulo em duas subseções. Na primeira iremos descrever o rio Jordão como um domínio retangular, como é encontrado em diversos trabalhos já citados mais acima, com um perfil de velocidade dito parabólico (FORTUNA, 2000). Já na segunda subseção, apresentaremos nossa proposta de domínio, também regular, mas não mais retangular como anteriormente, e ainda, descreveremos a função que rege a velocidade nesse novo domínio.

No capítulo 3 faremos uma breve descrição dos métodos numéricos que foram utilizados para a discretização da equação de difusão-advecção, começando pelo Método das Diferenças Finitas, passando por Crank-Nicolson e, por fim, um breve comentário sobre a estabilidade da solução numérica apresentando o Núcleo de Péclet.

No capítulo 4 apresentaremos os procedimentos numéricos utilizados nos dois domínios apresentado neste trabalho. Apresentaremos as condições de contorno que consideramos durante as simulações dos algoritmos, além das equações já discretizadas, os sistemas lineares e como eles são formados, tanto considerando um domínio que dizemos retangular, tanto como o novo domínio proposto.

O capítulo 5 está destinado a exibição dos resultados das simulações computacionais com base na modelagem proposta no capítulo 2, seguido de análises dos resultados.

Por fim, no capítulo 6 encontram-se as conclusões.

1 Breve histórico do rio Jordão

1.1 O estudo de caso

O foco deste trabalho será o estudo da dispersão do poluente no rio Jordão (Figura 1), este afluente do rio Paranaíba, que hoje é represado e responsável pela energia de vários municípios do estado de Minas Gerais e Goiás. O córrego Brejo Alegre é o principal responsável pela poluição do rio Jordão, fazendo-o um dos rios mais poluídos do estado de Minas. De acordo com a reportagem do Diário de Araguari de 1997, com o título “Rio Jordão está ameaçado pela poluição”, o córrego Brejo Alegre é apontado como o maior poluidor do rio. O texto relata que, em estudo realizado em 1987, a cidade de Araguari é a maior poluidora do rio Jordão, ficando constatado que a cidade era responsável por 48% da poluição e os outros 52% eram distribuídos entre as empresas, incluindo o cortume (O Diário, 1997, p.01). A situação atual do rio só piorou desde esta época, dada a falta de investimento em tratamento de esgoto no córrego e/ou no rio.

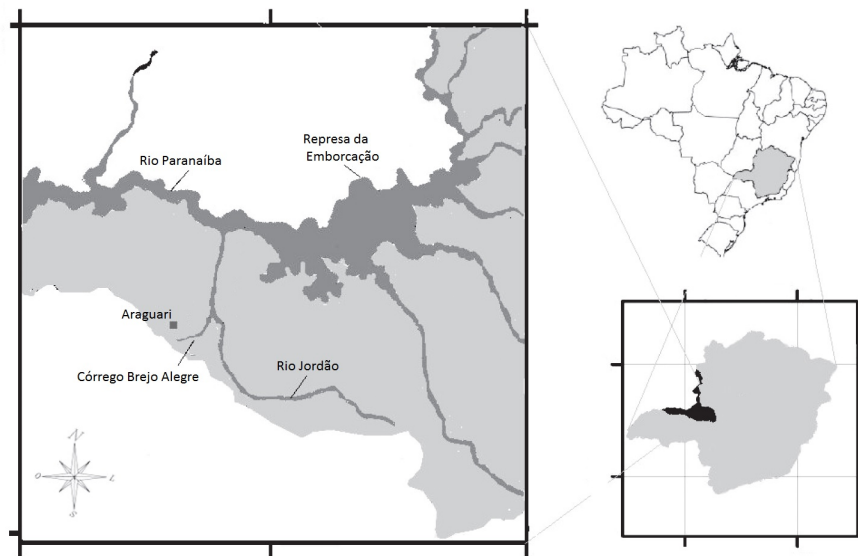


Figura 1 – Identificação do córrego Brejo Alegre (afluente do rio Jordão) e do rio Jordão (SALLA et al., 2013).

Como uma medida da poluição, é utilizado o Índice de Qualidade das Águas. De acordo com a agência nacional de água (ANA), esse índice foi criado em 1970, nos Estados Unidos. Em 1975 o índice começou a ser utilizado pela Companhia Ambiental do Estado de São Paulo, sendo implementado nos outros estados nas décadas seguintes e hoje é o principal índice de qualidade da água utilizado em todo país.

A escolha pela utilização do IQA neste trabalho se deve aos parâmetros utiliza-

dos em seu cálculo, que são, em sua maioria, indicadores de contaminação causada pelo lançamento de esgoto doméstico.

Para o nosso local de estudo, observamos que a poluição do rio Jordão pode ser comprovada pelo Instituto Mineiro de Gestão de Águas - IGAM (2013) quanto à classificação das águas da bacia do rio Paranaíba: o Índice de Qualidade de Água (IQA) mostra que o rio Jordão não apresenta bons índices, sendo qualificado como Razoável (IQA entre 51-70) e, de acordo com o relatório trimestral de águas superficiais do 1º trimestre de 2015 (COELHO, 2015), o rio possui uma elevada violação de limites de fósforo total passando de 500% do parâmetro permitido (Figura 2). Isto indica que o rio Jordão recebe, como fonte poluidora, o esgoto doméstico. Em relação ao IQA, foi obtido um índice de 58, sendo que no mesmo período de 2014 o índice era de 57,1 (Figura 3).

Bacia/Sub-bacia Hidrográfica	UPGRH	Corpo d'água	Estação	Classe de Enquadramento	Municípios	Parâmetros que não atenderam ao limite (DN COPAM / CERH 01/08)	Percentual de Violação do parâmetro
Rio Paraíba do Sul	PS2	Rio Xopotó	DS077	Classe 2	Visconde do Rio Branco	Demanda Bioquímica de Oxigênio	660%
Rio Paraíba do Sul	PS2	Rio Xopotó	BS077	Classe 2	Visconde do Rio Branco	<i>Escherichia coli</i>	24096%
Rio Paraíba do Sul	PS2	Rio Xopotó	BS077	Classe 2	Visconde do Rio Branco	Fósforo total	60%
Rio Paraíba do Sul	PS2	Rio Xopotó	DS077	Classe 2	Visconde do Rio Branco	Manganês total	457%
Rio Paraíba do Sul	PS2	Rio Xopotó	BS077	Classe 2	Visconde do Rio Branco	Oxigênio dissolvido	900%
Rio Paraíba do Sul	PS2	Rio Xopotó	BS077	Classe 2	Visconde do Rio Branco	Sólidos em suspensão totais	47%
Rio Paraíba do Sul	PS2	Rio Xopotó	BS077	Classe 2	Visconde do Rio Branco	Substâncias tensoativas	514%
Rio Paranaíba	PN1	Ribeirão da Batalha	PB036	Classe 2	Paracatu	pH in loco	11%
Rio Paranaíba	PN1	Rio Bagagem	PB040	Classe 2	Estrela do Sul	<i>Escherichia coli</i>	768%
Rio Paranaíba	PN1	Rio Dourados	PB036	Classe 2	Abadia dos Dourados	<i>Escherichia coli</i>	1020%
Rio Paranaíba	PN1	Rio Jordão	PB009	Classe 2	Araguari	<i>Escherichia coli</i>	245%
Rio Paranaíba	PN1	Rio Jordão	PB009	Classe 2	Araguari	Fósforo total	560%
Rio Paranaíba	PN1	Rio Paranaíba	PB001	Classe 2	Rio Paranaíba	<i>Escherichia coli</i>	208%
Rio Paranaíba	PN1	Rio Paranaíba	PB001	Classe 2	Rio Paranaíba	Manganês total	15%

Figura 2 – Parâmetro de fósforo elevado do rio Jordão (COELHO, 2015).

Atualmente, o rio Jordão (Figura 1) está classificado como razoável, com o IQA em torno de 58. Nosso objetivo então, é construir uma ferramenta matemática que analise qual a localização mais eficiente para fazer o tratamento de esgoto. Se seria melhor junto à fonte poluidora ainda no córrego, ou já no rio Jordão, onde o córrego deságua no rio. Com os resultados obtidos, podemos futuramente trabalhar com políticas públicas que possam auxiliar na melhoria do índice de IQA da água na região.

Corpo de água	Estação	Municípios	INDICADORES						PARÂMETROS QUE NÃO ATENDERAM O LIMITE LEGAL		
			Resultados dos indicadores 1º Trimestre						Mapa do Panorama de Qualidade das Águas no 1º Trimestre de 2015		
			IQA		CT		IET		Parâmetros indicativos de:		
			2014	2015	2014	2015	2014	2015	Contaminação Fecal	Enriquecimento orgânico	Substâncias tóxicas
Rio Jordão	PB009	Araguari	57,1	58	BAIXA	BAIXA	53,8	53,2	<i>Escherichia coli.</i>	Fósforo total.	---

Figura 3 – Índice de Qualidade da Água do rio Jordão do primeiro trimestre de 2015 e relação ao mesmo período de 2014 (COELHO, 2015).

2 O modelo matemático

Diversos trabalhos citados anteriormente como (GUACA, 2015) nos lembra da complexidade que é modelar fenômenos reais, cujos comportamentos, se analisados todos, são difíceis de observar. Restringindo estes fenômenos podemos retratar o problema real de forma matemática, de modo que mantenha sua relevância.

2.1 O modelo geral

Mesmo que não possamos retratar o problema real de maneira exata, podemos escolher variáveis do fenômeno estudado, de tal forma que possamos desenvolver e interpretar os resultados e assim, aplica-los na realidade. Uma variedade de fenômenos relativos a problemas ecológicos e ambientais podem ser modelados baseado na equação diferencial parcial de difusão-advecção-reação. Esta formulação é chamada formulação clássica ou formulação forte do problema. Essa formulação exige mais condições sobre o problema, no entanto conseguimos atender a essas condições pois iremos trabalhar em um domínio bem comportado e contínuo, de fronteira regular, possibilitando a garantia da solução aproximada da equação em sua formulação forte.

Inicialmente, vamos apresentar a equação diferencial parcial de difusão-advecção-reação, que, neste trabalho, irá modelar a dispersão de poluentes em um trecho do rio Jordão nos arredores da cidade de Araguari - MG. Consideraremos também os fenômenos de decaimento e as fontes poluidoras que desaguam no córrego Brejo Alegre, afluente do rio Jordão.

De acordo com (OKUBO, 2001) a equação é descrita por:

$$\left\{ \begin{array}{l} c = c(x, y, t), \quad \text{com} \quad (x, y) \in \Omega \subset R^2 \quad \text{e} \quad t \in J = (0, T], \\ \frac{\partial c}{\partial t} - \text{div}(\alpha \nabla c - c \vee) = f - \mu c, \\ c(x, y, 0) = c_0(x, y), \quad \forall (x, y) \in \Omega, \quad \text{com adequadas condições de contorno} \end{array} \right. , \quad (2.1)$$

onde

- $c(x, y, t)$ é a função de concentração de poluente no ponto (x, y) e instante de tempo t ;

- $c(x, y, 0)$ é a condição inicial;
- α representa o coeficiente de difusão;
- v representa o termo de advecção, ou seja, velocidade de transporte;
- μ representa o termo de decaimento do poluente e
- f representa a fonte.

O fluxo de difusão e advecção já foi bem retratado nos trabalhos de (GUACA, 2015) e (VASQUEZ, 2005). Segundo os autores, a difusão que eles trabalharam é a chamada difusão efetiva (OKUBO, 2001), ou difusão macroscópica devida à tensão superficial. Pela lei de Fick assumimos que a matéria tem uma tendência a se espalhar de forma regular, ou seja, que a matéria se movimenta de locais de maior concentração para os de menor concentração.

Ainda sobre o fluxo advectivo, no trabalho de (ABREU, 2009), a autora ressalta que o transporte advectivo é muito mais complexo do que na superfície, o que exigiria uma análise mais elaborada ou até outras ferramentas matemáticas para tratar o problema. Em razão disto, vamos nos restringir aos poluentes da superfície, os despejos orgânicos. Esse tipo de poluente está sujeito ao transporte advectivo da superfície, que é quase totalmente influenciado pelo vento, segundo estudo realizado em (POFFO, 2007).

2.1.1 Domínio Ω_1

Iremos, inicialmente, aproximar o domínio destacado na Figura 4 por um retângulo conforme (Figura 5). Dessa forma teremos $\Omega_1 = \{(x, y) | 0 \leq x \leq L \text{ e } 0 \leq y \leq H\}$.

onde H é a largura do rio Jordão e L é o comprimento do rio da parte inicial em que começa a poluição até uma distância considerada conveniente para estudar a dispersão da poluição.

Nosso primeiro estudo será em relação ao perfil de velocidade adotado no domínio Ω_1 . De acordo com (FORTUNA, 2000), muitas vezes não podemos precisar a distribuição de velocidades, então a solução é prescrever um perfil de velocidade quando possível, avaliado *in loco*. Neste trabalho é adotado o perfil parabólico, pois o mesmo descreve um perfil encontrado em escoamento entre duas fronteiras paralelas. Para descrever a função velocidade do domínio Ω_1 , adaptamos as ideias de (FORTUNA, 2000), conforme os passos a seguir:

Suponhamos que as fronteiras paralelas estejam separadas por uma distância H , como segue na Figura 6 e Figura 7.

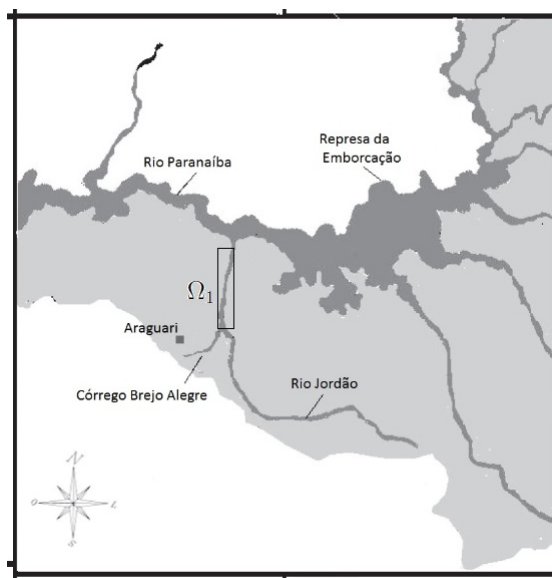


Figura 4 – Domínio retangular Ω_1 .

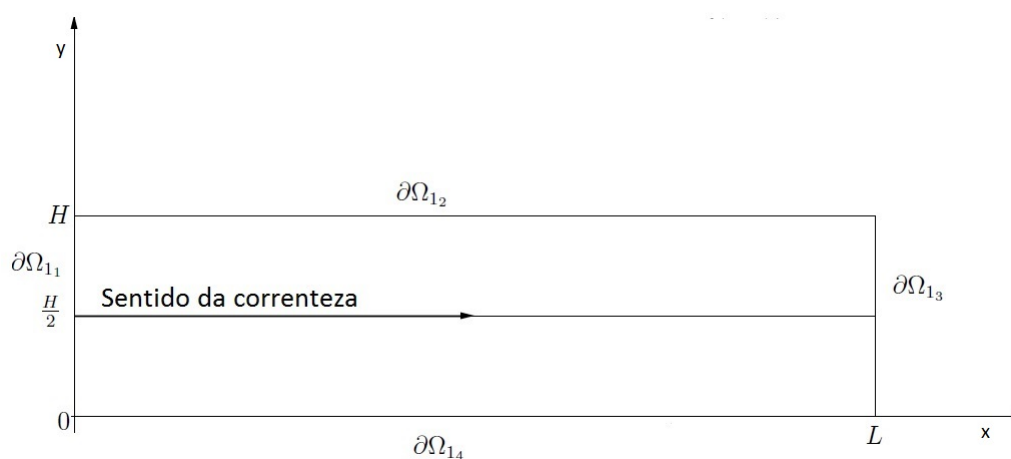


Figura 5 – Domínio Ω_1 .

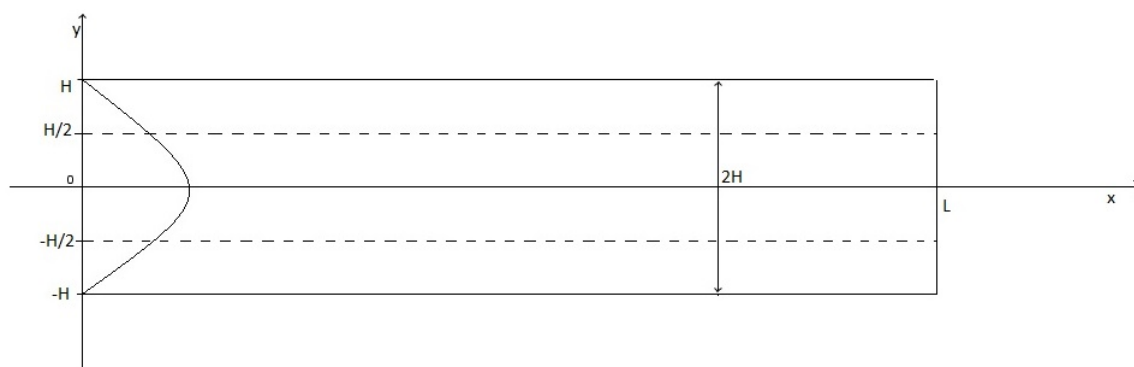
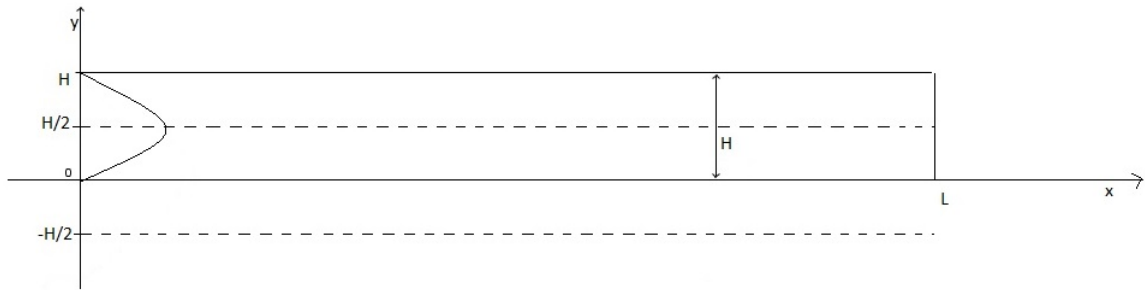


Figura 6 – Perfil parabólico retirado de (FORTUNA, 2000).

Figura 7 – Perfil parabólico do domínio Ω_1 .

A equação que descreve o perfil da Figura 6 apresentado em (FORTUNA, 2000) segue como na equação (2.2)

$$u(y) = u_0 \left(1 - \frac{y^2}{H^2} \right) \quad (2.2)$$

onde u_0 é a velocidade da calha do rio a distâncias equidistantes das margens.

Utilizando das ideias da Figura 6 para chegar à equação (2.2), tivemos um deslocamento para cima da parábola, para que sua origem seja na origem do plano cartesiano e sua segunda raiz seja na margem do rio que denotamos como iniciado no ponto $(0, H)$. Além disso, mudamos a abertura da parábola de $2H$ (Figura 6) para H (Figura 7). Então, a função que descreve o perfil de velocidade de Ω_1 é dada pela equação (2.3)

$$u(y) = u_0 \left(1 - \frac{\left(y - \frac{H}{2}\right)^2}{\left(\frac{H}{2}\right)^2} \right) \quad \text{ou}$$

$$u(y) = u_0 \left(\frac{\frac{H^2}{4} - \left(y^2 - Hy + \frac{H^2}{4}\right)}{\frac{H^2}{4}} \right) \quad \text{e, finalmente}$$

$$u(y) = 4u_0 \left(\frac{y}{H} - \frac{y^2}{H^2} \right). \quad (2.3)$$

Aqui, u_0 é velocidade conhecida da calha do rio é a mesma da equação (2.2).

2.1.2 Domínio proposto Ω_2

Agora, vamos considerar um novo domínio Ω_2 , ainda regular, mas diferente de Ω_1 . Este novo domínio Ω_2 aproxima o domínio destacado na figura 8. Dessa forma, $\Omega_2 = \{(x, y) | 0 \leq x \leq L_1 \text{ e } 0 \leq y \leq H_1\} \cup \{(x, y) | x_1 \leq x \leq x_2 \text{ e } H_1 \leq y \leq H_2\}$.

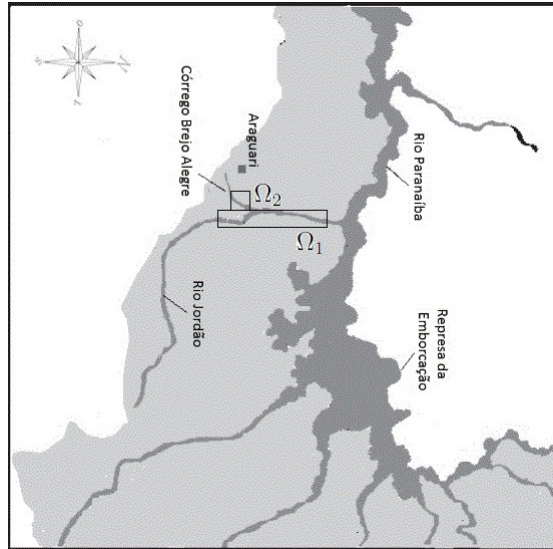


Figura 8 – Domínio Ω_2 .

Agora, introduzemos uma nova ideia, que será complementada no capítulo 3.

Primeiramente, vamos definir uma malha no domínio $\Omega = \Omega_1 \cup \Omega_2$, que é um conjunto finito de pontos que discretizam o domínio Ω . Esses pontos são chamados de nós da malha. Seja x_0 um ponto de referência e h um número positivo. A malha de passo h da Figura 9 e associada a x_0 e consiste no conjunto de pontos $x_i = x_0 + ih, i = 1, 2, \dots, n$.

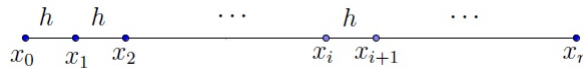


Figura 9 – Malha de passo h .

Fazendo $h = \Delta x$ e considerando 2 dimensões com $k = \Delta y$, observemos as (Figura 10) e (Figura 11), que representam o domínio Ω_1 e Ω_2 , respectivamente.

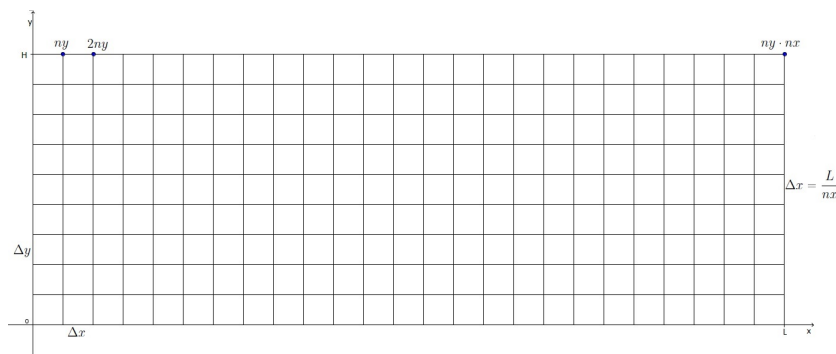


Figura 10 – Malha do domínio Ω_1 .

Na (Figura 10) temos os seguintes valores:

- Quantidade de nós em cada coluna: ny ;
- Quantidade de nós em cada linha: nx ;
- Quantidade total de nós: $ny \cdot nx$.

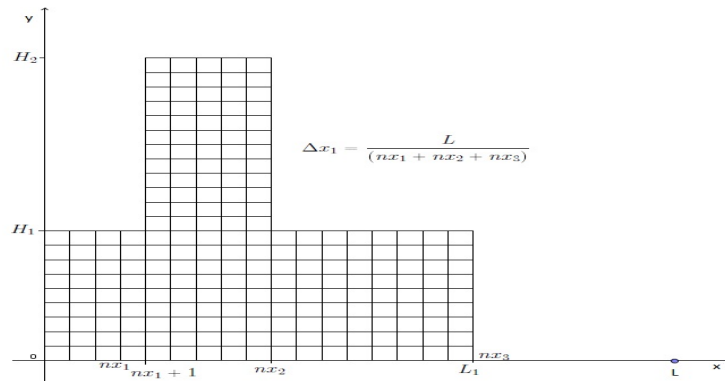


Figura 11 – Malha do domínio Ω_2 .

Observemos que, a quantidade de nós da malha (Figura 11) (domínio Ω_2) é a mesma quantidade de nós da malha representada pela (Figura 10). Isso acontece porque, consideramos o córrego como parte do domínio do problema e utilizamos o algoritmo ((A) para que pudéssemos descrever o domínio discretizado (B).

Considerando o problema de descrever a função velocidade do domínio Ω_2 , segue que: no sentido do rio, temos a mesma função que do domínio Ω_1 , representada na equação (2.3). Já a velocidade no sentido do córrego descendo até o rio é obtida através dos passos que veremos a seguir. Para obter a função procurada, iremos utilizar novamente as ideias de (FORTUNA, 2000).

Observemos as parábolas da Figura 12

De acordo com Poiseuille (BATSCHELET, 1978), o perfil de velocidade de uma uma função quadrática tem a forma mostrada na equação (2.4)

$$v(x) = v_0 \left(\frac{x^2}{H^2} - 1 \right), \quad (2.4)$$

diminuindo a distância entre as raízes da parábola, temos a equação (2.5) (parábola em vermelho) na Figura 12

$$v(x) = v_0 \left(\frac{x^2}{a_1^2} - 1 \right), \quad (2.5)$$

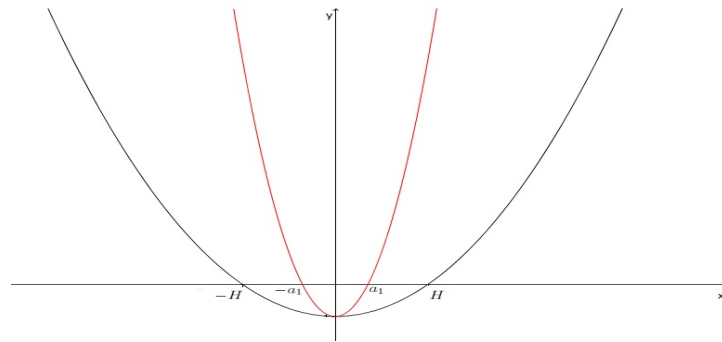


Figura 12 – Parábolas

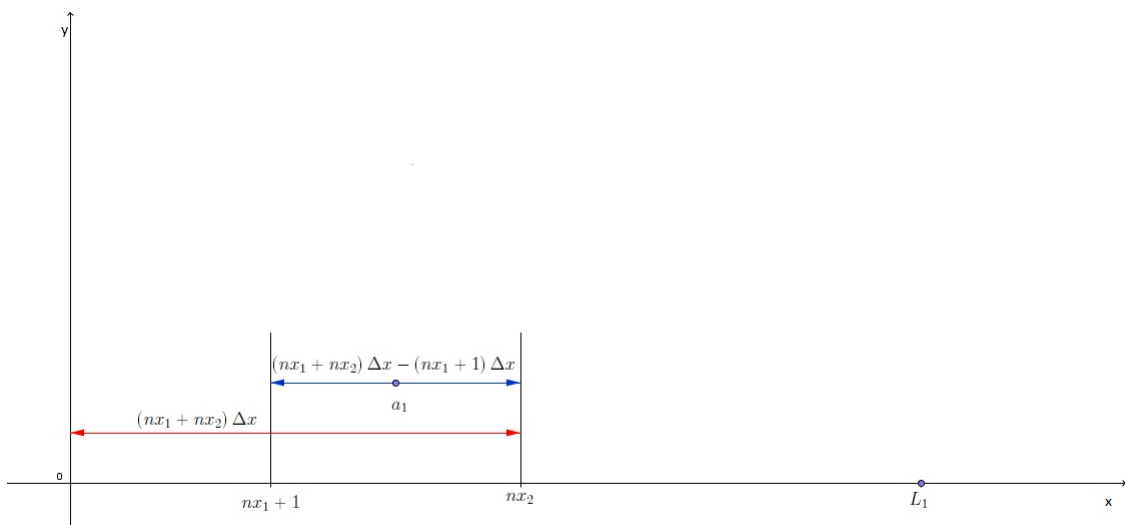


Figura 13 – Amplitude da Parábola

onde $a_1 = \frac{(nx_1 + nx_2) \Delta x - (nx_1 + 1) \Delta x}{2}$, como pode ser visto na [Figura 13](#).

Agora que temos a amplitude que queríamos, temos que observar o deslocamento da parábola para a direita; para isto consideremos a ([Figura 14](#)).

Assim, a equação que descreve a velocidade no córrego é dada pela equação [2.6](#)

$$v(x) = v_0 \left(\frac{(x - c_1)^2}{a_1^2} - 1 \right) \quad (2.6)$$

onde $c_1 = L_1 - (nx_3 \Delta x) - a_1$.

Agora que já temos a representação das funções de velocidade, introduziremos no próximo capítulo a ideia de Diferenças Finitas e Crank-Nicolson para a discretização do nosso problema.

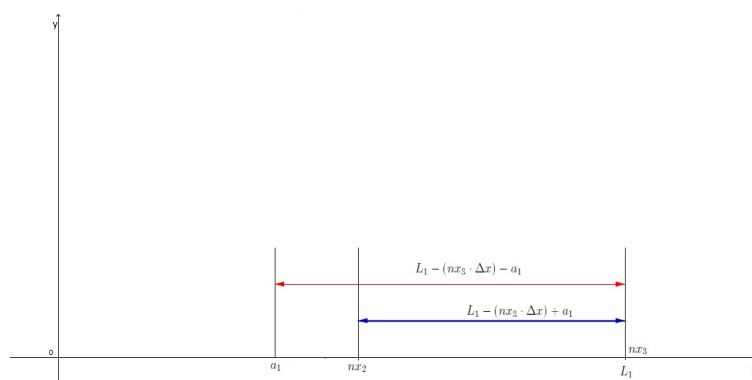


Figura 14 – Amplitude da Parábola

3 Métodos Numéricos

3.1 Método das Diferenças Finitas, Método de Crank-Nicolson e Núcleo de Péclet

Como estamos trabalhando com a equação diferencial parcial (2.1), sabemos que a possibilidade de uma expressão analítica para a solução ainda não existe. Por conta disso, utilizaremos os resultados aproximados da equação (2.1) utilizando o Método das Diferenças Finitas para discretizar o espaço e Crank-Nicolson para discretizar o tempo (CUNHA, 2000), ambos métodos de segunda ordem e respeitando o núcleo de Péclet. Daí a necessidade de uma breve introdução sobre o Método das Diferenças Finitas, Método de Crank-Nicolson e Núcleo de Péclet.

3.1.1 Métodos das Diferenças Finitas

O método de diferenças finitas centradas tem características de estabilidade numérica, mas requer mais recursos computacionais para obter uma precisão específica (BURDEN, 2013). Este método consiste na discretização das derivadas que aparecem na equação diferencial. Neste passo, as derivadas são aproximadas por diferenças entre valores da solução discretizada. A ferramenta básica na definição de aproximações para as derivadas é a série de Taylor. O mérito da série de Taylor é o de nos dizer como várias informações sobre a função, no ponto x , podem ser usadas na avaliação desta função numa vizinhança de x , isto é, no ponto $x + h$. No entanto, este método exige uma regularidade da função $c(x)$, esta que consideramos ser uma formulação forte da equação diferencial (2.1) que estamos trabalhando. Se assumirmos que $c(x)$ têm derivada até a ordem $n + 1$ em x , sua expansão em série de Taylor é:

$$c(x + h) = c(x) + hc'(x) + \frac{h^2}{2!}c''(x) + \dots + \frac{h^n}{n!}c^{(n)}(x) + \frac{h^{n+1}}{(n+1)!}c^{(n+1)}(\xi) \quad (3.1)$$

com $x < \xi < x + h$. O último termo da expressão (3.1) representa o erro da aproximação de $c(x + h)$ pelos $n + 1$ primeiros termos da série de Taylor.

Se tomarmos $n = 1$ em (3.1), teremos a fórmula avançada para a discretização da derivada e seu erro:

$$c'(x) = \frac{c(x + h) - c(x)}{h} - \frac{h}{2}c''(\xi).$$

De modo similar, tomando $-h$ em (3.1) ainda com $n = 1$, obtemos a fórmula atrasada e seu erro:

$$c'(x) = \frac{c(x) - c(x-h)}{h} + \frac{h}{2} c''(\xi).$$

Tomando agora $n = 2$ em (3.1), segue que:

$$c(x+h) = c(x) + hc'(x) + \frac{h^2}{2!} c''(x) + \frac{h^3}{3!} c^{(3)}(\xi_1) \quad (3.2)$$

e

$$c(x-h) = c(x) - hc'(x) + \frac{h^2}{2!} c''(x) - \frac{h^3}{3!} c^{(3)}(\xi_1) \quad (3.3)$$

fazendo (3.2)-(3.3), obtemos a fórmula centrada para a discretização da derivada e o seu erro:

$$c'(x) = \frac{c(x+h) - c(x-h)}{2h} - \frac{h^2}{3!} c^{(3)}(\xi).$$

De maneira análoga, tomando $n = 4$, temos a aproximação para a derivada de segunda ordem, ou seja

$$c''(x) = \frac{c(x-h) - 2c(x) + c(x+h)}{h^2} - \frac{h^2}{12} c^{(iv)}(\xi)$$

com $x-h < \xi < x+h$.

No método das Diferenças Finitas, as derivadas presentes na equação diferencial são substituídas por aproximações como as descritas acima usando h constante. Ou seja, para cada ponto da malha no interior do intervalo onde a equação diferencial está definida, denotamos por c_i a aproximação de $c(x_i)$, então, as aproximações centradas vistas acima serão representadas como segue nas equações (3.4) e (3.5).

$$c'_i = \frac{c_{i+1} - c_{i-1}}{2h} \quad (3.4)$$

$$c''_i = \frac{c_{i+1} - 2c_i + c_{i-1}}{h^2}. \quad (3.5)$$

A equação bidimensional que rege este trabalho é a equação diferencial parcial de difusão-advectação-reação (2.1), e pelas condições geofísicas do problema, tratamo-os de maneira

bidimensional. Ao dividir o domínio finito $[0, L]$ das abscissas em nx subintervalos, obtemos $\Delta x = h = \frac{L}{nx}$, e este valor é o do espaçamento da malha no sentido do eixo x e $c = c(x, y)$ uma função qualquer de duas variáveis de classe C^∞ . Usando a série de Taylor na incógnita x , segue que:

$$c(x + h, y) = c(x, y) + \frac{\partial c}{\partial x}(x, y)h + \frac{\partial^2 c}{\partial x^2}(x, y)\frac{h^2}{2!} + \frac{\partial^3 c}{\partial x^3}(x, y)\frac{h^3}{3!} \quad (3.6)$$

e

$$c(x - h, y) = c(x, y) - \frac{\partial c}{\partial x}(x, y)h + \frac{\partial^2 c}{\partial x^2}(x, y)\frac{h^2}{2!} - \frac{\partial^3 c}{\partial x^3}(x, y)\frac{h^3}{3!}. \quad (3.7)$$

Fazendo algumas operações algébricas, de maneira adequada, nas equações (3.6) e (3.7) segue que:

$$\frac{\partial c}{\partial x} \approx \frac{c(x + h, y) - c(x, y)}{h}, O(h); \quad (3.8)$$

$$\frac{\partial c}{\partial x} \approx \frac{c(x, y) - c(x - h, y)}{h}, O(h); \quad (3.9)$$

$$\frac{\partial^2 c}{\partial x^2} \approx \frac{c(x + h, y) - 2c(x, y) + c(x - h, y)}{h^2}, O(h^2), \quad (3.10)$$

onde $O(h)$ e $O(h^2)$ são, respectivamente, erros da ordem de h e h^2 . Somando (3.8) e (3.9) temos a aproximação centrada, a qual utilizaremos:

$$\frac{\partial c}{\partial x}(x, y) \approx \frac{c(x + h, y) - c(x - h, y)}{2h}, O(h^2). \quad (3.11)$$

De maneira semelhante, dividindo $[0, H]$ em ny subintervalos obtemos as aproximações em relação a variável y , agora com $\Delta y = k$ o espaçamento da malha no sentido do eixo y . Utilizamos a seguinte notação para os pontos da malha:

$$\begin{cases} c(x_i, y_j) \approx c_{i,j} \\ c(x_i + h, y_j) \approx c_{i+1,j} \\ c(x_i - h, y_j) \approx c_{i-1,j} \\ c(x_i, y_j + k) \approx c_{i,j+1} \\ c(x_i, y_j - k) \approx c_{i,j-1} \end{cases} \quad (3.12)$$

e ainda:

$$\left\{ \begin{array}{l} \frac{\partial c}{\partial x}(x_i, y_j) \approx \frac{c_{i+1,j} - c_{i-1,j}}{2h} \\ \frac{\partial c}{\partial y}(x_i, y_j) \approx \frac{c_{i,j+1} - c_{i,j-1}}{2k} \\ \frac{\partial^2 c}{\partial x^2}(x_i, y_j) \approx \frac{c_{i+1,j} - 2c_{i,j} + c_{i-1,j}}{h^2} \\ \frac{\partial^2 c}{\partial y^2}(x_i, y_j) \approx \frac{c_{i,j+1} - 2c_{i,j} + c_{i,j-1}}{k^2} \end{array} \right. \quad (3.13)$$

No entanto, embora esta notação confira a $c_{i,j}$ uma característica matrcial, algoritmicamente iremos trabalhar este conjunto de valores como um vetor, reenumerando de modo adequado os pontos x_i, y_j .

Assim, no lugar de, por exemplo, $c_{i,j-1}^k$ iremos usar c_{l-ny}^k identificando o nó imediatamente à esquerda de c_l^k - que corresponde à aproximação $c(x_l, y_l)$. Uma notação análoga é usada para c_{l+ny}^k , c_{l-1}^k e c_{l+1}^k (PRESTES, 2011).

Para mais detalhes sobre diferenças finitas, ver (CUNHA, 2000) e (BURDEN, 2013).

3.1.2 Métodos de Crank-Nicolson

Em decorrência, e já utilizando esta notação que trata a aproximação de c como vetor, decorre que para a discretização do tempo da equação (2.1), iremos proceder de maneira análoga ao que foi feito para discretizar o espaço usando diferenças finitas. Desse modo, obtemos uma aproximação para a primeira derivada da função $c(x, y, t)$ em função do tempo. Primeiro definimos uma malha temporal, e seja $\Delta t = T/nt$ o espaçamento dessa malha, com $(0, T] \subset R$ e nt subintervalos. Daí, utilizando novamente o polinômio de Taylor agora na variável temporal, segue que:

$$c_i^k = c_i^{k+\frac{1}{2}} - \frac{\Delta t}{2} \frac{\partial c}{\partial t} + \frac{\left(\frac{-\Delta t}{2}\right)^2}{2} \frac{\partial^2 c}{\partial t^2} + O(\Delta t^3) \quad (3.14)$$

$$c_i^{k+1} = c_i^{k+\frac{1}{2}} + \frac{\Delta t}{2} \frac{\partial c}{\partial t} + \frac{\left(\frac{\Delta t}{2}\right)^2}{2} \frac{\partial^2 c}{\partial t^2} + O(\Delta t^3). \quad (3.15)$$

Utilizamos (3.14) e (3.15) para estimar a derivada parcial $\frac{\partial c}{\partial t}$ nos pontos $t_n + \frac{\Delta t}{2}$ pois estamos usando o método incondicionalmente estável de Crank-Nicolson. Em termos de notação $c(x_i, y_i, t_k) \approx c_i^k$ e $c(x_i, y_i, t_{k+\frac{\Delta t}{2}}) \approx c_i^{k+\frac{1}{2}}$.

Assim, subtraindo (3.14) de (3.15), e fazendo operações algébricas, segue que a derivada $\frac{\partial c}{\partial t}$ em $t = k$ será como na equação (3.16) que se segue:

$$\frac{\partial c}{\partial t} \approx \frac{c_i^{k+1} - c_i^k}{\Delta t}, \quad \text{com erro da ordem de } \Delta t^2. \quad (3.16)$$

Agora, somando as equações (3.14) e (3.15) e desenvolvendo-as, segue que, no tempo $t = k + \frac{1}{2}$ temos a equação (3.17)

$$c_i^{k+\frac{1}{2}} \approx \frac{c_i^k + c_i^{k+1}}{2}, \quad \text{também com erro da ordem de } \Delta t^2. \quad (3.17)$$

Desse modo, por (3.16) e (3.17) temos os instrumentos numéricos para a discretização temporal pelo método de Crank-Nicolson.

3.1.3 Núcleo de Péclét

Como se sabe, a EDP Difusão-Advecção-reação apresenta, entre outros, os operadores de advecção (tipo hiperbólico) e de difusão (tipo parabólico). De acordo com (VIEIRA, 1990) e (BROOKS; HUGHES, 1982), as técnicas clássicas de diferenças centradas permitem simular adequadamente os termos da equação, se o operador de difusão não for dominante, ou seja, se o número de Péclét (como definido na equação 3.18) não for respeitado como um limitante, oscilações falsas ocorrem. Quando o número de Péclét é maior ou igual a 2, a solução não é confiável. Daí, para nosso estudo:

$$P_x = \frac{u\Delta x}{\alpha} < 2 \quad \text{e} \quad P_y = \frac{v\Delta y}{\alpha} < 2 \quad (3.18)$$

onde u e v são as componentes de velocidade do termo advectivo ($V = \langle u, v \rangle$) nos eixos x e y , Δx e Δy é o espaçamento da malha no sentido x e y respectivamente e α é o coeficiente de difusibilidade (área/tempo).

4 Capitulo 4

4.1 Processos numéricos

Neste capítulo apresentaremos as aproximações realizadas na equação (2.1) para aproximar a solução do modelo descrito no capítulo 2, utilizando os métodos propostos. Apresentaremos uma breve introdução sobre condições de contorno em geral, para depois especificá-las em cada domínio considerado. Além disso, mostraremos a equação de difusão-advecção-reação que será aproximada, e o procedimento numérico para os domínios Ω_1 e Ω_2 .

4.1.1 Condições de contorno

O modelo de difusão-advecção-reação com que estamos trabalhando (2.1), descreve a concentração de poluente $c = c(x, y, t)$ em um certo instante de tempo, que difunde-se na água no ponto (x, y) e é transportado por uma velocidade $\vec{v} = (u, v)$. As condições de contorno relacionam o fluxo de poluente à densidade da fronteira. Especificamente, seja $\Omega \in R^2$ uma região limitada com fronteira suave $\partial\Omega$, e denote-se $\vec{\eta}$ como o vetor normal. Daí, o fluxo que atravessa a fronteira $\partial\Omega$ é proporcional à densidade com constante de probabilidade $\beta(x, y)$. A condição de contorno para o modelo (CANTRELL; COSNER, 2004) que é feito neste trabalho é como segue em (4.1);

$$\alpha \frac{\partial c}{\partial \vec{\eta}} = [\beta - \vec{v} \cdot \vec{\eta}] c = 0$$

$$\text{para } \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 = \partial\Omega \quad \text{com } \Gamma_i \cap \Gamma_j = \emptyset \quad c|_{\Gamma_1} = 0 \quad \frac{\partial c}{\partial \vec{\eta}}|_{\Gamma_2} = 0 \quad \text{e } c|_{\Gamma_3} = g, \quad \forall t \in J. \quad (4.1)$$

Aqui, $g = g(x, y, t)$ é uma função dada, em virtude das condições físicas do problema. O que se pretende modelar é a presença de material impactante proveniente de atividades agrícolas às margens do rio Jordão. Consideremos, no algoritmo, $g(x, y, t) = cte$.

4.1.2 Equação difusão-advecção-reação

Consideremos,

$$\frac{\partial c}{\partial t} - \text{div}(\alpha \nabla c - c \mathbf{v}) + \mu c = f \quad (4.2)$$

onde $c = c(x, y, t)$, $(x, y) \in \Omega$, $t \in I$, α é constante, $\text{div}(\mathbf{v}) = 0$ em todo o Ω e $\mathbf{v} = \langle u, v \rangle$.

Desenvolvendo a equação apresentada (4.2) e fazendo operações algébricas adequadas nas derivadas, segue que a equação que será discretizada. Então, temos que:

$$\frac{\partial c}{\partial t} - \left[\frac{\partial}{\partial x} \left(\alpha \frac{\partial c}{\partial x} \right) + \frac{\partial}{\partial y} \left(\alpha \frac{\partial c}{\partial y} \right) \right] + \frac{\partial}{\partial x}(uc) + \frac{\partial}{\partial y}(vc) + \mu c = f,$$

$$\frac{\partial c}{\partial t} - \left[\alpha \frac{\partial^2 c}{\partial x^2} + \frac{\partial c}{\partial x} \left(\frac{\partial \alpha}{\partial x} \right) + \alpha \frac{\partial^2 c}{\partial y^2} + \frac{\partial c}{\partial y} \left(\frac{\partial \alpha}{\partial y} \right) \right] + \frac{\partial u}{\partial x}c + u \frac{\partial c}{\partial x} + \frac{\partial v}{\partial y}c + v \frac{\partial c}{\partial y} + \mu c = f \quad e$$

$$\frac{\partial c}{\partial t} - \alpha \left(\frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right) + \left(u \frac{\partial c}{\partial x} + v \frac{\partial c}{\partial y} \right) + c \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \mu \right) = f. \quad (4.3)$$

A discretização da equação (4.3) é feita pelo método das diferenças finitas, com a opção para as diferenças centradas de 2ª ordem para a variável espacial e Crank-Nicolson, para o tempo, também da ordem $O(\Delta t^2)$. Assim, temos que:

$$c_i^{k+\frac{1}{2}} \cong \frac{c_i^{k+1} + c_i^k}{2}; \quad (4.4)$$

$$\frac{\partial c(x_i, y_i, t_{k+\frac{\Delta t}{2}})}{\partial t} \cong \frac{c_i^{k+1} - c_i^k}{\Delta t}; \quad (4.5)$$

$$\frac{\partial c(x_i, y_i, t_{k+\frac{\Delta t}{2}})}{\partial x} \cong \frac{c_{i+ny}^{k+1} - c_{i-ny}^{k+1} + c_{i+ny}^k - c_{i-ny}^k}{4\Delta x}. \quad (4.6)$$

Analogamente,

$$\frac{\partial c(x_i, y_i, t_{k+\frac{\Delta t}{2}})}{\partial y} \cong \frac{c_{i+1}^{k+1} - c_{i-1}^{k+1} + c_{i+1}^k - c_{i-1}^k}{4\Delta y}; \quad (4.7)$$

$$\frac{\partial^2 c(x_i, y_i, t_{k+\frac{\Delta t}{2}})}{\partial x^2} \cong \frac{c_{i-ny}^{k+1} - 2c_i^{k+1} + c_{i+ny}^{k+1} + c_{i-ny}^k - 2c_i^k + c_{i+ny}^k}{2(\Delta x)^2}; \quad (4.8)$$

$$\frac{\partial^2 c(x_i, y_i, t_{k+\frac{\Delta t}{2}})}{\partial y^2} \cong \frac{c_{i-1}^{k+1} - 2c_i^{k+1} + c_{i+1}^{k+1} + c_{i-1}^k - 2c_i^k + c_{i+1}^k}{2(\Delta y)^2}. \quad (4.9)$$

4.1.3 Procedimento Numérico para Ω_1

Considerando um domínio simplificado $\Omega_1 \subset R^2$, descrito em forma de retângulo, como visto na figura (15), onde $\partial\Omega_1$, $\partial\Omega_2$, $\partial\Omega_3$ e $\partial\Omega_4$ são as fronteiras do Domínio Ω_1 . Dividindo o intervalo $[0, l]$ em nx subintervalos, obtemos $\Delta x = \frac{l}{nx}$ e de maneira análoga, $\Delta y = \frac{h}{ny}$, sendo ny o número de subintervalos de $[0, h]$.

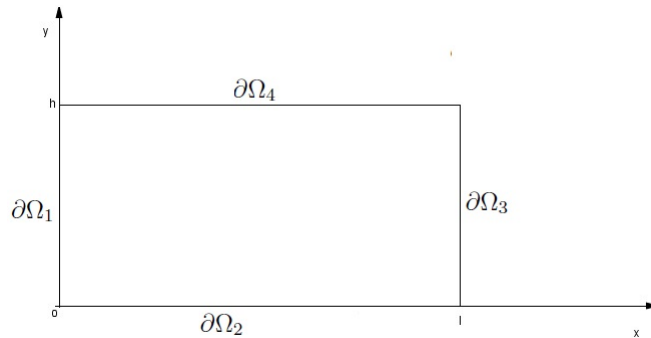


Figura 15 – Domínio Ω_1 .

Para enumerar os nós obtidos na malha decorrente da discretização, iremos adotar uma seqüência nos intervalos verticais sucessivos, como podemos ver na figura (16). Observe-se que a numeração da malha não começa na primeira coluna e nem abrange a primeira linha. Isso ocorre por conta da escolha das condições de contorno do domínio Ω_1 , em $\partial\Omega_1$ e $\partial\Omega_2$, que são dirichlet homogênea e não homogênea, respectivamente .

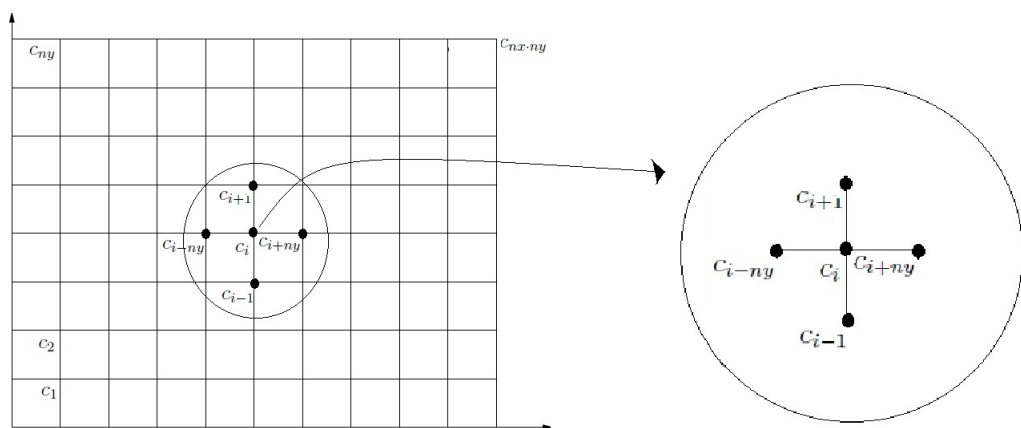


Figura 16 – Contagem da malha.

Desse modo, temos pontos interiores do domínio, como indicado na figura (17). Usando as aproximações dadas pelas equações (4.4), (4.5), (4.6), (4.7), (4.8) e (4.9) na equação (4.3), segue-se que:

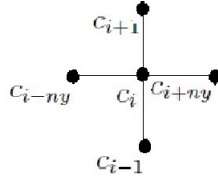


Figura 17 – Pontos do interior da malha.

$$\begin{aligned}
\frac{c_i^{k+1} - c_i^k}{\Delta t} &= \alpha \left(\frac{c_{i-ny}^{k+1} - 2c_i^{k+1} + c_{i+ny}^{k+1} + c_{i-ny}^k - 2c_i^k + c_{i+ny}^k}{2(\Delta x)^2} \right) + \\
&+ \alpha \left(\frac{c_{i-1}^{k+1} - 2c_i^{k+1} + c_{i+1}^{k+1} + c_{i-1}^k - 2c_i^k + c_{i+1}^k}{2(\Delta y)^2} \right) - u \left(\frac{c_{i+ny}^{k+1} - c_{i-ny}^{k+1} + c_{i+ny}^k - c_{i-ny}^k}{4\Delta x} \right) + \\
&- v \left(\frac{c_{i+1}^{k+1} - c_{i-1}^{k+1} + c_{i+1}^k - c_{i-1}^k}{4\Delta y} \right) - \mu \left(\frac{c_{i+1}^{k+1} + c_i^k}{2} \right) + f_i^{k+\frac{1}{2}}. \\
c_i^{k+1} - c_i^k &= \alpha \left(\frac{c_{i-ny}^{k+1} - 2c_i^{k+1} + c_{i+ny}^{k+1}}{2(\Delta x)^2} + \frac{c_{i-1}^{k+1} - 2c_i^{k+1} + c_{i+1}^{k+1}}{2(\Delta y)^2} \right) \Delta t + \\
&+ \alpha \left(\frac{c_{i-ny}^k - 2c_i^k + c_{i+ny}^k}{2(\Delta x)^2} + \frac{c_{i-1}^k - 2c_i^k + c_{i+1}^k}{2(\Delta y)^2} \right) \Delta t + \\
&- u \left(\frac{c_{i+ny}^{k+1} - c_{i-ny}^{k+1}}{4\Delta x} \right) \Delta t - u \left(\frac{c_{i+ny}^k - c_{i-ny}^k}{4\Delta x} \right) \Delta t - v \left(\frac{c_{i+1}^{k+1} - c_{i-1}^{k+1}}{4\Delta y} \right) \Delta t + \\
&- v \left(\frac{c_{i+1}^k - c_{i-1}^k}{4\Delta y} \right) \Delta t - \frac{\mu}{2} c_i^{k+1} \Delta t - \frac{\mu}{2} c_i^k \Delta t + f_i^{k+\frac{1}{2}} \Delta t.
\end{aligned}$$

$$\begin{aligned}
c_i^{k+1} - \alpha \left(\frac{c_{i-ny}^{k+1} - 2c_i^{k+1} + c_{i+ny}^{k+1}}{2(\Delta x)^2} + \frac{c_{i-1}^{k+1} - 2c_i^{k+1} + c_{i+1}^{k+1}}{2(\Delta y)^2} \right) \Delta t + u \left(\frac{c_{i+ny}^{k+1} - c_{i-ny}^{k+1}}{4\Delta x} \right) \Delta t + \\
+ v \left(\frac{c_{i+1}^{k+1} - c_{i-1}^{k+1}}{4\Delta y} \right) \Delta t + \frac{\mu}{2} c_i^{k+1} \Delta t = \\
= c_i^k + \alpha \left(\frac{c_{i-ny}^k - 2c_i^k + c_{i+ny}^k}{2(\Delta x)^2} + \frac{c_{i-1}^k - 2c_i^k + c_{i+1}^k}{2(\Delta y)^2} \right) \Delta t - u \left(\frac{c_{i+ny}^k - c_{i-ny}^k}{4\Delta x} \right) \Delta t + \\
- v \left(\frac{c_{i+1}^k - c_{i-1}^k}{4\Delta y} \right) \Delta t - \frac{\mu}{2} c_i^k \Delta t + f_i^{k+\frac{1}{2}} \Delta t. \\
c_{i-1}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta y)^2} - \frac{v \Delta t}{4\Delta y} \right) + c_{i-ny}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta x)^2} - \frac{u \Delta t}{4\Delta x} \right) + c_i^{k+1} \left(1 + \frac{\alpha \Delta t}{(\Delta x)^2} + \frac{\alpha \Delta t}{(\Delta y)^2} + \frac{\mu}{2} \Delta t \right) \\
+ c_{i+1}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta y)^2} + \frac{v \Delta t}{4\Delta y} \right) + c_{i+ny}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta x)^2} + \frac{u \Delta t}{4\Delta x} \right) = \\
c_{i-1}^k \left(\frac{\alpha \Delta t}{2(\Delta y)^2} + \frac{v \Delta t}{4\Delta y} \right) + c_{i-ny}^k \left(\frac{\alpha \Delta t}{2(\Delta x)^2} + \frac{u \Delta t}{4\Delta x} \right) + c_i^k \left(1 - \frac{\alpha \Delta t}{(\Delta x)^2} - \frac{\alpha \Delta t}{(\Delta y)^2} - \frac{\mu}{2} \Delta t \right) + \\
+ c_{i+1}^k \left(\frac{\alpha \Delta t}{2(\Delta y)^2} - \frac{v \Delta t}{4\Delta y} \right) + c_{i+ny}^k \left(\frac{\alpha \Delta t}{2(\Delta x)^2} - \frac{u \Delta t}{4\Delta x} \right) + f_i^{k+\frac{1}{2}} \Delta t. \quad (4.10)
\end{aligned}$$

Na forma matricial, segue que:

$$Ac^{k+1} = Bc^k + f_i^{k+\frac{1}{2}} \Delta t \quad (4.11)$$

onde os termos das matrizes esparsas A e B estão identificados em (4.10), sendo c^k o vetor de componentes $c_1^k, c_2^k, \dots, c_{nn}^k$ e ainda $f_i^{k+\frac{1}{2}}$ é o vetor de componentes $f_{1,2}^{k+\frac{1}{2}}, \dots, f_{nn}^{k+\frac{1}{2}}$. As matrizes A e B possuem, cada uma, uma diagonal principal e outras quatro diagonais secundárias, distintas da seguinte maneira:

$dpe = 1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\mu}{2}\Delta t$ é a Diagonal principal da matriz A^{k+1} ;

$dspie = -\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y}$ é a Diagonal secundária inferior da matriz A^{k+1} ;

$dspse = -\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y}$ é a Diagonal secundária superior da matriz A^{k+1} ;

$dsdie = -\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x}$ é a Diagonal secundária distante inferior da matriz A^{k+1} ;

$dsdse = -\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x}$ é a Diagonal secundária distante superior da matriz A^{k+1} ;

$dpd = 1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\mu}{2}\Delta t$ é a Diagonal principal da matriz B^k ;

$dspid = \frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y}$ é a Diagonal secundária inferior da matriz B^k ;

$dspsd = \frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y}$ é a Diagonal secundária superior da matriz B^k ;

$dsdid = \frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x}$ é a Diagonal secundária distante inferior da matriz B^k ;

$dsdsd = \frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x}$ é a Diagonal secundária distante superior da matriz B^k ;

4.1.4 Pontos da fronteira de Ω_1

Iremos utilizar condições de contorno de Dirichlet homogênea na fronteira $\partial\Omega_1$, não homogênea em $\partial\Omega_2$ e Robin nas fronteiras $\partial\Omega_3$ e $\partial\Omega_4$ (ver figura 15).

onde $g(x, y, t) = \kappa$ é uma função constante e $c = c(x, y, t)$ é a função de concentração de poluentes que estamos utilizando neste trabalho, discretizada por c_i^k com $i = 1, \dots, nn$, $k = 1, \dots, npt$ e npt é o número de passos no tempo.

4.1.4.1 Pontos da fronteira acima

Observemos a figura (19), referente aos pontos que existem no domínio Ω_1 :

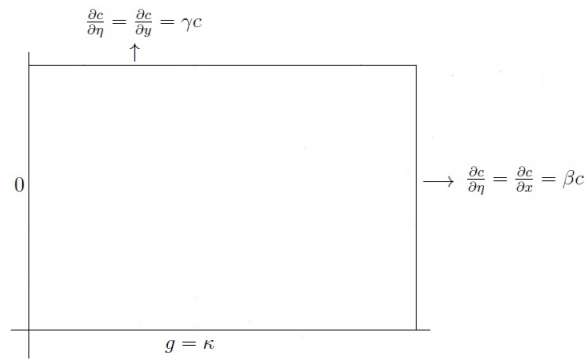


Figura 18 – Condições de Contorno.

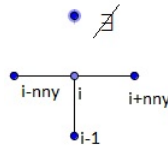


Figura 19 – Pontos da fronteira acima.

Na fronteira acima de Ω_1 , temos uma condição de contorno de Robin, então $\frac{\partial c}{\partial \eta} = \frac{\partial c}{\partial y} = \gamma c$ e, neste caso, temos um ponto (c_{i+1}) que não pertence ao domínio Ω_1 (ver Figura 19), daí fazemos as seguinte operações

$$\frac{\partial c}{\partial y} = \frac{c_{\bar{i}}^{k+1} - c_{i-1}^{k+1} + c_{\bar{i}}^k - c_{i-1}^k}{4\Delta y}$$

$$c_{\bar{i}}^{k+1} + c_{\bar{i}}^k = 4\Delta y \gamma c_i + c_{i-1}^{k+1} + c_{i-1}^k$$

$$c_{\bar{i}}^{k+1} + c_{\bar{i}}^k = 4\Delta y \gamma \left(\frac{c_i^{k+1} + c_i^k}{2} \right) + c_{i-1}^{k+1} + c_{i-1}^k$$

$$c_{\bar{i}}^{k+1} + c_{\bar{i}}^k = 2\Delta y \gamma c_i^{k+1} + 2\Delta y \gamma c_i^k + c_{i-1}^{k+1} + c_{i-1}^k. \tag{4.12}$$

Usando a equação (4.12) na equação (4.10), segue-se que a discretização do domínio Ω_1 na fronteira acima, com condição de contorno de Robin, fica da seguinte maneira:

$$\begin{aligned}
 & c_{i-1}^{k+1} \left(-\frac{\alpha\Delta t}{(\Delta y)^2} \right) + c_{i-ny}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) + \\
 & + c_i^{k+1} \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\gamma\Delta t}{\Delta y} + \frac{v\gamma\Delta t}{2} + \frac{\mu}{2}\Delta t \right) + c_{i+ny}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) = \\
 & = c_{i-1}^k \left(\frac{\alpha\Delta t}{(\Delta y)^2} \right) + c_{i-ny}^k \left(\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) + \\
 & + c_i^k \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\gamma\Delta t}{\Delta y} - \frac{v\gamma\Delta t}{2} - \frac{\mu}{2}\Delta t \right) + \\
 & + c_{i+ny}^k \left(\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) + f_i^{k+\frac{1}{2}} \Delta t.
 \end{aligned}$$

4.1.4.2 Pontos da fronteira à direita

Observe a figura (20), referente aos pontos que existem na fronteira $\partial\Omega_3$ do domínio Ω_1 .

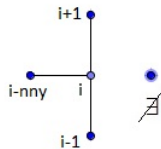


Figura 20 – Pontos da fronteira à direita.

Na fronteira à direita do domínio ($\partial\Omega_3$), temos uma condição de contorno de Robin, então $\frac{\partial c}{\partial \eta} = \frac{\partial c}{\partial x} = \beta c$, e neste caso, temos um ponto à direita da fronteira que também não pertence a Ω_1 : $c_{\frac{\beta}{2}}^k$ (ver Figura 20), daí utilizando as equações (4.4) e (4.6):

$$\frac{c_{\frac{\beta}{2}}^{k+1} - c_{i-ny}^{k+1} + c_{\frac{\beta}{2}}^k - c_{i-ny}^k}{4\Delta x} = \beta \left(\frac{c_i^{k+1} + c_i^k}{2} \right)$$

$$c_{\frac{\beta}{2}}^{k+1} + c_{\frac{\beta}{2}}^k = 2\Delta x \beta c_i^{k+1} + 2\Delta x \beta c_i^k + c_{i-ny}^{k+1} + c_{i-ny}^k \quad (4.13)$$

Usando a equação (4.13) na equação (4.10), temos a discretização do domínio Ω_1 na fronteira à direita, com condição de contorno de Robin:

$$\begin{aligned} & c_{i-1}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) + c_{i-ny}^{k+1} \left(-\frac{\alpha\Delta t}{(\Delta x)^2} \right) + \\ & + c_i^{k+1} \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\beta\Delta t}{\Delta x} + \frac{u\beta\Delta t}{2} + \frac{\mu}{2}\Delta t \right) + c_{i+1}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) = \\ & = c_{i-1}^k \left(\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) + c_{i-ny}^k \left(\frac{\alpha\Delta t}{(\Delta x)^2} \right) \\ & + c_i^k \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\beta\Delta t}{\Delta x} - \frac{u\beta\Delta t}{2} - \frac{\mu}{2}\Delta t \right) + c_{i+1}^k \left(\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) + f_i^{k+\frac{1}{2}} \Delta t. \end{aligned}$$

4.1.4.3 Pontos da fronteira à esquerda

A figura (21), refere-se aos pontos que existem na fronteira à esquerda do domínio Ω_1

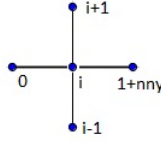


Figura 21 – Pontos da fronteira à esquerda.

Na fronteira à esquerda do domínio, temos uma condição de contorno de Dirichlet homogênea, então todos os pontos são iguais a 0. Usando esta condição na equação (4.10), segue que a discretização do domínio Ω_1 nesta fronteira será como se segue:

$$\begin{aligned}
 & c_{i-1}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta y)^2} - \frac{v \Delta t}{4\Delta y} \right) + c_i^{k+1} \left(1 + \frac{\alpha \Delta t}{(\Delta x)^2} + \frac{\alpha \Delta t}{(\Delta y)^2} + \frac{\mu}{2} \Delta t \right) + c_{i+1}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta y)^2} + \frac{v \Delta t}{4\Delta y} \right) + \\
 & + c_{i+ny}^{k+1} \left(-\frac{\alpha \Delta t}{2(\Delta x)^2} + \frac{u \Delta t}{4\Delta x} \right) = c_{i-1}^k \left(\frac{\alpha \Delta t}{2(\Delta y)^2} + \frac{v \Delta t}{4\Delta y} \right) + c_i^k \left(1 - \frac{\alpha \Delta t}{(\Delta x)^2} - \frac{\alpha \Delta t}{(\Delta y)^2} - \frac{\mu}{2} \Delta t \right) + \\
 & + c_{i+1}^k \left(\frac{\alpha \Delta t}{2(\Delta y)^2} - \frac{v \Delta t}{4\Delta y} \right) + c_{i+ny}^k \left(\frac{\alpha \Delta t}{(\Delta x)^2} - \frac{u \Delta t}{2\Delta x} \right) + f_i^{k+\frac{1}{2}} \Delta t.
 \end{aligned}$$

4.1.4.4 Pontos da fronteira abaixo

Observando a (Figura 22), referente aos pontos que existem na fronteira inferior $\partial\Omega_2$ do domínio Ω_1 , temos:

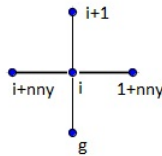


Figura 22 – Pontos da fronteira inferior.

Na fronteira na parte de baixo do domínio, temos uma condição de contorno de Dirichlet não homogênea, em que todos os pontos são iguais a uma função $g(x, y, t) = \kappa$ constante dada. Usando esta condição na equação (4.10), segue que a discretização do domínio Ω_1 nesta fronteira é dada como se segue:

$$\begin{aligned}
& g \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) + c_{i-ny}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) + c_i^{k+1} \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\mu}{2}\Delta t \right) + \\
& + c_{i+1}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) + c_{i+ny}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) = g \left(\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) + \\
& + c_{i-ny}^k \left(\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) + c_i^k \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\mu}{2}\Delta t \right) + c_{i+1}^k \left(\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) + \\
& + c_{i+ny}^k \left(\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) + f_i^{k+\frac{1}{2}}\Delta t.
\end{aligned}$$

Então, a discretização do domínio Ω_1 na fronteira abaixo fica da seguinte maneira:

$$\begin{aligned}
& c_{i-ny}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) + c_i^{k+1} \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\mu}{2}\Delta t \right) + \\
& + c_{i+1}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) + c_{i+ny}^{k+1} \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) = g \left(\frac{\alpha\Delta t}{(\Delta y)^2} + \frac{v\Delta t}{2\Delta y} \right) + \\
& + c_{i-ny}^k \left(\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) + c_i^k \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\mu}{2}\Delta t \right) + c_{i+1}^k \left(\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) + \\
& + c_{i+ny}^k \left(\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) + f_i^{k+\frac{1}{2}}\Delta t.
\end{aligned}$$

4.1.5 Procedimento Numérico para Ω_2

Para que possamos atingir nossos objetivos, que consistem em estudar o melhor local para fazer o tratamento de esgoto, devemos considerar não só o domínio Ω_1 com uma fonte pontual sobre o retângulo que aproxima o rio Jordão. Para isso, olhamos o problema de uma nova perspectiva e construímos um novo domínio Ω_2 , que agora aproxima o rio Jordão com outro retângulo que representa o córrego Brejo Alegre. Agora não temos mais uma fonte pontual ao longo da margem acima de rio: nossa fonte é todo o córrego, apresentando uma certa concentração de poluente, velocidade de correnteza e todas as características físicas que são consideradas na primeira abordagem, referente apenas ao rio Jordão.

A equação de difusão-advecção-reação permanece a mesma, mas agora com mais condições de contorno, como se pode ver na equação (4.14) e o domínio trabalhado segue na figura 23

$$\left\{ \begin{array}{l} c = c(x, y, t), \quad \text{com} \quad (x, y) \in \Omega \subset R^2 \quad \text{e} \quad t \in J = (0, T], \\ \frac{\partial c}{\partial t} - \text{div}(\alpha \nabla c - c \mathbf{v}) = f - \mu c, \quad \text{e} \\ C(x, y, 0) = C_0(x, y), \quad \forall (x, y) \in \Omega \\ \frac{\partial c}{\partial \eta} \Big|_{\partial \Omega_{2_i}} = \gamma_i c \quad \text{com} \quad i = 1, \dots, 6 \\ \frac{\partial c}{\partial \eta} \Big|_{\partial \Omega_{2_i}} = g c \quad \text{com} \quad g(x, y, t) = \text{constante} \quad \text{e} \quad i = 7 \quad \text{e} \quad 8. \end{array} \right. \quad (4.14)$$

sendo estes subíndices $\partial \Omega_{2_i}$ e de γ_i referentes a localização de diferentes trechos da fronteira total Ω_2 em que ocorrem fenômenos de tipo ou de intensidades diferentes. Além disso, α , \mathbf{v} , μ e f são os mesmos da equação (2.1).

Considerando o domínio $\Omega_2 \subset R^2$, descrito como visto na figura (23), onde $\partial \Omega_{2_1}$, $\partial \Omega_{2_2}$, $\partial \Omega_{2_3}$, $\partial \Omega_{2_4}$, $\partial \Omega_{2_5}$, $\partial \Omega_{2_6}$, $\partial \Omega_{2_7}$ e $\partial \Omega_{2_8}$ são as fronteiras do Domínio Ω_2 . Dividindo o intervalo $[0, l_1]$ em $Nx = nx_1 + nx_2 + nx_3$ subintervalos, obtemos $\Delta x = \frac{l}{Nx}$ e de maneira análoga, $\Delta y = \frac{h}{ny}$, com $ny = ny_2$ são os subintervalos de $[0, h]$ (ver figura 24).

Para enumerar os nós obtidos na malha decorrente da discretização, iremos adotar uma sequência nos intervalos verticais sucessivos, como podemos ver na figura (24).

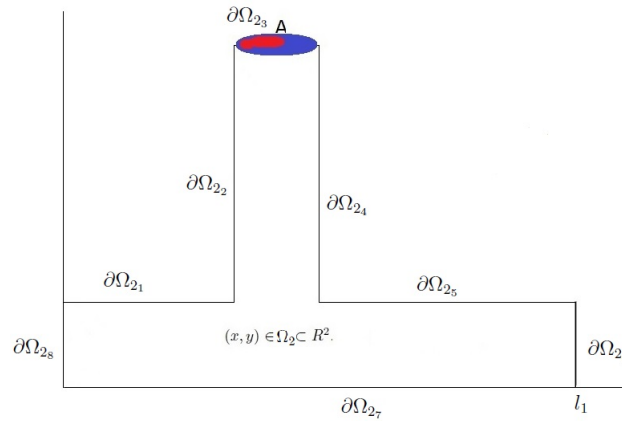


Figura 23 – Novo domínio Ω_2 .

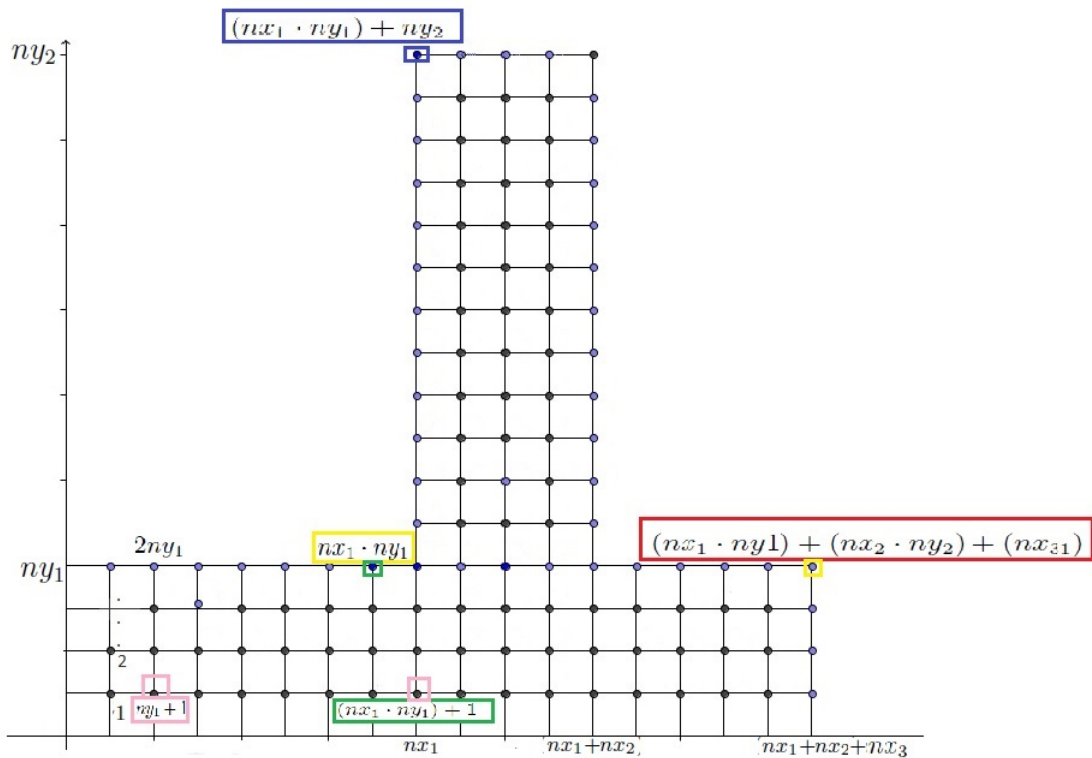


Figura 24 – Discretização do domínio Ω_2 .

De maneira análoga ao domínio Ω_1 a numeração da malha não começa na primeira coluna e nem abrange a primeira linha.

A discretização dos pontos interiores de Ω_2 é igual à discretização dos pontos interiores de Ω_1 (ver equação 4.10), diferindo apenas nas condições de fronteira, que é o que indicaremos apenas enunciando as fórmulas finais, seguindo o procedimento já adotado em 4.1.4.

4.1.5.1 Fronteira acima ($\partial\Omega_{2_1}$) de Ω_2 :

$$\begin{aligned} & \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) c_{i-ny}^{k+1} + \left(-\frac{\alpha\Delta t}{(\Delta y)^2} \right) c_{i-1}^{k+1} + \\ & \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\gamma_1\Delta t}{\Delta y} + \frac{v\gamma_1\Delta t}{2} + \frac{\mu\Delta t}{2} \right) c_i^{k+1} + \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) c_{i+ny}^{k+1} = \\ & \left(\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x} \right) c_{i-ny}^k + \left(\frac{\alpha\Delta t}{(\Delta y)^2} \right) c_{i-1}^k + \\ & \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\gamma_1\Delta t}{\Delta y} - \frac{v\gamma_1\Delta t}{2} - \frac{\mu\Delta t}{2} \right) c_i^k + \left(\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) c_{i+ny}^k + \Delta t f_i^{k+\frac{1}{2}}. \end{aligned}$$

4.1.5.2 Fronteira à esquerda do córrego ($\partial\Omega_{2_2}$) de Ω_2 :

$$\begin{aligned} & \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) c_{i-1}^{k+1} + \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\gamma_2\Delta t}{\Delta x} - \frac{u\gamma_2\Delta t}{2\Delta x} + \frac{\mu\Delta t}{2} \right) c_i^{k+1} \\ & + \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) c_{i+1}^{k+1} + \left(-\frac{\alpha\Delta t}{(\Delta x)^2} + \frac{u\Delta t}{2\Delta x} \right) c_{i+ny}^{k+1} = \\ & \left(\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) c_{i-1}^k + \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\gamma_2\Delta t}{\Delta x} + \frac{u\gamma_2\Delta t}{2\Delta x} - \frac{\mu\Delta t}{2} \right) c_i^k \\ & + \left(\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) c_{i+1}^k + \left(\frac{\alpha\Delta t}{(\Delta x)^2} - \frac{u\Delta t}{2\Delta x} \right) c_{i+ny}^k + \Delta t f_i^{k+\frac{1}{2}}. \end{aligned}$$

4.1.5.3 Fronteira acima do córrego ($\partial\Omega_{2_3}$) de Ω_2 :

$$\left(-\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x} \right) c_{i-ny}^{k+1} + \left(-\frac{\alpha\Delta t}{(\Delta y)^2} \right) c_{i-1}^{k+1} +$$

$$\begin{aligned} & \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\gamma_3\Delta t}{\Delta y} + \frac{v\gamma_3\Delta t}{2} + \frac{\mu\Delta t}{2}\right) c_i^{k+1} + \left(-\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x}\right) c_{i+ny}^{k+1} = \\ & \left(\frac{\alpha\Delta t}{2(\Delta x)^2} + \frac{u\Delta t}{4\Delta x}\right) c_{i-ny}^k + \left(\frac{\alpha\Delta t}{(\Delta y)^2}\right) c_{i-1}^k + \\ & \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\gamma_3\Delta t}{\Delta y} - \frac{v\gamma_3\Delta t}{2} - \frac{\mu\Delta t}{2}\right) c_i^k + \left(\frac{\alpha\Delta t}{2(\Delta x)^2} - \frac{u\Delta t}{4\Delta x}\right) c_{i+ny}^k + \Delta t f_i^{k+\frac{1}{2}}. \end{aligned}$$

4.1.5.4 Fronteira à direita do córrego ($\partial\Omega_{2_4}$) de Ω_2 :

$$\begin{aligned} & \left(-\frac{\alpha\Delta t}{(\Delta x)^2}\right) c_{i-ny}^{k+1} + \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y}\right) c_{i-1}^{k+1} + \\ & + \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\gamma_4\Delta t}{\Delta x} + \frac{u\gamma_4\Delta t}{2\Delta x} + \frac{\mu\Delta t}{2}\right) c_i^{k+1} + \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y}\right) c_{i+1}^{k+1} = \\ & \left(\frac{\alpha\Delta t}{(\Delta x)^2}\right) c_{i-ny}^k + \left(\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y}\right) c_{i-1}^k + \\ & + \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\gamma_4\Delta t}{\Delta x} - \frac{u\gamma_4\Delta t}{2\Delta x} - \frac{\mu\Delta t}{2}\right) c_i^k + \left(\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y}\right) c_{i+1}^k + \Delta t f_i^{k+\frac{1}{2}}. \end{aligned}$$

4.1.5.5 Fronteira acima ($\partial\Omega_{2_5}$) de Ω_2 :

Consideremos, na fronteira $\partial\Omega_{2_5}$, $\frac{\partial c}{\partial \eta} = \gamma_5 c \Rightarrow \frac{\partial c}{\partial y} = \gamma_5 c$, então, segue-se que a discretização da malha nos pontos da fronteira $\partial\Omega_{2_5}$ será (usando a mesma notação anteriormente adotado):

$$\frac{c_{\bar{x}}^{k+1} - c_{i-1}^{k+1} + c_{\bar{x}}^k - c_{i-1}^k}{4\Delta y} = \gamma_5 \left(\frac{c_{i-1}^{k+1} + c_i^k}{2}\right) \Rightarrow$$

↓

$$c_{\beta}^{k+1} + c_{\beta}^k = 2\gamma_5 \Delta y c_i^{k+1} + 2\gamma_5 \Delta y c_i^k + c_{i-1}^{k+1} + c_{i-1}^k$$

Portanto,

$$\begin{aligned} & \left(-\frac{\alpha \Delta t}{2(\Delta x)^2} - \frac{u \Delta t}{4\Delta x} \right) c_{i-ny}^{k+1} + \left(-\frac{\alpha \Delta t}{(\Delta y)^2} \right) c_{i-1}^{k+1} + \\ & \left(1 + \frac{\alpha \Delta t}{(\Delta x)^2} + \frac{\alpha \Delta t}{(\Delta y)^2} - \frac{\alpha \gamma_5 \Delta t}{\Delta y} + \frac{v \gamma_5 \Delta t}{2} + \frac{\mu \Delta t}{2} \right) c_i^{k+1} + \left(-\frac{\alpha \Delta t}{2(\Delta x)^2} + \frac{u \Delta t}{4\Delta x} \right) c_{i+ny}^{k+1} = \\ & \left(\frac{\alpha \Delta t}{2(\Delta x)^2} + \frac{u \Delta t}{4\Delta x} \right) c_{i-ny}^k + \left(\frac{\alpha \Delta t}{(\Delta y)^2} \right) c_{i-1}^k + \\ & \left(1 - \frac{\alpha \Delta t}{(\Delta x)^2} - \frac{\alpha \Delta t}{(\Delta y)^2} + \frac{\alpha \gamma_5 \Delta t}{\Delta y} - \frac{v \gamma_5 \Delta t}{2} - \frac{\mu \Delta t}{2} \right) c_i^k + \left(\frac{\alpha \Delta t}{2(\Delta x)^2} - \frac{u \Delta t}{4\Delta x} \right) c_{i+ny}^k + \Delta t f_i^{k+\frac{1}{2}}. \end{aligned}$$

4.1.5.6 Fronteira à direita do rio ($\partial\Omega_{2_6}$) de Ω_2 :

Na fronteira $\partial\Omega_{2_6}$ temos $\frac{\partial c}{\partial \eta} = \gamma_6 c \Rightarrow \frac{\partial c}{\partial x} = \gamma_6 c$ e, então, segue-se que a discretização da malha nos pontos da fronteira $\partial\Omega_{2_6}$ será:

$$\frac{c_{\beta}^{k+1} - c_{i-ny}^{k+1} + c_{\beta}^k - c_{i-ny}^k}{4\Delta x} = \gamma_6 \left(\frac{c_i^{k+1} + c_i^k}{2} \right)$$

↓

$$c_{\beta}^{k+1} + c_{\beta}^k = 2\gamma_6 \Delta x c_i^{k+1} + 2\gamma_6 \Delta x c_i^k + c_{i-ny}^{k+1} + c_{i-ny}^k$$

Portanto,

$$\begin{aligned}
& \left(-\frac{\alpha\Delta t}{(\Delta x)^2} \right) c_{i-ny}^{k+1} + \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) c_{i-1}^{k+1} + \\
& + \left(1 + \frac{\alpha\Delta t}{(\Delta x)^2} + \frac{\alpha\Delta t}{(\Delta y)^2} - \frac{\alpha\gamma_6\Delta t}{\Delta x} + \frac{u\gamma_6\Delta t}{2\Delta x} + \frac{\mu\Delta t}{2} \right) c_i^{k+1} + \left(-\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) c_{i+1}^{k+1} = \\
& \left(\frac{\alpha\Delta t}{(\Delta x)^2} \right) c_{i-ny}^k + \left(\frac{\alpha\Delta t}{2(\Delta y)^2} + \frac{v\Delta t}{4\Delta y} \right) c_{i-1}^k + \\
& + \left(1 - \frac{\alpha\Delta t}{(\Delta x)^2} - \frac{\alpha\Delta t}{(\Delta y)^2} + \frac{\alpha\gamma_6\Delta t}{\Delta x} - \frac{u\gamma_6\Delta t}{2\Delta x} - \frac{\mu\Delta t}{2} \right) c_i^k + \left(\frac{\alpha\Delta t}{2(\Delta y)^2} - \frac{v\Delta t}{4\Delta y} \right) c_{i+1}^k + \Delta t f_i^{k+\frac{1}{2}}.
\end{aligned}$$

Observamos que as fronteiras Ω_7 e Ω_8 são as mesmas fronteiras, em que, no domínio Ω_1 $\frac{\partial c}{\partial \eta}|_{\partial\Omega_i} = gc$ com $g(x, y, t) = \text{constante}$ e $i = 7, 8$.

5 Capitulo 5

Neste capítulo apresentamos alguns cenários resultantes da simulação computacional do modelo. Os algoritmos foram programados no ambiente Matlab, os quais podem ser encontrados nos anexos A e B.

Para o estudo, separamos o domínio Ω_1 e Ω_2 em 3 cenários cada, onde o primeiro cenário de Ω_1 (25 e 26) corresponde nas figuras seguintes (27 a 29) ao primeiro cenário de Ω_2 ; o segundo cenário, que é um controle urbano da poluição, em Ω_1 corresponde ao segundo cenário de Ω_2 e o terceiro cenário respectivamente (30 a 32) e (33 a ??) acontece da mesma maneira. É importante ressaltar que os valores dos parâmetros são os mesmos em todos os cenários, exceto os valores das fontes, que serão explicados mais à frente.

5.1 Simulações e Análise dos Resultados

O rio Jordão tem, no córrego Brejo Alegre, uma fonte de emissão de poluente, e este córrego apresenta, por sua vez, várias possíveis fontes de emissão de poluentes, dos quais algumas são difusas, ou seja, não possuem um ponto de lançamento específico.

Mesmo sabendo da incidência permanente do problema de poluição do rio Jordão, não foi possível encontrar na literatura muitos trabalhos sobre o problema. O mais relevante dentre os poucos trabalhos foi (PAULA, 2011), “Avaliação da Qualidade da Água e Autodepuração do Rio Jordão, Araguari (MG)”. Mesmo sendo um trabalho relevante, não se obtiveram valores específicos para os parâmetros.

Assim, para o desenvolvimento do modelo matemático e para as simulações, os parâmetros foram tomados de pesquisas feitas por (GUACA, 2015), (ABREU, 2009) e (WOLMUTH, 2009), e alguns parâmetros foram estimados para manter as características das condições ambientais reais.

Para os parâmetros de velocidade tanto no rio quanto no córrego, utilizamos as ideias dos trabalhos de (OLIVEIRA, 2003) e (NOVAIS, 2011), em valores representativos dos ventos, tomados a 10 metros de altura. Pela equação de Ekman (OLIVEIRA, 2003), o valor superficial do vento que corresponde a correnteza superficial, é de 3% do valor do vento à 10 metros. Já o segundo trabalho disponibiliza esses valores em determinadas épocas do ano, como podemos observar na tabela 1.

Estação do ano	Mês	Velocidade Superficial (comprimento/tempo)
Verão	Dezembro	0,1575
	Janeiro	
	Fevereiro	
Outono	Março	0,1575
	Abril	
	Maio	
Inverno	Junho	0,18
	Julho	
	Agosto	
Primavera	Setembro	0,1125
	Outubro	
	Novembro	

Tabela 1 – Velocidade superficial (comprimento/tempo) dos ventos dominantes predominantes de Noroeste e Sudeste usando a fórmula de Ekman (valores estimados).

5.1.1 Primeiro cenário de Ω_1 e Ω_2

Neste cenário, estudamos a dispersão do poluente na direção de um vento de Noroeste.

Para esta simulação consideramos 6 fontes, sendo três delas representando as fontes que saem do córrego e entram no rio, e outras três fontes no rio representando as fontes de poluição oriundas da agricultura da região por *run-off*. A (Figura 25 e a Figura 26) apresentam 6 gráficos, para um tempo de simulação de 2 anos, em Ω_1 . A (Figura 27 e a Figura 28) apresentam outros 6 gráficos com o mesmo tempo de 2 anos, do domínio Ω_2 . As duas figuras mostram o comportamento evolutivo da poluição no domínio completo, Ω_2 .

A Figura 29 é o resultado final das simulações para o cenário neste domínio, com o mesmo número de iterações, mesmas velocidades (onde existem) e parâmetros. Os valores utilizados nas fontes são em relação ao IQA da água, regida pela função $f(x) = 1 - \frac{IQA_x}{100}$, onde x é o ponto da malha em que inserimos a fonte. Observe-se que, quanto mais próximo de 100 o IQA estiver, menos poluição teremos na fonte e, conseqüentemente, menor será a sua dispersão por todo o rio. Além disso, é importante ressaltar que, para as simulações, o valor do IQA em cada domínio foi diferente, visto que em Ω_1 o poluente já estava mais dissolvido do que em Ω_2 .

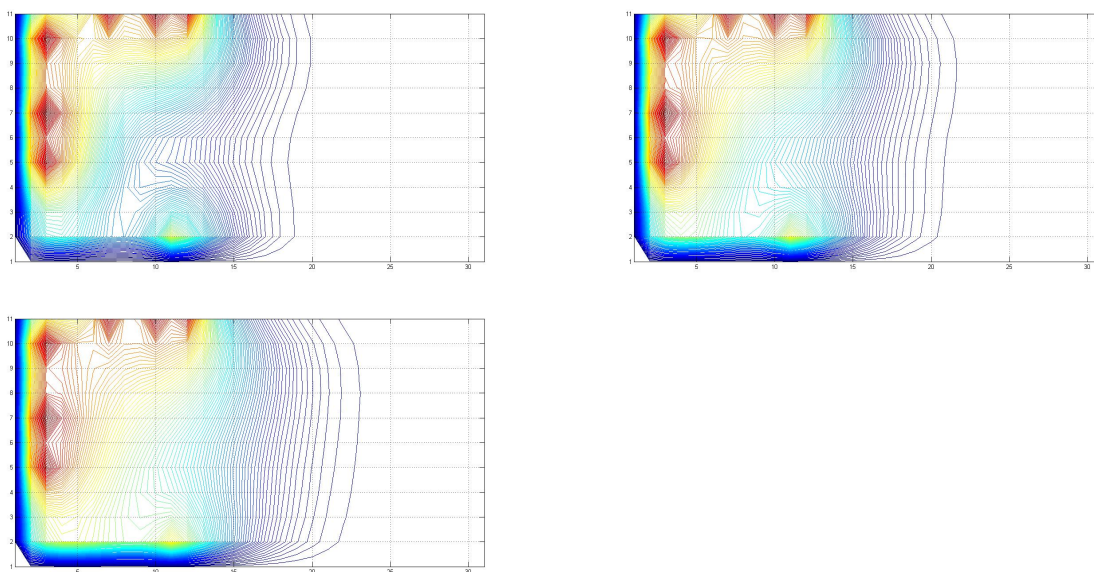


Figura 25 – Gráficos Referentes à dispersão de poluente em Ω_1 (365 dias) com 3 fontes onde o córrego Brejo Alegre deságua no rio Jordão e 3 fontes rio acima, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.

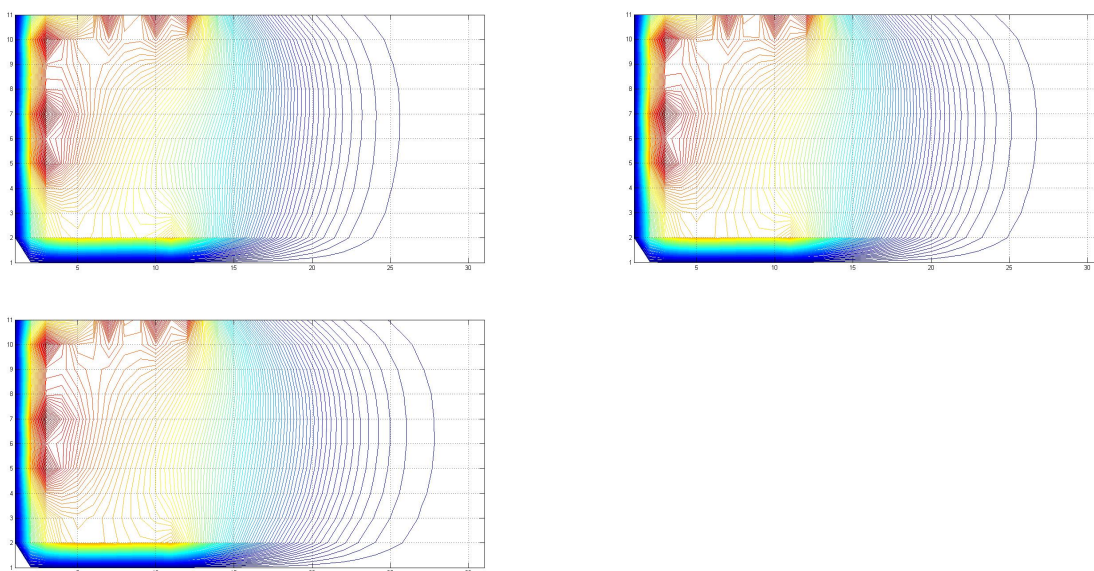


Figura 26 – Gráficos Referentes à dispersão de poluente em Ω_1 por mais 1 ano, no qual o antecessor é condição inicial para a próxima simulação, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.

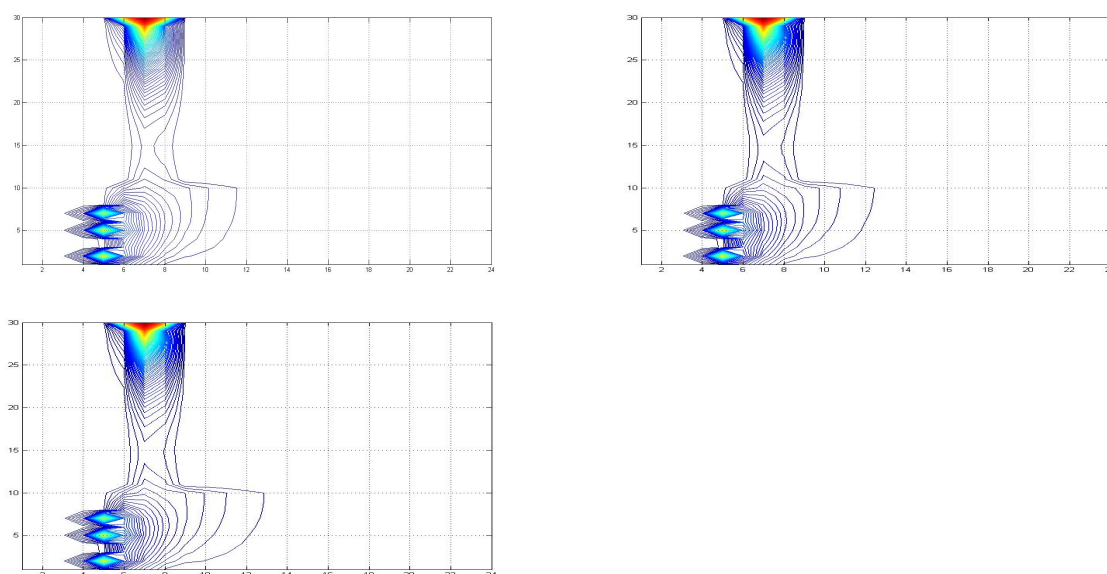


Figura 27 – Gráficos Referentes à dispersão de poluente em Ω_2 , com uma fonte no córrego Brejo Alegre, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.

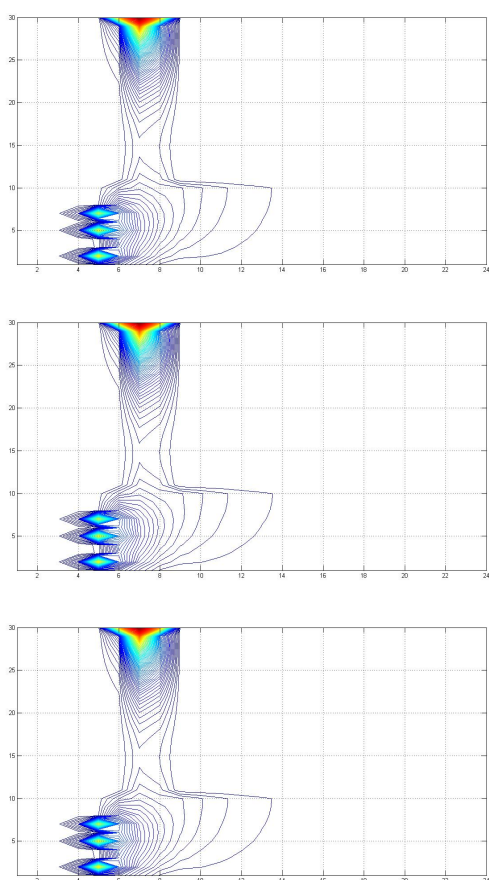


Figura 28 – Gráficos Referentes à dispersão de poluente em Ω_2 (365 dias), agora incluindo a dispersão no córrego Brejo Alegre com 3 fontes localizadas a montante do Rio Jordão e a montante do Córrego, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.

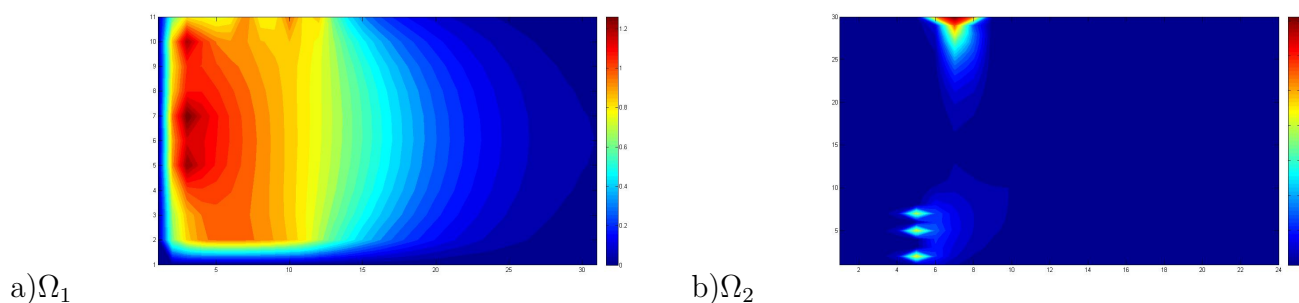


Figura 29 – Resultado de Ω_1 e Ω_2 depois de 2 anos de simulação.

Observamos depois das simulações que, mesmo considerando as fontes de poluentes de Ω_1 provocarem impactos mais diluídos na entrada do rio, podemos ver que o resultado, considerando o córrego como uma fonte constante de poluente, é melhor do que considerar apenas fontes pontuais no rio. Isto acontece porque ocorre uma absorção maior de poluente pelas margens e, além da correnteza do córrego ter um encontro com a correnteza do rio, ela contribui para o transporte, e assim, contribui para a dispersão do poluente. Com estas simulações, observamos então que o cenário de Ω_1 tem uma maior concentração de poluente, principalmente nos arredores das fontes, mas a contaminação do rio é mais demorada, ou seja, depois de 2 anos ainda existem lugares do rio que não foram atingidas pela poluição. Já no domínio Ω_2 , a concentração de poluente em cada ponto da malha é menor do que no caso do domínio anterior, também perto das fontes, mas a advecção do poluente também é maior, considerando que temos um aumento da velocidade total.

Agora que já temos um cenário que representa a situação atual do rio estudado, e podemos analisar a evolução do poluente em duas situações que achamos apropriadas. Na primeira, consideramos um tratamento da poluição pelas políticas públicas adotadas na cidade, considerando que não é financeiramente viável acabar com a fonte de poluição, mas sim diminuí-la. Já na segunda, consideramos um tratamento total da poluição acabando com a fonte poluidora do córrego. Além disso, é importante destacar que os resultados obtidos neste primeiro cenário foram utilizados como **condição inicial** para os cenários a seguir, aqui já comentados. Esses resultados serão visto a seguir. Em outras palavras, são cenários com políticas públicas de contingenciamento.

5.1.2 Cenário Ω_1 e Ω_2 - Controle Urbano Parcial

Neste cenário, modificamos o valor do IQA em todas as fontes de poluentes, representando um controle urbano de diminuição de emissão de poluente no córrego. Em Ω_1 fizemos uma redução do valor das fontes pontuais que representam a emissão do poluente feita pela cidade e, em Ω_2 , fizemos a mesma redução nas fontes vindas do córrego. Observamos nas figuras (Figura 30, Figura 31, Figura 32, Figura 33, Figura 34

e Figura 35) que houve uma diminuição do nível de concentração de poluição no rio a partir da situação inicial que é formada pelos níveis assintóticos gerados nos cenários anteriores. Pela Figura 32 observamos que a poluição, mesmo diminuindo, ainda está se dispersando ao longo do rio, enquanto que, na Figura 35, a poluição já se estabilizou e não há mais mudanças. Essas observações foram confirmadas ao fazermos diversas simulações, já que, em Ω_1 , mesmo após 3 anos de simulação, ainda conseguimos ver uma mudança de comportamento da poluição, ou seja, não houve uma estabilidade assintótica do resultado, enquanto que, em Ω_2 , conseguimos essa estabilidade dos resultados em menos de 3 anos.

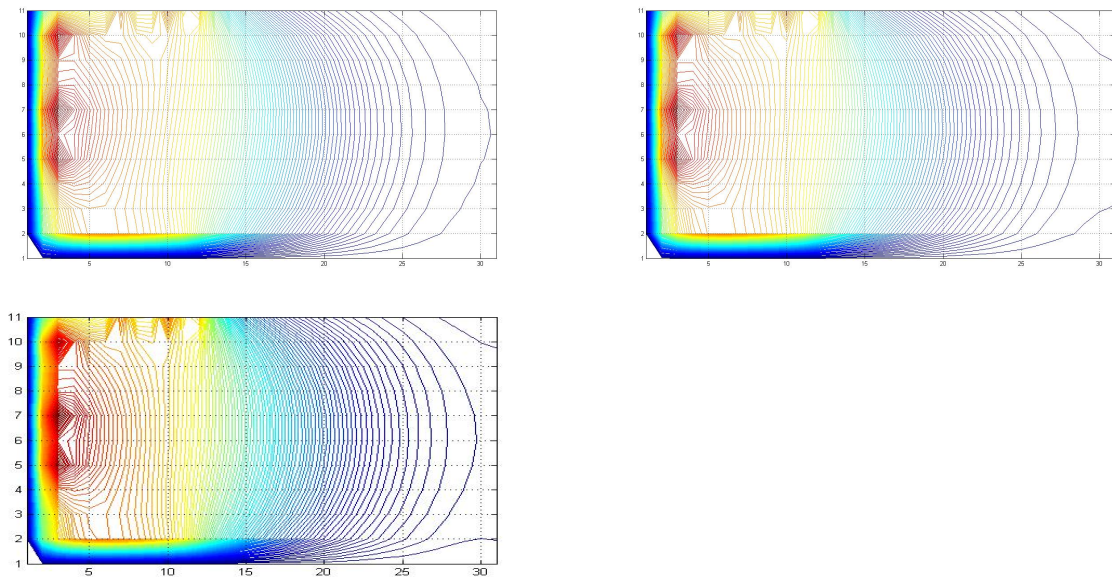


Figura 30 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_1 (primeiro ano).

É necessário comentar que, no cenário (5.1.2) figuras 35 a 33 pode ser observada uma difusão a montante das fontes, ainda que muito inferior àquele do trecho a jusante. Isto se deve, principalmente, às características predominante parabólicas do modelo em função de velocidades relativamente baixas do rio Jordão nesse trecho.

Esses resultados são bastante indicativos do potencial deste trabalho em função daquilo que podemos concluir quanto ao tratamento de água ao longo do córrego ser mais eficiente, em relação ao tempo, do que no encontro posterior do córrego com o rio.

E, por fim, para concluir as simulações, no cenário apresentado a seguir, consideramos um controle total da poluição vinda da cidade, ou seja, toda a fonte de poluição proveniente do córrego agora é considerada como sendo zero, permanecendo apenas as fontes provenientes da agricultura local nas margens a montante do rio Jordão.

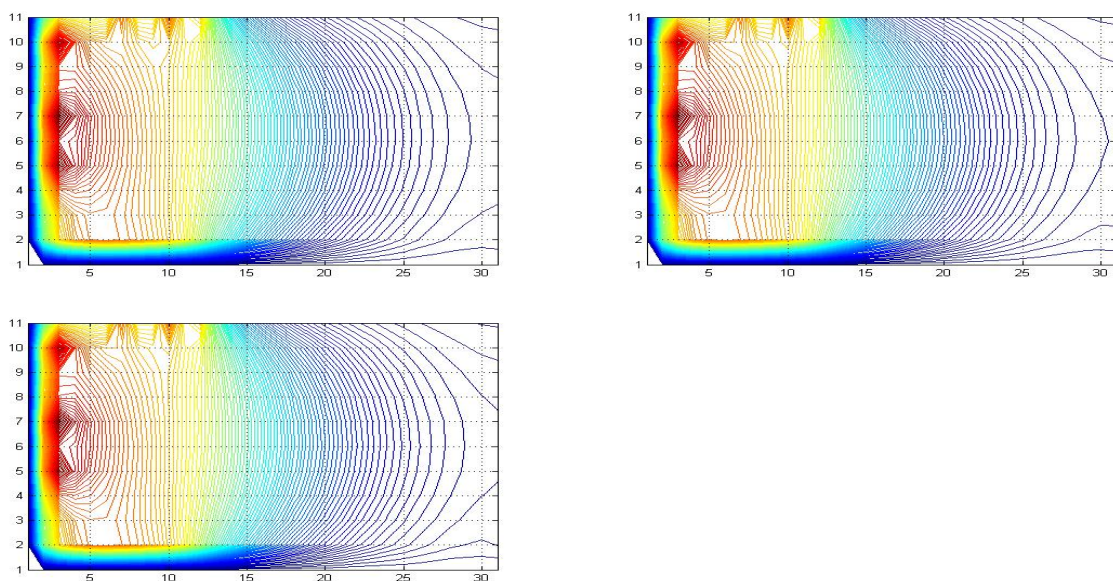


Figura 31 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_1 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$ respectivamente.

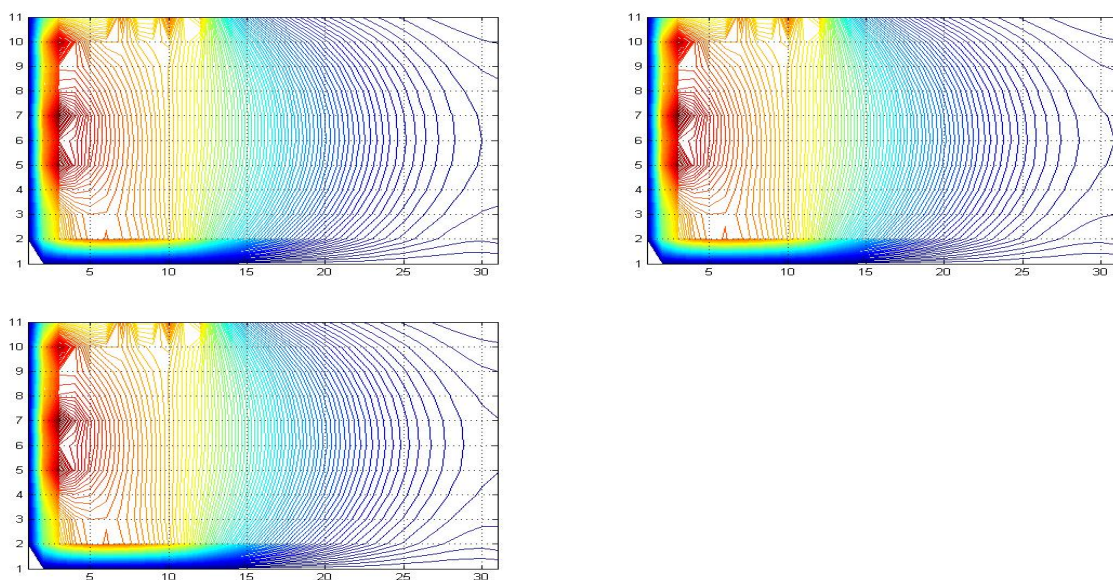


Figura 32 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_1 no terceiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

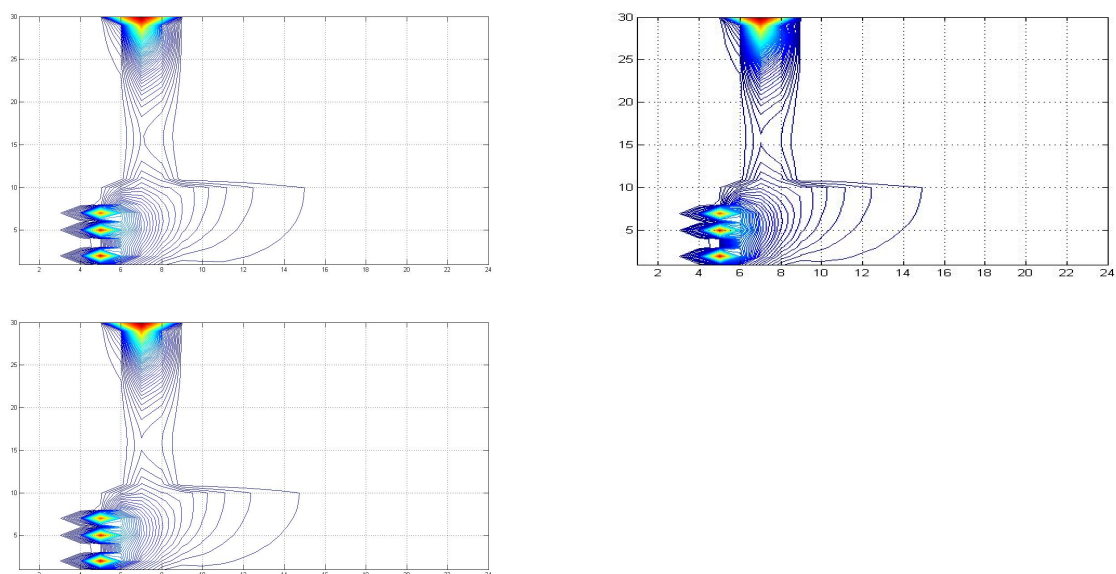


Figura 33 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_2 no primeiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

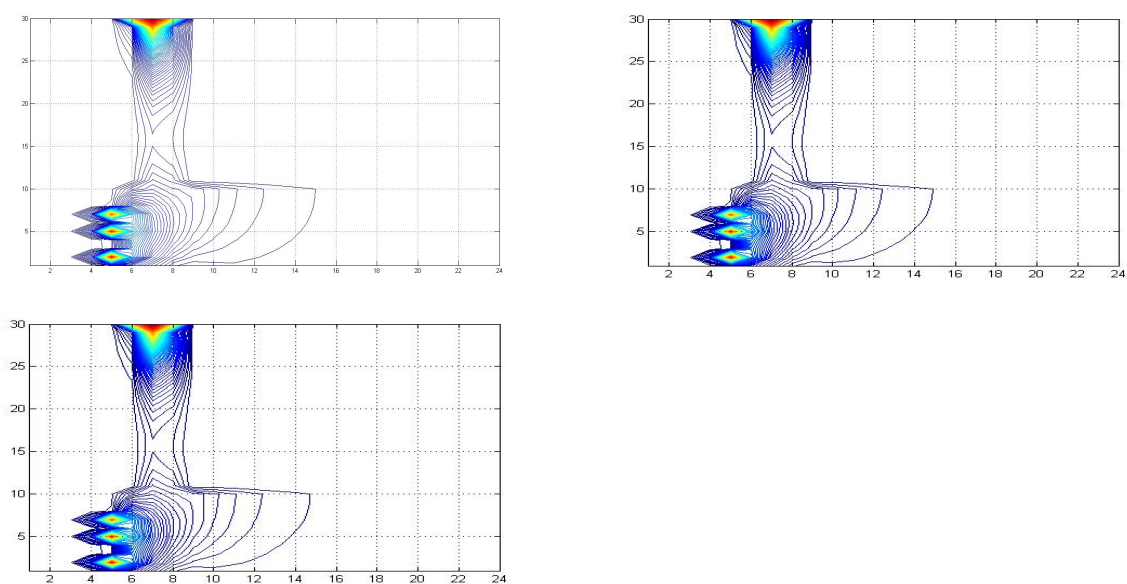


Figura 34 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_2 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

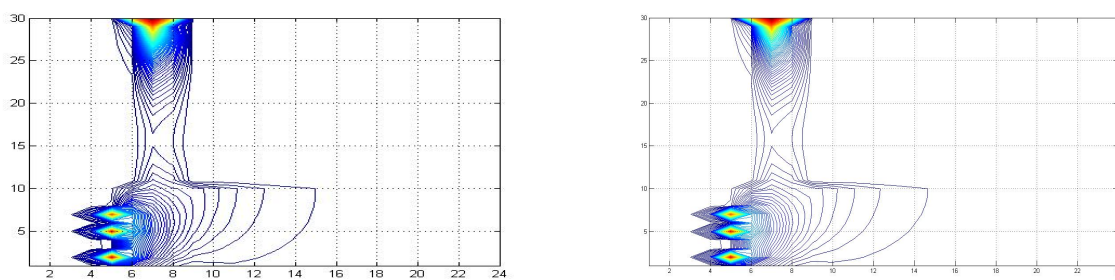


Figura 35 – Gráficos Referentes à dispersão de poluente no cenário dois em Ω_2 no Terceiro ano, nos tempos $t_1 = 181$, $t_2 = 92$.

5.1.3 Cenário Ω_1 e Ω_2 - Controle Total Urbano

Neste cenário zeramos o valor do IQA (IQA= 100) em todas as fontes de poluentes do córrego, representado um controle urbano total de emissão de poluentes no córrego. Em Ω_1 desligamos as fontes pontuais que representam a emissão do poluente feita pela cidade, e em Ω_2 fizemos a mesma redução nas fontes vindas do córrego. Observa-se nas figuras [Figura 36a](#) [Figura 43](#) que a poluição vinda da agricultura da região é muito prejudicial ao meio ambiente, visto que, neste cenário, ela é a única fonte contínua de poluição.

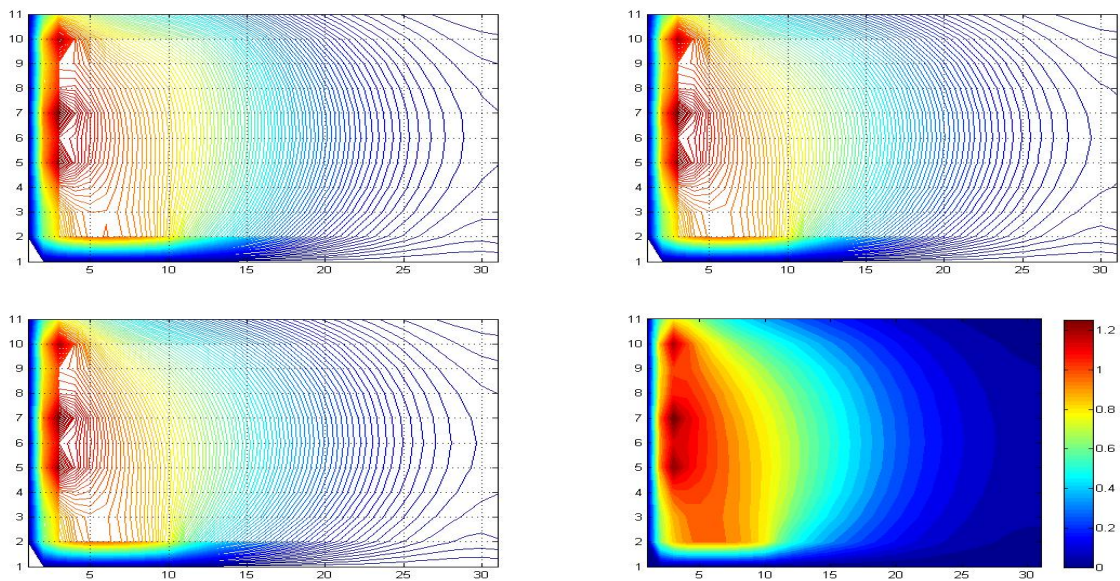


Figura 36 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 no primeiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

Observe-se também que o córrego consegue se limpar com sua autodepuração e, então, o córrego não influencia demasiadamente em termos da poluição do rio, apenas na sua advecção. Isso pode ser comprovado pelas figuras [Figura 40](#) a [Figura 43](#), em que se vê que a curva de poluição está mudando de tempos em tempos, e isso acontece pelo perfil de velocidade utilizado, e pela influencia do córrego ao desaguar no rio Jordão.

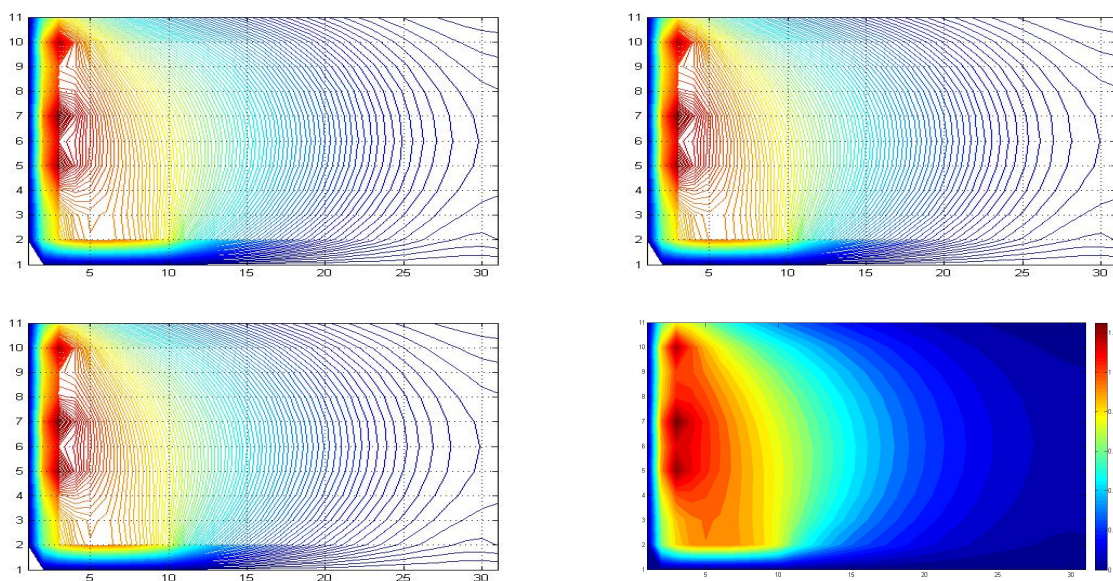


Figura 37 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

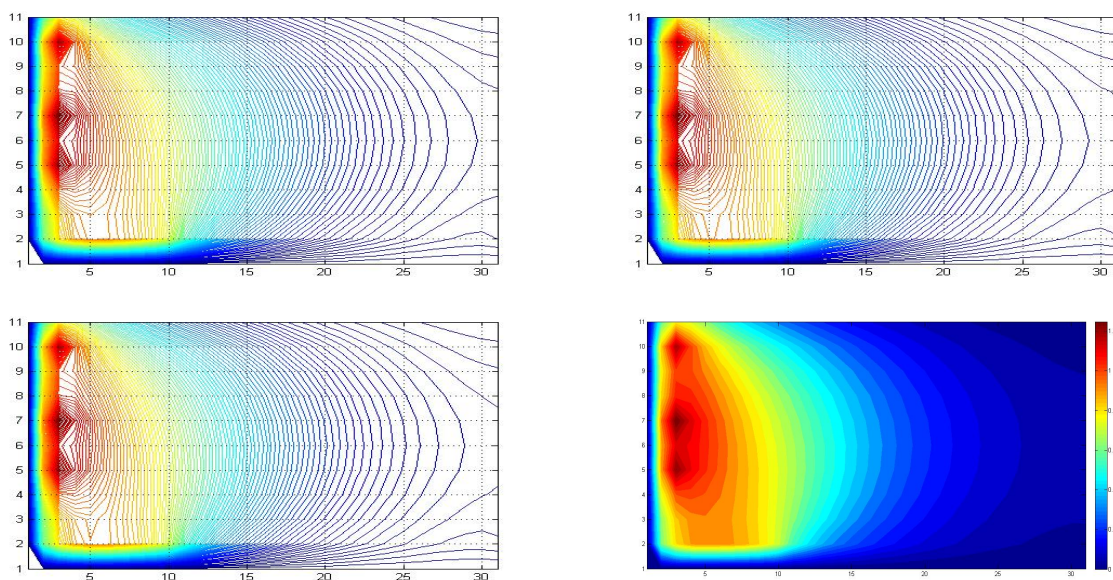


Figura 38 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 no terceiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

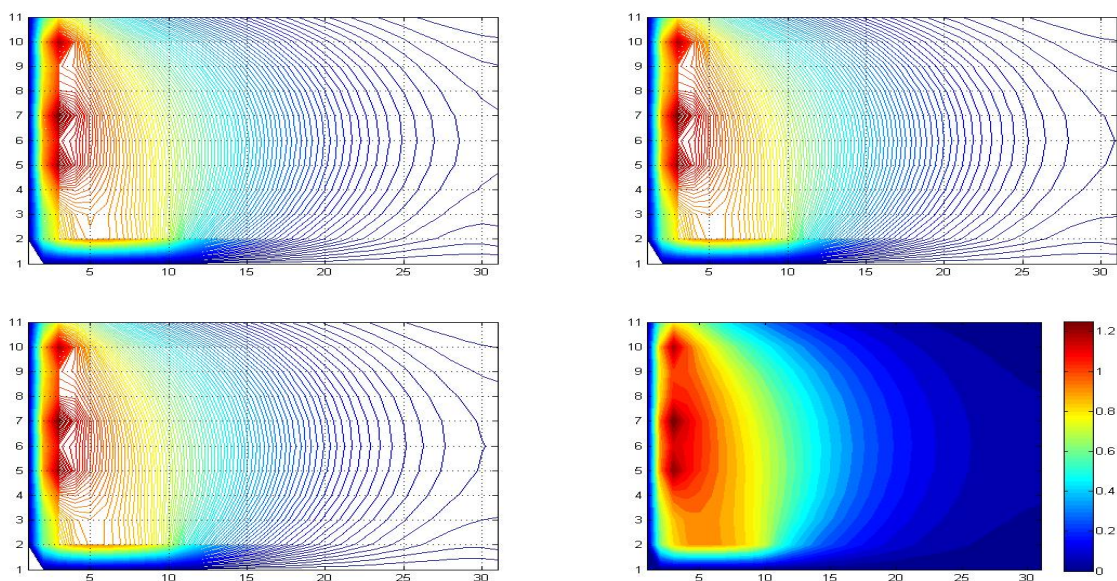


Figura 39 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_1 (1460 dias - 4 anos).

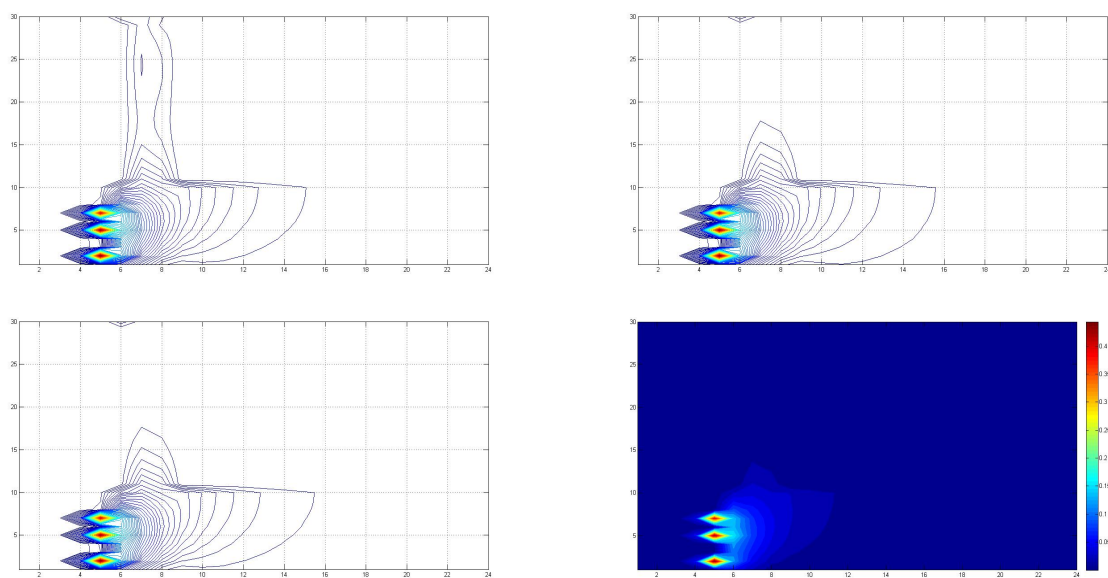


Figura 40 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no primeiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

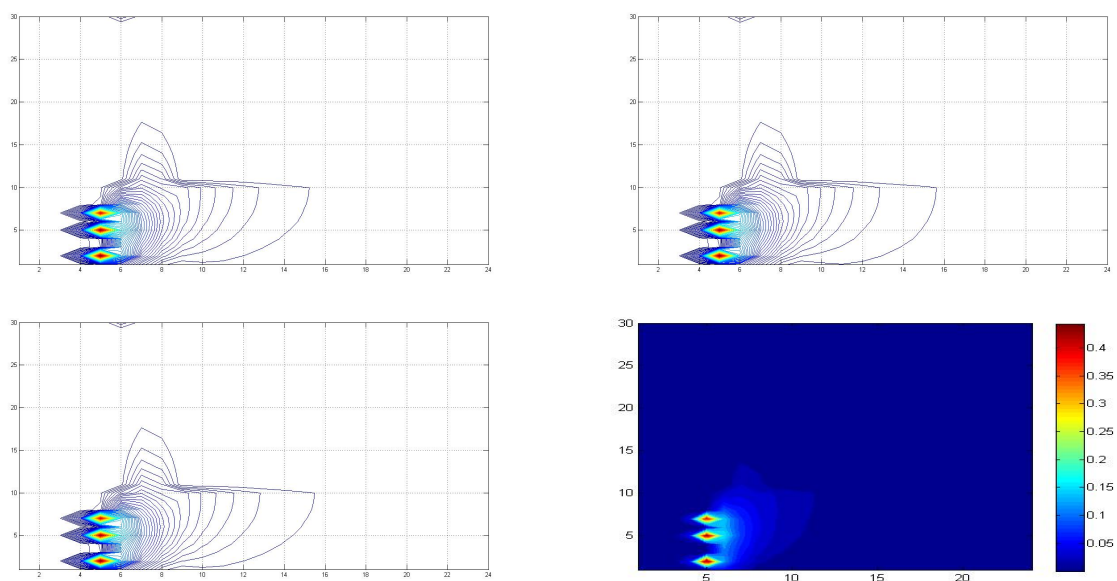


Figura 41 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no segundo ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

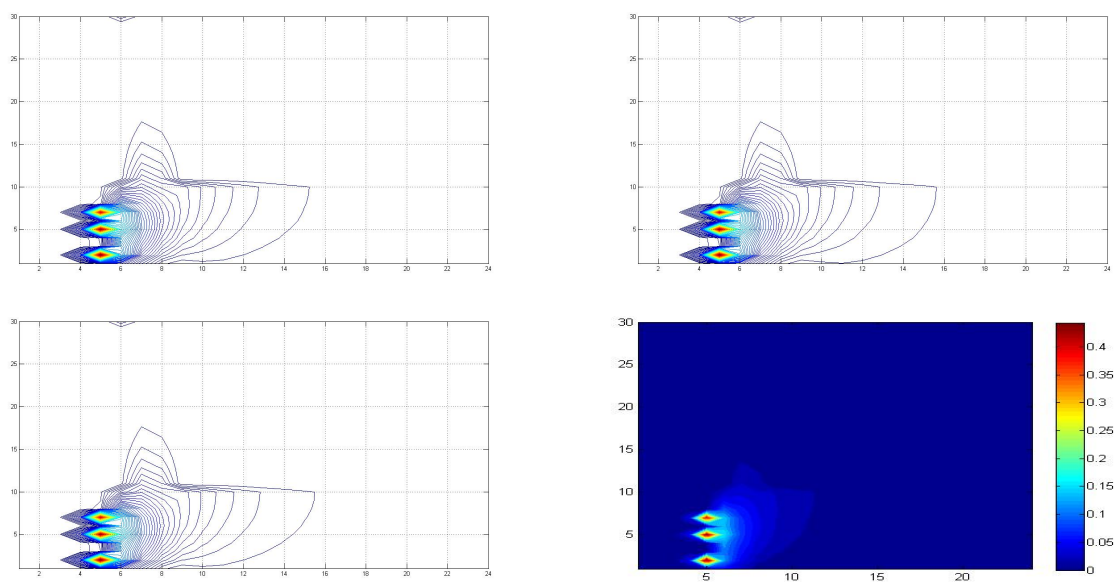


Figura 42 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no terceiro ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

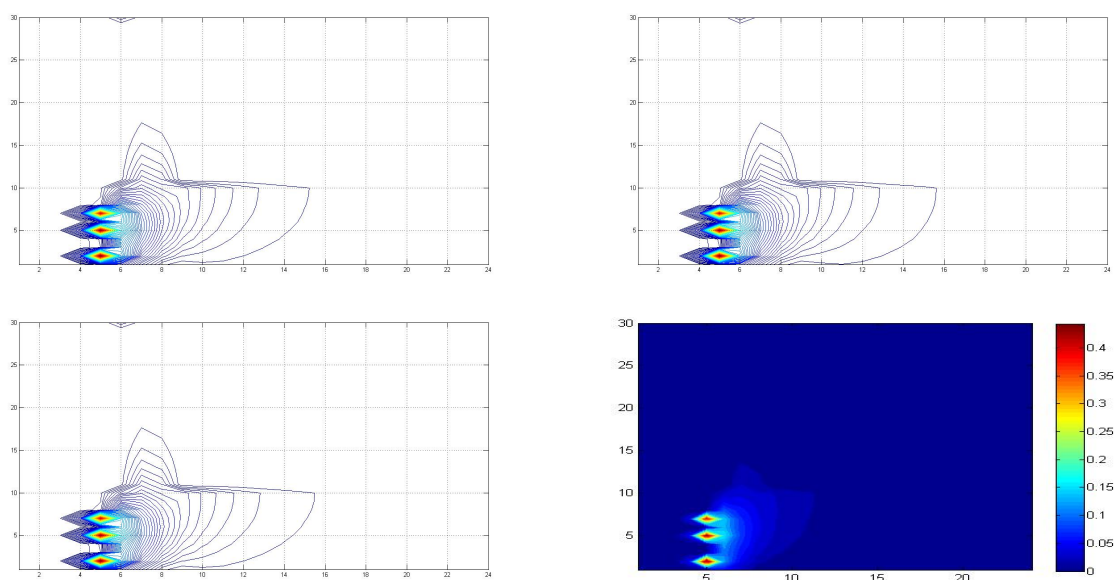


Figura 43 – Gráficos Referentes à dispersão de poluente no cenário três em Ω_2 no quarto ano, nos tempos $t_1 = 181$, $t_2 = t_3 = 92$.

6 Considerações Finais

Com o intuito de criar um instrumento auxiliar no estabelecimento de políticas públicas de preservação ambiental e seu planejamento. Com este fim em vista, foi produzido um instrumento algorítmico que permite simular situações de impacto no deságue de um rio (ou córrego) em outro maior.

Com poucas alterações, o programa implementado pode ser adaptado a muitas situações indicadas no parágrafo anterior.

Um aspecto inovador deste trabalho se manifesta na combinação dos perfis de velocidade na região do deságue, combinando 2 perfis de Poiseuille, para obter o perfil a jusante desse deságue, o que resulta em simulações mais fidedignas qualitativamente.

É realmente simples agregar ao modelo a presença de ventos predominantes e sua influência nas manchas de poluente de superfície e, finalmente, acreditamos que este trabalho se preste a ser um instrumento auxiliar no estabelecimento de políticas públicas de ações emergenciais de contenção, de preservação e de avaliação de impacto, bem como vemos nele um texto que pode ajudar na mudança de legislação em locais onde o algoritmo venha a ser usado do modo indicado.

Finalmente, após o término deste trabalho, verificamos que se trata, na realidade, de um ponto de partida para uma vasta gama de outras situações como a inclusão de um ou mais deságues, com radicais diferenças em seus estados de poluição bem como o uso de Equações como as de Stokes ou Navier-Stokes na simulação do perfil de velocidades e, ainda, na introdução, no modelo de técnicas de limpeza e contenção, efetuadas de modo discreto, ou seja, apenas periodicamente e não de maneira contínua.

Referências

- ABREU, L. C. *Influência de poluentes sobre macroalgas na Baía de Sepetiba, RJ : modelagem matemática, análise numérica e simulações computacionais*. Dissertação (Mestrado) — Universidade Estadual de Campinas - UNICAMP, 2009.
- ALVES, L. F. B. *Modelagens matemáticas para simulações computacionais de Impacto ambiental no Rio Balsas*. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2009.
- BASSANEZI, R. C. *Ensino-aprendizagem com modelagem matemática: ua nova estratégia*. [S.l.: s.n.], 2004.
- BATSCHELET, E. *Introdu temca para Biocientistas*. [S.l.: s.n.], 1978.
- BROOKS, A. N.; HUGHES, T. J. Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer methods in applied mechanics and engineering*, Elsevier, v. 32, n. 1, p. 199–259, 1982.
- BURDEN, R. L. *Análise Numérica*. [S.l.: s.n.], 2013.
- CANTAO, R. F. *Modelagem e simulação numérica de derrames de óleo no Canal de São Sebastião, SP*. Dissertação (Mestrado) — Universidade Estadual de Campinas, 1998.
- CANTRELL, R. S.; COSNER, C. *Spatial ecology via reaction-diffusion equations*. [S.l.: s.n.], 2004.
- COELHO, M. de F. C. D. *Qualidade das aguas superficiais de Minas Gerais em 2014: resumo executivo*. [S.l.]: Instituto Mineiro de Gestas Aguas, 2015.
- CUNHA, M. C. C. *Métodos Numéricos*. [S.l.: s.n.], 2000.
- DINIZ, G. L. *A Mudança no habitat de populações de peixes: de rio a represa - o modelo matemático*. Dissertação (Mestrado) — Universidade Estadual de Campinas, 1994.
- DINIZ, G. L. *Dispersão de poluentes num sistema ar-água: modelagem, aproximação e aplicaesções*. Tese (Doutorado) — Universidade Estadual de Campinas - UNICAMP, 2003.
- FORTUNA, A. de O. *Técnicas Computacionais Para Dinâmica dos Fluídos: Conceitos Básicos e Aplicações*. [S.l.: s.n.], 2000.
- GUACA, D. C. *Impacto ambiental em meios aquáticos : modelagem, aproximação e simulação de um estudo na Baía de Buenaventura-Colômbia*. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2015.
- INFORZATO, N. F. *Dispersão de poluentes num sistema ar-água: modelagem matemática, aproximação numérica e simulação computacional*. Tese (Doutorado) — Universidade Estadual de Campinas, 2008.

- KRINDGES, A. *Modelagem e simulação computacional de um problema tridimensional de difusão-advecção com uso de Navier - Stokes*. Tese (Doutorado) — Universidade Estadual de Campinas - UNICAMP, 2011.
- MISTRO, D. C. *O problema da poluição em rios por mercúrio metálico: Modelagem e simulação numérica*. Dissertação (Mestrado) — Universidade Estadual de Campinas - UNICAMP, 1992.
- NOVAIS, G. T. *Caracterização climática da mesorregião do triângulo mineiro/alto paranaíba e do entorno da serra da canastra (mg)*. 2011.
- OKUBO, S. A. L. A. *Diffusion and Ecological Problems: Modern Perspectives*. [S.l.: s.n.], 2001.
- OLIVEIRA, R. F. de. *O comportamento evolutivo de uma mancha de óleo na Baía de Ilha Grande, RJ : modelagem, análise numérica e simulações*. Tese (Doutorado) — Universidade Estadual de Campinas - UNICAMP, 2003.
- PAULA, L. M. de. *Avaliação da Qualidade da água e Autodepuração do Rio Jordão, Araguari (MG)*. Dissertação (Mestrado) — Universidade Federal de Uberlândia, 2011.
- POFFO, I. R. F. *Gerenciamento de riscos socioambientais no complexo portuário de Santos na ótica ecossistêmica*. Tese (Doutorado) — Universidade de São Paulo, 2007.
- PRESTES, M. F. B. *Dispersão de material impactante em meio aquático: modelo matemático, aproximação numérica e simulação Computacional - Lagoa do Taquaral, Campinas, SP*. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2011.
- SALLA, M. R. et al. Estudo da autodepuração do rio jordão, localizado na bacia hidrográfica do rio dourados. *Eng Sanit Ambient, SciELO Brasil*, v. 18, n. 2, p. 105–114, 2013.
- VASQUEZ, J. C. S. *Comportamento evolutivo de descarga de água de produção decorrente de atividade offshore: tratamento numérico e simulação computacional*. Tese (Doutorado) — Universidade Estadual de Campinas - UNICAMP, 2005.
- VIEIRA, J. R. *Modelação matemática da circulação e transporte hidrodinâmicos em regiões costeiras. aplicação ao rio de la plata*. Centro Internacional de Métodos Numéricos en Ingeniería, 1990.
- WOLMUTH, L. D. *Modelagem e simulações do Comportamento evolutivo de poluentes em corpos aquáticos de grande extensão: o caso da represa do rio Manso*. Dissertação (Mestrado) — Universidade Estadual de Campinas - UNICAMP, 2009.

Anexos

ANEXO A – Domínio Ω_1

```

%clear all
%clc

function [cm,ctes ,condInit]=paraboliconovo(cz ,u0 ,v ,aaa ,bbb ,...
                                             ccc ,ddd ,eee , fff , tfinal ,npt)

% -----
% O problema
% Estudar a polui do Rio Jordó (Minas Gerais)
% -----

% -----
% P a r metros do Problema
% -----

tic
alf=0.00375;
beta=-0.1;
gamma=-0.1;

mu=0.000001;
f=1.0;

% -----
% P a r metros do domino
% -----

l=10;
h=4;
%tfinal=182;

% -----
% P a r metros da discretiza
% -----

nx=30;
ny=10;

```

```

%npt=500;
dx=1/nx;
dy=h/ny;
dt=tfinal/npt;
ddx=dx*dx;
ddy=dy*dy;
nnx=nx;
nny=ny;
nn=nnx*nny;

% -----
% Campo de velocidades
% -----

y = dy:dy:h;
u=(4/9)*u0*y.*h - (4/9)*u0*(y.^2);
m = size(y);

% -----
% Ncleo de P clet
% -----

r = u*dx/alf;
r = norm(r, inf)
if r>=2
    disp('Pare, p clet alto:')
else

% -----
% Calculo dos Valores Auxiliares
% -----

adtx=(alf*dt)/ddx;
adty=(alf*dt)/ddy;
vty=(v*dt)/(2*dy);
mt=(mu*dt)/2;

dpe=1+adtx+adty+mt; %Diagonal principal esquerda
dpd=1-adtx-adty-mt; %Diagonal principal direita

```

```

% -----
% Montagem da Matriz sem incluir o contorno
% -----

ae=sparse(nn);
ad=sparse(nn);
f=zeros (nn,1);
tic;
for i=1:nn
    ae(i,i)=dpe;
    ad(i,i)=dpd;
end
for j = 1:m(2);
    for i=1:nn-1
        ae(i+1,i)=(-adty-vty)/2;
        ad(i+1,i)=(adty+vty)/2;
        ae(i,i+1)=(-adty+vty)/2;
        ad(i,i+1)=(adty-vty)/2;
    end
    for i=1:nn-nny
        ae(i+nny,i)=(-adtx-(u(j)*dt)/(2*dx))/2;
        ad(i+nny,i)=(adtx+(u(j)*dt)/(2*dx))/2;
        ae(i,i+nny)=(-adtx+(u(j)*dt)/(2*dx))/2;
        ad(i,i+nny)=(adtx-(u(j)*dt)/(2*dx))/2;
    end
end

% -----
% C o r r e e s   d a   M a t r i z
% C o r r e e s   d a   B o r d a   i n f e r i o r
% -----

g=0.0;
for j = 1:m(2);
    utx=(u(j)*dt)/(2*dx);
    for i=2:nnx
        ind=(i-1)*nny+1;
        ae(ind,ind-1)=0;
    end
end

```



```

        ad(ind , ind -1)=0;
    end
    for i=1:nnx
        ind=(i-1)*nny+1;
        f(ind)=2*g*(adty+vty)/2;
    end
end
end

% -----
% Condição de Robin superior (condição de Robin)
% -----

for j = 1:m(2);
    utx=(u(j)*dt)/(2*dx);
    for i=1:nny-1
        ind=i*nny;
        robin2=(alf*gamma*dt)/dy -(v*gamma*dt)/2;
        ae(ind , ind+1)=0;
        ad(ind , ind+1)=0;
        ae(ind , ind-1)=-adty;
        ad(ind , ind-1)=adty;
        ae(ind , ind)=dpe+robin2;
        ad(ind , ind)=dpd-robin2;
    end
    ae(nn , nn-1)=-adty;
    ad(nn , nn-1)=adty;
end

% -----
% Lateral lado direito (condição de Robin)
% -----

for j = 1:m(2);
    utx=(u(j)*dt)/(2*dx);
    for i=1:nny
        robin=(u(j)*beta*dt)/2-(alf*beta*dt)/dx;
        ind=(nnx-1)*nny+i;
        ae(ind , ind-nny)=-adtx;
        ad(ind , ind-nny)=adtx;
    end
end

```

```

        ae(ind , ind)=dpe+robin ;
        ad(ind , ind)=dpd-robin ;
    end
end

% -----
% Fontes
% -----
iqa1=aaa ;
iqa2=bbb ;
iqa3=ccc ;
    f(81)=1-(iqa1/100);
    f(51)=1-(iqa2/100);
    f(101)=1-(iqa3/100);
iqa4=ddd ;
iqa5=eee ;
iqa6=fff ;
    f(12)=1-(iqa4/100);
    f(15)=1-(iqa5/100);
    f(17)=1-(iqa6/100);

% -----
% Resolver o Sistema
% -----

    cz=zeros(nn,1);
    c=zeros(nn,npt);
    p=zeros(4,npt);
    p=zeros(4+nny,npt);
it=1;
erro=2;
    while it<npt && erro>0.00001
    %for j=1:npt
        c=ae\ (ad*cz+(f*dt));

        cz=c;

        lncz=log10(1+cz);

```

```

    cm(:, it)=cz;
    ctes(:, it)=lncz;

    vp1(it)=cm(101, it);
    vp2(it)=cm(37, it);
    vp3(it)=cm(51, it);
    vp4(it)=cm(76, it);
    if it==1
        erro=norm(cm(:, it)-f);
    else
        erro=norm(cm(:, it)-cm(:, it-1))
    end
    it=it+1;
end
save anacarol c
condInit=cm(:, it-1);
% -----
% Visualizacao dos resultados para c
% -----
verp=zeros(nny+1,nnx+1);

for j=1:nny+1
    verp(j,1)=0;
end
for i=2:nnx+1
    verp(1,i)=g;
end
size(verp)
for i=2:nnx+1
    for j=2:nny+1
        indc=(i-2)*nny + (j-1);
        verp(nny+2-(j-1),i)=ctes(indc, it-1);
    end
end
end
figure
contour(verp,100),grid
figure
surf(verp),grid
figure

```

```
contourf(verp,30), colorbar, shading flat;  
pause(0.05)  
end  
toc  
end
```

ANEXO B – Domínio Ω_2

```

%clear all
%clc
function [c,condInit]=corregoteste(cz,u0,v0,aaa,bbb,ccc,ddd,...
                                   eee,fff,tfinal,npt)

global pp

pp=31;

% -----
% O problema
% Estudar a polui do Rio Jordó (Minas Gerais)
% -----

% -----
% Parametros do Problema
% -----

alf=0.00375;

mu=0.000001;
f=1.0;
gamma=-0.01;
gamma2=0.008;
gamma3=-0.01;
gamma4=-0.01;
gamma5=-0.001;
gamma6=-0.025;

% -----
% Parametros do domino
% -----

l = 10;
h = 4;
%tfinal = 182;

```

```

% -----
% Parametros da discretiza
% -----
    nx1=5;
    nx2=3;
    nx3=16;
NNx=nx1+nx2+nx3;
    nx4=6;

nx=nx1+nx2+nx3+nx4;

ny1=10;
ny2=30;
ny3=ny2-ny1;
nn1=nx1*ny1;
ny=ny2;
nn2=(nx1+nx2)*ny1+(nx2*ny3);
nn3=(nx1+nx2+nx3)*ny1+(nx2*ny3);
%npt=182;
dx=1/nx;
L=(nx1+nx2+nx3)*dx;

dx1=L/(nx1+nx2+nx3);
nn=nx*ny1;
dy=h/ny;
dt=tfinal/npt;
ddx=dx1*dx1;
ddy=dy*dy;

P1=(nx1+1)*dx1;
P2=(nx1+nx2)*dx1;
a1=(P2-P1)/2;
c1=L-nx3*dx-a1;

% -----
% Campo de velocidades u
% -----

```

```

y = 0:dy:h;
u = 4*u0*((y/h)-((y.^2)/(h*h)));
m = size(y);

% -----
% Campo de velocidade v
% -----

w=dx1*(nx1+1):dx1:(nx1+nx2)*dx1;
v=v0*((w-c1).^2)/(a1^2-1);
m1 = size(w);
% -----
% Ncleo de P clet
% -----

r1 = u*dx/alf;
r1 = norm(r1, inf);
r2 = v*dy/alf;
r2 = norm(r2, inf);
if r1>=2
    disp('Pare, p clet alto:')
else
    if r2>=2
        disp('Pare, p clet alto:')
    else
        % -----
        % Calculo dos Valores Auxiliares
        % -----

        adtx=(alf*dt)/ddx;
        adty=(alf*dt)/ddy;
        mt=(mu*dt)/2;
        dpe=1+adtx+adty+mt; % Diagonal Principal Esquerda
        dpd=1-adtx-adty-mt; % Diagonal Principal Direita
        rob2=(-alf*gamma*dt)/dy;
        dspie=(-adty)/2;
        dspid=(adty)/2;
        dspse=(-adty)/2;
        dspsd=(adty)/2;

```

```

robin5=(-alf*gamma5*dt)/dx;
robin6=(-alf*gamma6*dt)/dx;
dsdie1=(-adtx)/2;
dsdid1=(adtx)/2;
dsdse1=(-adtx)/2;
dsdsd1=(adtx)/2;

% -----
% Montagem da Matriz de dimenso ny1*nx1 sem corre (1)
% -----
%
% |-----|
% |         | |
% |         | |
% |         | |
% |         | |
% |         | |
% -----|-----|-----|-----|-----|-----|-----|-----|
%|         | |         | | |
%|         | |         | | |
%|         | |         | | |
%|         | |         | | |
% -----|-----|-----|-----|-----|-----|-----|-----|

ae=sparse(nn);
ad=sparse(nn);
f=zeros(nn,1);
tic;

%   for gh=1:nn1
%       ae(gh,gh)=dpe;
%       ad(gh,gh)=dpd;
%   end
for gh=1:nn1
    ae(gh,gh)=dpe;
    ad(gh,gh)=dpd;
end
    for df=1:nn1-1
        ae(df+1,df)=(-alf*dt)/2*ddy;
        ad(df+1,df)=(alf*dt)/2*ddy;
        ae(df,df+1)=(-alf*dt)/2*ddy;
    end

```



```

        ad(df, df+1)=(alf*dt)/2*ddy;
    end
    for ja=1:m(2)
        for kl=1:nn1-ny1
            ae(kl+ny1, kl)=(-alf*dt)/(2*ddx)-((u(ja)*dt)/4*dx);
            ad(kl+ny1, kl)=(alf*dt)/(2*ddx)+((u(ja)*dt)/4*dx);
            ae(kl, kl+ny1)=(-alf*dt)/(2*ddx)+((u(ja)*dt)/4*dx);
            ad(kl, kl+ny1)=(alf*dt)/(2*ddx)-((u(ja)*dt)/4*dx);
        end
    end
end

% -----
% Montei a primeira matriz de nn1 pontos
% -----
% Corre da fronteira acima da primeira matriz de nn1 pontos

    for i=1:nx1-1
        ind=(1+i)*ny1;
        ae(ind, ind+1)=0;
        ad(ind, ind+1)=0;
        ae(ind, ind-1)=(-alf*dt)/(2*ddy);
        ad(ind, ind-1)=(alf*dt)/(2*ddy);
        ae(ind, ind)=dpe+rob2;
        ad(ind, ind)=dpd-rob2;
    end

% -----
% Corre de fronteira abaixo da primeira matriz de nn1 pontos
% -----
g=0.01;
    for i=2:nx1
        ind=(i-1)*ny1+1;
        ae(ind, ind-1)=0;
        ad(ind, ind-1)=0;
    end
    for i=1:nx1
        ind=(i-1)*ny1+1;
        f(ind)=2*g*((alf*dt)/(2*ddy));
    end
end

```

```

% -----
% Pontos da matriz de (nx1)*ny1+1 at nx1*ny1+ny1
% -----
for q3=1:m1(2)
    for jb=1:m(2)
        for i=(nx1)*ny1+1:(nx1)*ny1+ny1
            ae(i,i)=dpe;
            ad(i,i)=dpd;
            ae(i+1,i)=(-alf*dt)/(2*ddy)-(v(q3)*dt)/(4*dy);
            ad(i+1,i)=(alf*dt)/(2*ddy)+(v(q3)*dt)/(4*dy);
            ae(i,i+1)=(-alf*dt)/(2*ddy)+(v(q3)*dt)/(4*dy);
            ad(i,i+1)=(alf*dt)/(2*ddy)-(v(q3)*dt)/(4*dy);
            ae(i-ny1,i)=(-adtx-((u(jb)*dt)/(2*dx)))/2;
            ad(i-ny1,i)=(adtx+((u(jb)*dt)/(2*dx)))/2;
            ae(i,i-ny1)=(-adtx+((u(jb)*dt)/(2*dx)))/2;
            ad(i,i-ny1)=(adtx-((u(jb)*dt)/(2*dx)))/2;
            ae(i,i+ny2)=(-alf*dt)/(2*ddx)+(u(jb)*dt)/(4*dx);
            ad(i,i+ny2)=(alf*dt)/(2*ddx)-(u(jb)*dt)/(4*dx);
        end
    end
end

% -----
% Pontos da matriz de nx1*ny1+ny1+1 at nx1*ny1+ny2
% -----
for he=(nx1*ny1)+ny1+1:(nx1*ny1)+ny2-1
    ae(he+ny2,he)=0;
    ad(he+ny2,he)=0;
    ae(he,he+ny2)=-(alf*dt)/ddx;
    ad(he,he+ny2)=(alf*dt)/ddx;
    ae(he,he)=dpe+robin5;
    ad(he,he)=dpd-robin5;
    ae(he,he-1)=(-alf*dt)/2*ddy;
    ad(he,he-1)=(alf*dt)/2*ddy;
    ae(he-1,he)=(-alf*dt)/2*ddy;
    ad(he-1,he)=(alf*dt)/2*ddy;
end

% ponto de duas fronteiras.

```

```

i4=nx1*ny1+ny2;
ae(i4,i4)=dpe+((-alf*gamma2*dt)/dy)+robin5;
ad(i4,i4)=dpd+((alf*gamma2*dt)/dy)-robin5;
ae(i4,i4+1)=0;
ad(i4,i4+1)=0;
ae(i4,i4-1)=(-alf*dt)/ddy;
ad(i4,i4-1)=(alf*dt)/ddy;
ae(i4,ny2+i4)=(-alf*dt)/ddx;
ad(i4,i4+ny2)=(alf*dt)/ddx;
ae(i4,i4-ny2)=0;
ad(i4,i4-ny2)=0;

% - - - - -
% Montagem da matriz de nn2
% - - - - -

for i=nn1+ny2+1:nn2-ny2
    ae(i,i)=dpe;
    ad(i,i)=dpd;
end
for tb=1:m1(2)
    for l=2:nx2
        for k=ny1+1:ny2
            ind=(l-1)*ny3+(nx1*ny1)+(l-1)*ny1+k;
            ae(ind,ind-1)=(-alf*dt)/(2*ddy)-(v(tb)*dt)/(4*dy);
            ad(ind,ind-1)=(alf*dt)/(2*ddy)+(v(tb)*dt)/(4*dy);
            ae(ind-1,ind)=(-alf*dt)/(2*ddy)+(v(tb)*dt)/(4*dy);
            ad(ind-1,ind)=(alf*dt)/(2*ddy)-(v(tb)*dt)/(4*dy);
            ae(ind,ind-ny2)=(-alf*dt)/(2*ddx);
            ad(ind,ind-ny2)=(alf*dt)/(2*ddx);
            ae(ind-ny2,ind)=(-alf*dt)/(2*ddx);
            ad(ind-ny2,ind)=(alf*dt)/(2*ddx);
        end
    end
end
end
% Soma das velocidades
for q=1:m(2)
    for wd=1:m1(2)
        for i=2:nx2-1

```

```

for j=1:ny1
    ind=(i-1)*ny2+(nx1)*ny1+j;
    ae(ind,ind-1)=(-alf*dt)/(2*ddy) - ...
        (v(wd)*dt)/(4*dy);
    ad(ind,ind-1)=(alf*dt)/(2*ddy) + ...
        (v(wd)*dt)/(4*dy);
    ae(ind-1,ind)=(-alf*dt)/(2*ddy) + ...
        (v(wd)*dt)/(4*dy);
    ad(ind-1,ind)=(alf*dt)/(2*ddy) - ...
        (v(wd)*dt)/(4*dy);
    ae(ind,ind-ny2)=(-alf*dt)/(2*ddx) - ...
        (u(q)*dt)/(4*dx);
    ad(ind,ind-ny2)=(alf*dt)/(2*ddx) + ...
        (u(q)*dt)/(4*dx);
    ae(ind-ny2,ind)=(-alf*dt)/(2*ddx) + ...
        (u(q)*dt)/(4*dx);
    ad(ind-ny2,ind)=(alf*dt)/(2*ddx) - ...
        (u(q)*dt)/(4*dx);
    ae(ind,ind+ny2)=(-alf*dt)/(2*ddx) + ...
        (u(q)*dt)/(4*dx);
    ad(ind,ind+ny2)=(alf*dt)/(2*ddx) - ...
        (u(q)*dt)/(4*dx);
end
end
end
end

for p=1:m(2) % Corre fronteira abaixo. Soma as velocidades
    for p1=1:m1(2)
        for i=1:nx2
            ind=ny1*nx1+i*ny2+1;
            ae(ind,ind-1)=0;
            ad(ind,ind-1)=0;
            f(ind)=2*g*((alf*dt)/(2*ddy)+(v(p1)*dt)/(4*dy));
        end
    end
end

for r=2:nx2

```

```

ind=(1+r)*ny2+ny3;
ae(ind , ind+1)=0;
ad(ind , ind+1)=0;
ae(ind , ind-1)=(- alf*dt)/ddy;
ad(ind , ind-1)=( alf*dt)/ddy;
ae(ind , ind)=dpe+((- alf*gamma2*dt)/dy);
ad(ind , ind)=dpd+(( alf*gamma2*dt)/dy);

end
% - - - - -
% de 2*ny2+ny3+1 at 2*ny2+ny3+ny1 (soma das velocidades) na
% diagonal distante a esquerda e velocidade em y (u(x)) na
% diagonal distante a direita .
% - - - - -
for hy=1:m(2)
    for fv=1:m1(2)
        for et=nn2-ny2+2:nn2-ny2+ny1
            ae(et , et)=dpe;
            ad(et , et)=dpd;
            ae(et , et-ny2)=(- alf*dt)/(2*ddx)-(u(hy)*dt)/(4*dx);
            ad(et , et-ny2)=( alf*dt)/(2*ddx)+(u(hy)*dt)/(4*dx);
            ae(et , et+ny2)=(- alf*dt)/(2*ddx)+(u(hy)*dt)/(4*dx);
            ad(et , et+ny2)=( alf*dt)/(2*ddx)-(u(hy)*dt)/(4*dx);
            ae(et , et-1)=(- alf*dt)/(2*ddy)-(v(fv)*dt)/(4*dy);
            ad(et , et-1)=( alf*dt)/(2*ddy)+(v(fv)*dt)/(4*dy);
            ae(et-1 , et)=(- alf*dt)/(2*ddy)+(v(fv)*dt)/(4*dy);
            ad(et-1 , et)=( alf*dt)/(2*ddy)-(v(fv)*dt)/(4*dy);
        end
    end
end
for gh=1:m(2)
    for tas=1:m1(2)
        for i=nx2-1:nx2 % Corre fronteira abaixo .
            ind1=ny1*nx1+i*ny2+1;
            ae(ind1 , ind1-1)=0;
            ad(ind1 , ind1-1)=0;
            f(ind1)=2*g*(( alf*dt)/(2*ddy)+(v(tas)*dt)/(2*dy));
            ae(ind1 , ind1)=dpe;
            ad(ind1 , ind1)=dpd;
            ae(ind1 , ind1-ny2)=((- alf*dt)/(2*ddx) - ...

```

```

        (u(gh)*dt)/(2*dx));
    ad(ind1 , ind1-ny2)=(( alf*dt)/(2*ddx)+...
        (u(gh)*dt)/(2*dx));
    ae(ind1 , ind1+ny1)=((- alf*dt)/(2*ddx)+...
        (u(gh)*dt)/(2*dx));
    %velocidade apenas em y
    ad(ind1 , ind1+ny1)=(( alf*dt)/(2*ddx) -...
        (u(gh)*dt)/(2*dx));
    %velocidade apenas em y
    ae(ind1 -1, ind1)=0;
    ad(ind1 -1, ind1)=0;
    ae(ind1 , ind1+1)=(- alf*dt)/(2*ddx);
    ad(ind1 , ind1+1)=( alf*dt)/(2*ddx);
    end
end
end
for i=(nx1)*ny1+(nx2-1)*ny2+ny1+1:(nx1)*ny1+(nx2-1)*ny2+ny2-1
    ae(i-ny2 , i)=(- alf*dt)/ddx; % velocidade em x.
    ad(i-ny2 , i)=( alf*dt)/ddx;
    ae(i , i+ny2)=0;
    ad(i , i+ny2)=0;
    ae(i , i)=dpe+robin6;
    ad(i , i)=dpd-robin6;
    ae(ind , ind-1)=(- alf*dt)/2*ddy;
    ad(ind , ind-1)=( alf*dt)/2*ddy;
    ae(ind-1 , ind)=(- alf*dt)/2*ddy;
    ad(ind-1 , ind)=( alf*dt)/2*ddy;
end

% -----
% Ponto de duas fronteira do lado direito de cima do correjo
% -----

i5=nx1*ny1+nx2*ny2;
    ae(i5 , i5)=dpe+((- alf*gamma2*dt)/dy)+robin6;
    ad(i5 , i5)=dpd+(( alf*gamma2*dt)/dy)-robin6;
    ae(i5 , i5+1)=0;
    ad(i5 , i5+1)=0;
    ae(i5 , i5-1)=(- alf*dt)/ddy;

```

```

    ad(i5 , i5 -1)=( alf*dt )/ddy;
    ae(i5 , ny2+i5 )=0;
    ad(i5 , i5+ny2)=0;
    ae(i5 , i5 -ny2)=(- alf*dt )/ddx;
    ad(i5 , i5 -ny2)=( alf*dt )/ddx;

% -----
% primeira coluna da ltima matriz
% -----
    for i=nn2+1:nn3 % ltima matriz , velocidade em y (u(x))
        ae(i , i)=dpe;
        ad(i , i)=dpd;
    end
for hu=1:m(2)
    for dfv=1:m1(2)
        for pr=nn2+2:nn2+ny1-1 % 1 col. da ultima matriz s/ front
            ae(pr+1,pr)=(- alf*dt)/(2*ddy)-(v(dfv)*dt)/(4*dy);
            ad(pr+1,pr)=(+ alf*dt)/(2*ddy)+(v(dfv)*dt)/(4*dy);
            ae(pr , pr+1)=(- alf*dt)/(2*ddy)+(v(dfv)*dt)/(4*dy);
            ad(pr , pr+1)=( alf*dt)/(2*ddy)-(v(dfv)*dt)/(4*dy);
            ae(pr , pr -ny2)=(- alf*dt)/(2*ddx)-(u(hu)*dt)/(4*dx);
            ad(pr , pr -ny2)=( alf*dt)/(2*ddx)-(u(hu)*dt)/(4*dx);
            ae(pr , pr+ny1)=(- alf*dt)/(2*ddx)+(u(hu)*dt)/(4*dx);
            ad(pr , pr+ny1)=( alf*dt)/(2*ddx)-(u(hu)*dt)/(4*dx);
        end
    end
end
for ju=1:m(2)
    for mm=1:m1(2)
        nh=nn2+1; % Corre front abaixo da 1 col. ltima matriz
        ae(nh , nh-1)=0;
        ad(nh , nh-1)=0;
        f(nh)=2*g*(( alf*dt)/2*ddy+((u(ju)*dt))/4*dx);
        ae(nh , nh -ny2)=((- alf*dt)/(2*ddx)-(u(ju)*dt)/(4*dx));
        ad(nh , nh -ny2)=(( alf*dt)/(2*ddx)+(u(ju)*dt)/(4*dx));
        ae(nh , nh+ny1)=((- alf*dt)/(2*ddx)+(u(ju)*dt)/(4*dx));
        ad(nh , nh+ny1)=(( alf*dt)/(2*ddx)-(u(ju)*dt)/(4*dx));
        ae(nh-1,nh)=0;
        ad(nh-1,nh)=0;
    end
end

```

```

    end
end

for as=1:m(2)
    for asc=1:m1(2)
        itr=nn2+ny1; % Corre front acima, 1 col. ltima matriz
        ae(itr , itr+1)=0;
        ad(itr , itr+1)=0;
        ae(itr , itr-1)=-adty;
        ad(itr , itr-1)=adty;
        ae(itr , itr)=dpe+((-alf*gamma3*dt)/dy) +...
            (((u(as))*gamma3*dt)/2);
        ad(itr , itr)=dpd+((alf*gamma3*dt)/dy) - ...
            (((u(as))*gamma3*dt)/2);
        ae(itr , itr-ny2)=(-adtx-((u(as)*dt)/(2*dx))...
            + (v(asc)*dt)/(2*dy))/2;% velocidade em x e y
        ad(itr , itr-ny2)=(adtx+((u(as)*dt)/(2*dx))...
            + (v(asc)*dt)/(2*dy))/2;% velocidade em x e y
        ae(itr , itr+ny1)=(-adtx+((u(as)*dt)/(2*dx)))/2;% vel. y
        ad(itr , itr+ny1)=(adtx-((u(as)*dt)/(2*dx)))/2;% vel. y
    end
end

end

% -----
% Restante da ltima matriz
% -----

for i=nn2+ny1+1:nn3-1
    ae(i+1,i)=(-alf*dt)/(2*ddy);
    ad(i+1,i)=(alf*dt)/(2*ddy);
    ae(i , i+1)=(-alf*dt)/(2*ddy);
    ad(i , i+1)=(alf*dt)/(2*ddy);
end

for kg=1:m(2)
    for i=nn2+1:nn3-ny1
        ae(i+ny1 , i)=(-alf*dt)/(2*ddx)-(u(kg)*dt)/(4*dx);
        ad(i+ny1 , i)=(alf*dt)/(2*ddx)+(u(kg)*dt)/(4*dx);
        ae(i , i+ny1)=(-alf*dt)/(2*ddx)+(u(kg)*dt)/(4*dx);
        ad(i , i+ny1)=(alf*dt)/(2*ddx)-(u(kg)*dt)/(4*dx);
    end
end

```



```

        end
    end
    for kl=1:m(2)
        for i=(nn3-ny1)+2:(nn3-1)
            ae(i,i-ny1)=(-alf*dt)/(ddx);
            ad(i,i-ny1)=(alf*dt)/(ddx);
            ae(i,i)=dpe+(-alf*gamma4*dt)/dx + (u(kl)*gamma4*dt)/2;
            ad(i,i)=dpd-(-alf*gamma4*dt)/dx + (u(kl)*gamma4*dt)/2;
        end
    end
end
for ty=1:m(2)
    i2=nn-ny1+1;
    ae(i2,i2)=dpe+((-alf*gamma4*dt)/dx)+((u(ty)*gamma4*dt)/2);
    ad(i2,i2)=dpd+((alf*gamma4*dt)/dx) -((u(ty)*gamma4*dt)/2);
    ae(i2,i2-ny1)=(-alf*dt)/(ddx);
    ad(i2,i2-ny1)=(alf*dt)/(ddx);
    ae(i2,i2-1)=0;
    ad(i2,i2-1)=0;
    f(i2)=2*g*(alf*dt)/(2*ddx);
end

for sf=1:m(2)
    for i=2:nx3-1 %c o n d i   d e r o b i m   a c i m a   d a   l t i m a   m a t r i z
        ind=(i*ny1)+nn2;
        ae(ind,ind+1)=0;
        ad(ind,ind+1)=0;
        ae(ind,ind-1)=-adty;
        ad(ind,ind-1)=adty;
        ae(ind,ind)=dpe+((-alf*gamma3*dt)/dy) +...
            (((u(sf))*gamma3*dt)/2);
        ad(ind,ind)=dpd+((alf*gamma3*dt)/dy) -...
            (((u(sf))*gamma3*dt)/2);
    end
end
end

for q1=1:m(2)
    i3=nn;
    ae(i3,i3)=dpe+((-alf*gamma3*dt)/dy) +...
        (((u(q1))*gamma3*dt)/2)+((-alf*gamma4*dt)/dx) ...

```

```

        + ((u0*gamma4*dt)/2));
ad(i3 , i3)=dpd+(alf*gamma3*dt)/dy - ...
        ((u(q1))*gamma3*dt)/2+((alf*gamma4*dt)/dx) - ...
        ((u0*gamma4*dt)/2);
ae(i3 , i3-1)=-adty;
ad(i3 , i3-1)=adty;
ae(i3 , i3-ny1)=-adtx;
ad(i3 , i3-ny1)=adtx;
end
for df=1:m(2)
    for i=1:nx3-2 %dirichilet abaixo da ultima matriz
        ind=ny1*nx1+ny2*nx2+ny1+i*ny1+1;
        ae(ind , ind-1)=0;
        ad(ind , ind-1)=0;
        f(ind)=2*g*(alf*dt)/(2*ddy)+(u(df)*dt)/(4*dx);
    end
end
i1=ny1;
ae(i1 , i1)=dpe+rob2;
ad(i1 , i1)=dpd-rob2;
ae(i1 , i1+1)=0;
ad(i1 , i1+1)=0;
ae(i1 , i1-1)=-adty;
ad(i1 , i1-1)=adty;

i7=1;
f(1)=2*g*dspid;

% -----
% Fontes
% -----
%Fontes do c rrego ;
iqa1=aaa;
iqa2=bbb;
iqa3=ccc;

f(80)=1-(iqa1)/100;
f(110)=1-(iqa2)/100;

```

```

f(140)=1-(iqa3)/100;

%Fontes do rio;
iqa4=ddd;
iqa5=eee;
iqa6=fff;

f(42)=1-(iqa4)/100;
f(45)=1-(iqa5)/100;
f(47)=1-(iqa6)/100;
%f(85)=ddd;
%f(45)=ddd;

% - - - - -
% Resolver o Sistema
% - - - - -

%cz=zeros(nn,1);
c=zeros(nn,npt);
[malha]=MontaMalhaDif(24,20,10,3,5);
scrsz = get (0, 'ScreenSize ');
figure (1)
set(gcf, 'Position ', [1 scrsz(2) scrsz(3) scrsz(4)])
[Lm Um]=lu(ae);
zz=cz;
it=1;
erro=2;
while it<npt && erro>0.00001
% for it=1:npt
qq=ad*zz+f*dt;
yy=SLTI(Lm,qq);
zz=SLTS(Um,yy);
%z=Lm\ (ad*czl+ f*dt);
%clu=Um\z;
%cclu=cclu;
lnzz=log10(1+zz);
c(:,it)=zz;
ctes(:,it)=lnzz;
%c=ae\ (ad*cz+f*dt);

```

```
%cz=c;
ver=zeros (ny2 ,NNx);

for i=1:ny2
    for j=1:NNx
        if malha(i ,j)~=0
            ver(i ,j)=ctes (malha(i ,j) ,it );
        end
    end
end
ver=ver (ny2:-1:1 ,:);
contourf(ver ,30) , colorbar , shading flat;
%Img(it)=getframe;

pause(0.1);
if it==1
    erro=norm(c (: ,it)-f);
else
    erro=norm(c (: ,it)-c (: ,it -1))
end
it=it+1;

end
save isaac c
condInit=c (: ,it -1);

%createMovie('Simulacao' , 182 , Img)
figure
contourf(ver ,30) , colorbar , shading flat;
figure
contour(ver ,100) , grid;
toc
    end
end
end

end
```