



Unioeste - Universidade Estadual do Oeste do Paraná
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
Colegiado de Informática
Curso de Bacharelado em Informática

**Simulador Epidemiológico Baseado em Agentes Computacionais:
proposição, desenvolvimento e validação**

Danielly Veridiana Fontaniva Nunes

CASCADEL
2009

DANIELLY VERIDIANA FONTANIVA NUNES

**SIMULADOR EPIDEMIOLÓGICO BASEADO EM AGENTES
COMPUTACIONAIS: PROPOSIÇÃO, DESENVOLVIMENTO E
VALIDAÇÃO**

Monografia apresentada como requisito parcial
para obtenção do grau de Bacharel em Informática,
do Centro de Ciências Exatas e Tecnológicas da
Universidade Estadual do Oeste do Paraná - Cam-
pus de Cascavel

Orientadora: Profa. Dra. Claudia Brandelero Rizzi

CASCADEL
2009

DANIELLY VERIDIANA FONTANIVA NUNES

**SIMULADOR EPIDEMIOLÓGICO BASEADO EM AGENTES
COMPUTACIONAIS: PROPOSIÇÃO, DESENVOLVIMENTO E
VALIDAÇÃO**

Monografia apresentada como requisito parcial para obtenção do Título de Bacharel em Informática, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos professores:

Prof. Dr. Claudia Brandelero Rizzi (Orientadora)
Colegiado de Informática, UNIOESTE

Prof. MSc. Guilherme Galante
Colegiado de Informática, UNIOESTE

Prof. Dr. Rogério Luís Rizzi
Colegiado de Matemática, UNIOESTE

Cascavel, 15 de dezembro de 2009

“It is no measure of health to be well adjusted to a profoundly sick society.”

Krishnamurti

“As long as men massacre animals, they will kill each other. Indeed, he who sows the seeds of murder and pain cannot reap the joy of love.”

Pythagoras

AGRADECIMENTOS

Gostaria de expressar meus agradecimentos a algumas pessoas, dentre muitas que contribuíram para que esta obra viesse a existir.

Em primeiro lugar à minha mãe, Inês, que me deu condições para estudar. É graças a ela que pude vencer mais esta etapa da minha vida. Obrigada por se sacrificar tantas vezes, para que fosse possível eu chegar a este trabalho de conclusão de curso.

Ao meu namorado Adriano, pelos bons momentos que passamos junto. Agradeço também pela compreensão e apoio dado para a realização deste trabalho.

Agradeço aos meus irmãos, Diego e Douglas, e ao meu pai, Sadi, que sempre me incentivaram nos estudos.

A todos os meus colegas de graduação, mas em especial preciso agradecer a três deles: as meninas, Eliana Paula Teixeira e Renata Rockenbach, pelo apoio incondicional, e presença constante, nos momentos fáceis e difíceis; e ao Osmar dos Santos, por toda a sua ajuda e disponibilidade.

Agradeço à professora Claudia, minha orientadora, que me conduziu durante o percurso deste trabalho, e à ajuda dos professores componentes da banca, Guilherme Galante e Rogério Rizzi.

E, por fim, agradeço à UNIOESTE, que apesar dos problemas, me proporcionou inúmeras alegrias durante estes cinco anos.

Lista de Figuras

2.1	Modelos compartimentais dos tipos SIS e SI	8
2.2	Modelo compartimental do tipo SIR	8
2.3	Modelo compartimental do tipo SIRS	9
2.4	Modelo compartimental do tipo SIRC	10
2.5	Evolução do número de substitutos em uma epidemia	14
3.1	Transmissão em Autômatos Celulares	18
3.2	Tipos de redes complexas	21
5.1	Ilustração dos conceitos usados na modelagem do sistema multiagentes da ferramenta de simulação epidemiológica	31
5.2	Vizinhança de Moore representada pelas células em azul	33
5.3	Máquina de estados utilizada na modelagem do agente-pessoa	34
5.4	Ação referente à mudança de posição corrente	36
5.5	Ambiente dividido em comunidades	37
6.1	Conteúdo do documento de texto da simulação de uma epidemia SEIR. A primeira coluna refere-se ao dia do sistema, a segunda coluna refere-se ao número de indivíduos suscetíveis em cada dia do sistema. Na terceira coluna pode-se ver o número de indivíduos latentes, na quarta coluna o número de indivíduos infectados e na quinta coluna o número de indivíduos no compartimento dos imunes.	41
6.2	Gráfico gerado pela API Google Chart referente à simulação de uma epidemia SEIR	42
6.3	Diagrama de pacotes mais geral do aplicativo desenvolvido	43

6.4	Fluxograma da função principal da classe Pessoa	46
6.5	Fluxograma da função <i>mover</i> da classe Pessoa	46
6.6	Fluxograma da função <i>verificaVizinhanca</i> da classe Pessoa	47
6.7	Fluxograma da função <i>infectar</i> da classe Pessoa	48
6.8	Fluxograma da função <i>alterarEstadoInterno</i> da classe Pessoa	49
6.9	Tela principal da ferramenta desenvolvida	50
6.10	Ambientes desenvolvidos. a) nove comunidades altamente conectadas; b) nove comunidade com conectividade média; c) nove comunidade com baixa conectividade; d) nove comunidades sem conectividade; e) cinco comunidades com diferentes conexões entre si; f) ambiente representando os bairros do município de Cascavel.	53
7.1	Tela inicial do experimento desenvolvido	56
7.2	Tela da ferramenta parcial utilizando o ambiente dividido em cinco comunidades	58
7.3	Resultado da simulação de uma doença do modelo compartimental SEIR usando agentes computacionais inseridos em um ambiente dividido em cinco comunidades	59
7.4	Resultados dos experimentos de White. Na linha vertical, o números de células da malha dos autômatos, onde cada célula representa 100 indivíduos. Na linha horizontal as iterações do sistema.	60
7.5	Gráfico gerado pela simulação utilizando os parâmetros de White	61
7.6	Número acumulado de indivíduos infectados segundo o experimento de Huang.	62
7.7	Número acumulado de indivíduos infectados obtidos durante simulação de 90 dias com a ferramenta desenvolvida	62
7.8	Número de indivíduos infectados na simulação de 50 dias com taxa de transmissão igual a 100%	63
7.9	Em amarelo estão as conexões entre bairros do município de Cascavel	64
7.10	Espalhamento da epidemia inserida no centro da cidade	65
7.11	Espalhamento da epidemia inserida no bairro Floresta	66
7.12	Gráfico da simulação onde o indivíduo infectado é inserido no centro da cidade	66
7.13	Gráfico da simulação onde o indivíduo infectado é inserido no bairro Floresta .	67

Lista de Tabelas

5.1	Conceitos e analogias utilizados na modelagem do sistema multiagentes da ferramenta desenvolvida	31
6.1	Atributos da classe Pessoa	44
6.2	Opções de configuração da ferramenta de simulação epidemiológica	51
6.3	Atributos da classe <i>Ambiente</i>	53

Lista de Abreviaturas e Siglas

AC	Autômatos Celulares
API	<i>Application Programming Interface</i>
C3N	Congresso da Academia Trinacional de Ciências
ECA	Encontro em Computação Aplicada
FIPA	<i>Foundation for Intelligent Physical Agent</i>
GCAE	Grupo de Computação Aplicada à Epidemiologia
IDE	<i>Integrated Development Environment</i>
JVM	<i>Java Virtual Machine</i>
OMG	<i>Object Management Group</i>
OO	Orientado a Objetos
PNG	<i>Portable Network Graphics</i>
SDP	Solução Distribuída de Problemas
SI	Modelo Compartimental Suscetível-Infetado
SIR	Modelo Compartimental Suscetível-Infetado-Recuperado
SIRS	Modelo Compartimental Suscetível-Infetado-Recuperado-Suscetível
SIS	Modelo Compartimental Suscetível-Infetado-Suscetível
SMA	Sistema Multiagente
UML	<i>Unified Modeling Language</i>
UNIOESTE	Universidade Estadual do Oeste do Paraná
URL	<i>Universal Resource Identifier</i>
WYSIWYG	<i>What You See Is What You Get</i>

Lista de Símbolos

β	Número médio de contatos de um infeccioso em uma unidade de tempo
λ	Força da infecção
c	Taxa de recuperação da doença
d	Taxa de mortalidade
pmi	Período médio de infecção
R_0	Capacidade máxima da doença para aumentar a prole de infectados
R_0s	Número de substitutos

Sumário

Lista de Figuras	vi
Lista de Tabelas	viii
Lista de Abreviaturas e Siglas	ix
Lista de Símbolos	x
Sumário	xi
Resumo	xiv
1 Introdução	1
1.1 Justificativas	2
1.2 Objetivos	2
1.3 Motivação	3
1.4 Organização do Trabalho	3
2 Epidemiologia Matemática	5
2.1 Epidemia e Epidemiologia	6
2.2 Modelos Compartimentais	6
2.2.1 Modelos Elementares	7
2.3 O processo de transmissão	10
2.3.1 Contato	10
2.3.2 Incidência	11
2.3.3 Força de infecção	11
2.3.4 Período médio de infecção	12
2.3.5 Medida da capacidade máxima da doença	12
2.3.6 Número de substitutos	13
2.3.7 Anatomia de uma epidemia	13

2.4	Considerações finais	14
3	Abordagens Matemáticas e Computacionais na Modelagem de Epidemias	15
3.1	Baseado em equações	15
3.2	Baseado em autômatos celulares	17
3.3	Método de Monte Carlo	18
3.4	Baseado em Lógica <i>Fuzzy</i>	19
3.5	Redes complexas	20
3.5.1	Redes de mundo pequeno	21
3.5.2	Redes de escala livre	22
3.5.3	Redes randômicas	22
3.6	Considerações finais	22
4	Agentes Computacionais	23
4.1	Definições para Agentes Computacionais	23
4.2	Propriedades dos agentes	24
4.3	Classificação geral dos agentes	25
4.3.1	Agentes cognitivos	25
4.3.2	Agentes reativos	25
4.4	Arquitetura	26
4.5	Ambiente	27
4.6	Considerações finais	28
5	Modelagem da Ferramenta	30
5.1	Modelagem do Sistema Multiagente	30
5.2	Modelagem dos Agentes	32
5.2.1	Pessoa	32
5.2.2	Barreira	35
5.3	Modelagem do Ambiente	35
5.4	Considerações finais	35
6	A Implementação do Simulador	39
6.1	O pacote componentes	42
6.1.1	Classe Barreira	43

6.1.2	Classe Pessoa	44
6.2	O pacote <i>view</i>	49
6.2.1	A classe <i>Ambiente</i>	52
6.2.2	A classe <i>Simulador</i>	54
6.3	Considerações finais	54
7	Resultados Obtidos	55
7.1	Resultados Parciais - ECA	56
7.2	Resultados Parciais - C3N	57
7.3	Resultados Finais	58
7.3.1	Análises comparativas	59
7.3.2	Análise da influência dos parâmetros de configuração	61
7.3.3	Análise do comportamento da ferramenta desenvolvida em simulações realizadas em ambientes divididos em comunidades	63
7.4	Considerações Finais	67
8	Conclusões	68
	Referências Bibliográficas	70

Resumo

Os mecanismos de transmissão de doenças infecciosas são bem conhecidos e tratados na literatura. Recentemente, pesquisas científicas tem sido produzidas no sentido de simular epidemias. Nestes termos, este trabalho procura identificar e utilizar técnicas de Agentes Computacionais para demonstrar que o comportamento de uma epidemia em uma sociedade de pessoas pode ser bem traduzidas em algoritmos computacionais. Como resultado prático, é apresentado um aplicativo plenamente funcional, que é capaz de simular eventos epidemiológicos em uma sociedade de agentes que mimetizam seres humanos. A avaliação deste estudo foi feita por meio da realização de experimentos, comparando-se os resultados com os de outros trabalhos a cerca de epidemiologia computacional encontrados na literatura, que utilizam diversas técnicas de simulação.

Palavras-chave: Agentes computacionais, Epidemiologia, Simulação Epidemiológica, Modelos Compartimentais.

Capítulo 1

Introdução

O Grupo de Computação Aplicada da Universidade Estadual do Oeste do Paraná (UNIOESTE) está desenvolvendo estudos e pesquisas, cujo intento é “*desenvolver e aplicar requisitos, métodos, técnicas, algoritmos e produtos de software (...) para construir um Sistema Computacional Web Integrado sobre Fenômenos Epidemiológicos, (...) para fins de estudos epidemiológicos e ações de vigilância epidemiológica*” [45], sistema este nomeado EPIDEMIO.

Como consequência das dificuldades do desenvolvimento de sistemas tão complexos quanto este, uma abordagem de interesse é a criação de um sistema computacional formado por muitas unidades mais simples, porém interligadas entre si, de modo que um influencie no comportamento do outro [15].

Desta maneira, o principal objetivo do projeto EPIDEMIO é o desenvolvimento deste sistema computacional, que deve ser executado em arquiteturas de memória compartilhada e distribuída. Para isso, o desenvolvimento das funcionalidades previstas serão projetadas e implementadas através de módulos. Dentre estes módulos, um é de singular importância para o presente trabalho, o módulo 5, designado *Métodos e Estratégias de Solução*. “*Os integrantes deste módulo tem por objetivo especificar, projetar e desenvolver as abordagens computacionais a serem adotadas para a solução dos modelos epidemiológicos, especialmente quanto à simulação da disseminação de determinadas doenças*” [45].

Este trabalho envolveu atividades de **revisão bibliográfica**, **definições teórico-metodológicas**, a **implementação da ferramenta de software** que gera simulações do comportamento de diversos modelos compartimentais epidemiológicos, representando uma das partes do sistema EPIDEMIO.

1.1 Justificativas

Vários dos modelos propostos para o estudo da dinâmica de epidemias são determinísticos, e assumem explicitamente que o contato entre os indivíduos dos diversos grupos é uniforme, sem considerar que estes contatos são feitos entre indivíduos que tem algum fator em comum, levando assim a resultados que não são suficientemente condizentes com o real [25][23]. Especificamente, estes modelos falham em simular em geral [56]:

- os sistemas de contatos individuais;
- os efeitos do comportamento individual;
- os aspectos espaciais da disseminação da epidemia;
- e os efeitos da interação entre diferentes tipos de indivíduos.

Apesar de o uso de agentes computacionais não ser recente, sua utilização na modelagem de sistemas complexos e simulações data de pouco tempo. Nestes casos, a modelagem de um ambiente como um todo é feita definindo os níveis mais detalhados das relações entre os agentes com propriedades específicas [50]. Portanto, com o uso de agentes computacionais busca-se compreender as estruturas complexas das interações sociais. O uso de agentes na simulação de epidemias, contribui para a compreensão do impacto dos comportamentos dos indivíduos nos seus grupos.

1.2 Objetivos

O objetivo principal deste trabalho é a implementação de uma ferramenta para simulação epidemiológica empregando a abordagem de agentes computacionais de modo a dar atenção aos efeitos e aspectos mencionados não contemplados pelos modelos compartimentais determinísticos. Para o alcance desse objetivo, sete outros, mais específicos, deverão ser alcançados:

1. Realizar um estudo teórico a respeito de epidemiologia e modelos comportamentais de epidemias;
2. Pesquisar o estado da arte em epidemiologia computacional, com os métodos, técnicas e abordagens mais utilizadas para simulações epidemiológicas.

3. Compreender a teoria de sistemas multiagentes, de modo a utilizá-los como norteadores da concepção das modelagens apresentadas no trabalho;
4. Estudar algoritmos de agentes computacionais apropriados ou que possam ser adaptados para a simulação de eventos epidemiológicos;
5. Modelar um sistema multiagentes completo que mimetize o comportamento de uma doença quando inserida em uma sociedade de seres humanos;
6. Implementar os modelos propostos, de modo a ilustrar como a observação do comportamento de uma epidemia inserida em uma comunidade pode ser traduzido em algoritmos de agentes computacionais funcionais;
7. Realizar testes para validar o funcionamento e aplicabilidade do sistema proposto.

1.3 Motivação

Este trabalho é motivado pelas seguintes razões:

- O uso de agentes computacionais em simulações é relativamente novo, interessante e apresenta diversos aspectos teóricos e práticos ainda por serem pesquisados;
- A área de agentes computacionais oferece várias oportunidades de aplicações;
- O projeto EPIDEMIO está de acordo com o documento "Grandes Desafios da Computação no Brasil: 2006 - 2016", e é motivador contribuir com este trabalho;
- Ao serem aplicados, os resultados do sistema EPIDEMIO impactarão significativamente na vigilância epidemiológica, motivando ações relativas à saúde coletiva e melhorando o sistema de saúde social.

1.4 Organização do Trabalho

A especificação de um modelo teórico requer a identificação das principais variáveis responsáveis, bem como a determinação das hipóteses que fundamentam a dinâmica do evento. Tais suposições devem incluir leis físicas e biológicas e as apropriadas formulações matemáticas que

são aplicáveis à modelagem. A partir desses procedimentos são deduzidas previsões e realizadas simulações, que são testadas e comparadas com dados experimentais. Se o resultado é suficientemente bom para os propósitos determinados, a modelagem é aceita. Caso contrário, ela deve ser modificada e esse ciclo de análise se repete. Sendo assim, este trabalho está dividido como segue:

O Capítulo 2, relacionado ao primeiro objetivo apresenta os principais conceitos a respeito de Epidemiologia Matemática. Logo em seguida, o Capítulo 3, determinado pelo segundo objetivo específico, discorre sobre as soluções computacionais em torno de problemas epidemiológicos sendo atualmente usadas. O Capítulo 4, por sua vez, com o intuito de satisfazer ao terceiro e ao quarto objetivos, apresenta a técnica e as peculiaridades dos algoritmos de Agentes Computacionais. Estes três capítulos tratam basicamente de revisão bibliográfica, e aqui servem de base teórica para a implementação da ferramenta proposta.

Na sequência, o Capítulo 5 apresenta uma modelagem completa de um sistema multiagentes cujos agentes representam pessoas em uma sociedade, sendo logo depois concretizado pelo Capítulo 6, que explica uma implementação prática da modelagem proposta. No Capítulo 7, relacionado ao último objetivo específico, testes são realizados com a ferramenta gerada, comparando-se seus resultados com as respostas de outros trabalhos relacionados encontrados na literatura. E finalmente, no Capítulo 8 são discutidas as conclusões do trabalho, algumas contribuições e propostas para trabalhos futuros.

De posse de tal organização, pode-se destacar como resultado prático desta produção **um programa plenamente funcional, que é capaz de simular eventos epidemiológicos em uma sociedade de agentes que mimetizam seres humanos.**

Capítulo 2

Epidemiologia Matemática

Estudos na área de saúde pública vem se tornando cada vez mais frequentes, particularmente em epidemiologia, visando entender melhor a dinâmica epidemiológica, melhorar o controle de epidemias e encontrar formas de evitar novos surtos, diminuindo assim a taxa de mortalidade e melhorando o sistema de saúde pública.

Atualmente, existem esforços interdisciplinares voltados para o estudo da propagação de epidemias. Neste contexto, a epidemiologia matemática se concentra em estudar os modelos matemáticos do tipo compartimental, determinísticos ou estocásticos, de disseminação de infecções. A epidemiologia computacional por sua vez, vem atendendo a demanda por ferramentas computacionais específicas, que sejam capazes de implementar eficiente e precisamente tais modelos, objetivando contribuir para que profissionais e gestores da saúde pública possam responder adequadamente à demanda, direcionando esforços e recursos caso ocorra um surto epidêmico.

O desenvolvimento e a aplicação dessas ferramentas computacionais são de grande interesse seja do ponto de vista social, de saúde pública ou econômico, considerando as implicações e consequências de uma epidemia.

Sob este escopo, os integrantes do projeto EPIDEMIO [45] vem trabalhando na temática epidemiologia sob diferentes abordagens [2][4][22][23][43].

Este capítulo traz uma abordagem geral sobre epidemiologia matemática e os seus modelos, bem como, provê uma breve descrição dos meios mais utilizados para estudos na área epidemiológica e para estudos envolvendo simulações.

2.1 Epidemia e Epidemiologia

“Epidemia é o desenvolvimento de uma moléstia infecciosa que afeta, durante um período determinado e em certo território, um grande número de pessoas” [37].

Assim pode-se dizer que a **epidemia** se caracteriza por uma doença geralmente infecciosa que ataca simultaneamente grande número de indivíduos em uma determinada localidade, podendo ser também surto periódico de uma doença infecciosa em dada população ou região.

Alguns exemplos de epidemias são a Peste Negra, que matou quase metade da população da Europa em 1348 e dizimou parte da China e da Índia [13], a Gripe Espanhola causou cerca de 50 milhões de mortes, sendo 8 milhões somente na Espanha, em 1918 [29]. Mais recentemente a Gripe Aviária em 2004, levou à morte cerca de 2 centenas de pessoas, principalmente na Ásia [21][49], e atualmente, a Gripe Suína que em menos de seis meses, apresentou dezenas de milhares de casos e milhares de mortes¹ [55].

Por sua vez a **epidemiologia** é o ramo da ciência que estuda o processo saúde-doença em populações específicas, analisando a disposição e os fatores determinantes das enfermidades e dos agravos à saúde coletiva, propondo medidas específicas de prevenção, de controle ou de erradicação [53].

2.2 Modelos Compartimentais

Uma formulação de modelos biológicos ou físicos pode resultar numa equação diferencial, ou em um sistema de equações diferenciais, que acompanhados de dados e parâmetros, são designados de modelos matemáticos baseados em equações. A formulação de tais tipos de modelos é amplamente utilizada em Epidemiologia Matemática, e sua modelagem tradicional é baseada em estruturas compartimentais, visto que esta abordagem representa uma relação de causa e efeito para determinadas doenças.

Uma estrutura compartimental é baseada na divisão da população de hospedeiros em categorias, entre as quais os hospedeiros fluem com taxas que, em geral, dependem das características próprias da doença e dos vetores de transmissão. Na dinâmica da transmissão os indivíduos que compõem a população hospedeira são divididos em classes sendo que as mais comuns são

¹Até a segunda semana de novembro de 2009 foram notificadas mais de 6.250 mortes no mundo em consequência da pandemia da influenza A H1N1.

rotuladas como M, S, E, I, R e C, onde:

- M refere-se a indivíduos com imunidade passiva. Este tipo de imunidade acontece quando alguns anticorpos são passados da mãe para o filho através da placenta. Desta forma, o recém-nascido possui imunidade temporária a algum tipo de infecção [22];
- S são indivíduos suscetíveis. Esta classe inclui as pessoas saudáveis, suscetíveis a contrair a doença quando em contato com indivíduos infectados, ou seja, todas as pessoas que podem contrair a infecção [43];
- O rótulo E diz respeito aos indivíduos que estão portando a doença mas não são transmissoras. Diz-se que estes estão no período de latência [22][43];
- Após o período de latência, o indivíduo entra na classe dos infectados, a classe I. Nesta classe, os indivíduos, tendo sintomas ou não, são capazes de transmitir a doença [22][43];
- Na classe R, a classe dos removidos, se encontram as pessoas que são isoladas, mortas, ou que se recuperaram da infecção e passaram a ser imunes [43];
- O compartimento C, que pode ser chamado de imunidade cruzada, projeta um estado intermediário entre o estado totalmente suscetível (S) e o estado totalmente protegido (R), onde o indivíduo que pertence à esse grupo possui imunidade parcial [10].

2.2.1 Modelos Elementares

As doenças infecciosas causadas por microparasitas podem ser separadas em dois grupos [30] [22]:

- Grupo das doenças que conferem imunidade ao infectado uma vez recuperado. A maioria das doenças pertencentes a este grupo são de origem viral, como por exemplo o sarampo e a varicela.
- Grupo das doenças em que o indivíduo uma vez recuperado volta a ser suscetível. Normalmente causadas por agentes bacterianos, como no caso da peste, ou por protozoários, como é o caso da malária.

Levando em conta os diversos estados relacionados com o processo infeccioso, os modelos epidemiológicos se dividem em três grandes grupos:

- **Modelo Suscetível-Infetado-Suscetível (SIS)**, utilizado em casos em que a doença não confere imunidade passando o indivíduo assim de infectado para suscetível novamente (Figura 2.1a). Um caso particular deste modelo é quando o indivíduo uma vez infectado, nunca se recupera da doença. Neste caso tem-se o modelo *Suscetível-Infetado (SI)* (Figura 2.1b). Os modelos SIS são apropriados para modelar várias doenças causadas por agentes bacterianos, nas quais a recuperação não protege contra uma reinfecção, como a meningite meningocócica, a peste, muitas doenças venéreas, e também por protozoários, como a malária e a doença do sono.

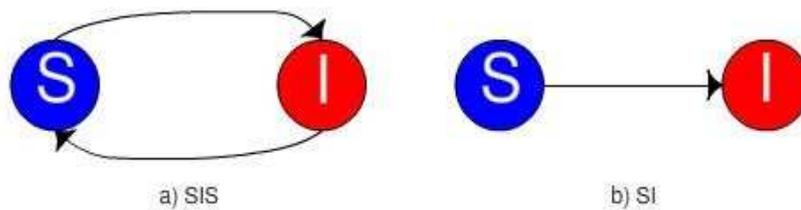


Figura 2.1: Modelos compartimentais dos tipos SIS e SI

- **Modelo Suscetível-Infetado-Recuperado (SIR)**, relacionado com as doenças em que os indivíduos infecciosos podem recuperar-se e adquirir imunidade permanente. Doenças infecciosas que ocorrem com maior frequência na infância, como rubéola, varicela, sarampo e caxumba são exemplos de doenças que costumam ser modeladas através de modelos SIR. A Figura 2.2 mostra o modelo compartimental do tipo SIR.

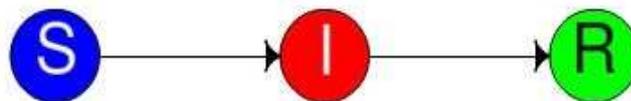


Figura 2.2: Modelo compartimental do tipo SIR

- **Modelo Suscetível-Infetado-Recuperado-Suscetível (SIRS)**, idêntico ao anterior, porém aplicável a casos em que a imunidade adquirida pelo indivíduo ao recuperar-se não é permante, passando o indivíduo a ser suscetível depois de certo tempo (imunidade temporária) ou, no caso em que a imunidade obtida, desde o primeiro momento, não proporciona proteção total (imunidade parcial). O vírus da gripe é talvez o exemplo mais familiar. A gripe adquirida num inverno confere imunidade apenas parcial contra a gripe do inverno seguinte. A Figura 2.3 mostra o modelo compartimental do tipo SIRS.

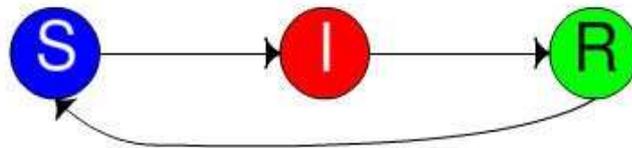


Figura 2.3: Modelo compartimental do tipo SIRS

- **Modelo Suscetível-Infetado-Recuperado-Imunidade Cruzada (SIRC)**, apresenta diferença no fluxograma (Figura 2.4) em relação aos outros modelos apresentados (Figuras 2.1, 2.2 e 2.3) já que, após o período de imunidade de uma infecção os indivíduos passam a fazer parte da classe C. Ou seja, como dito anteriormente os indivíduos presentes no compartimento R são aqueles que possuem imunidade total à doença, após o período de imunidade, os indivíduos são movidos para a classe C pois eles perdem parte da imunidade (isso ocorre quando surge um novo tipo da doença). Então, os indivíduos passam a se movimentar da classe C para os outros compartimentos pois: eles podem se tornar totalmente suscetíveis à doença, passando para ao compartimento S; eles podem contrair a infecção, sendo movidos para a classe dos infetados, I; ou eles podem adquirir total imunidade da doença, passando à classe R.

Todos estes modelos podem ainda acomodar a existência de uma categoria de indivíduos latentes (E), isto é, já infetados mas não-transmissores, situada entre a classe dos suscetíveis e a dos infeciosos. Porém esta classe e a dos imunes passivos (M) comumente são omitidas do modelo compartimental da doença, pois não são cruciais na interação suscetível-infecioso, ou seja, o comportamento qualitativo dos modelos é essencialmente idêntico, quer se inclua ou não

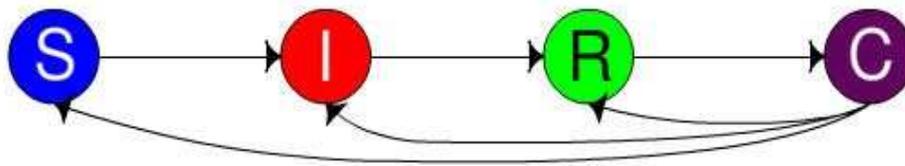


Figura 2.4: Modelo compartimental do tipo SIRC

o período de latência e imunidade passiva [28]. Usando essas classes pode-se obter uma série de modelos, tais como o MSEIR, MSEIRS, SEIR, SEIRS, SIR, SIRS, SIRC, SEIRC, SIRCS, SEI, SEIS, SI, e SIS [10][30].

2.3 O processo de transmissão

A dinâmica de uma doença transmissível consiste, essencialmente, em como a quantidade de indivíduos pertencentes a cada um dos compartimentos M, S, E, I, C e R, varia à medida que o tempo passa. Detalhes de como esta variação ocorre são apresentados a seguir.

2.3.1 Contato

Durante um certo período de tempo, um indivíduo infectado mantém contato com outros indivíduos da população, sejam eles da fração M, S, E, I, C ou R. Durante este contato, a infecção pode ou não ser transmitida. Uma interação entre dois indivíduos em que a epidemia pode ser propagada é chamada **contato adequado**.

Designa-se por contato adequado um contato em que a doença pode ser efetivamente transmitida. Desta maneira, o adjetivo “adequado” serve para distinguir entre contatos em que não é possível ocorrer contágio mesmo que um dos indivíduos seja suscetível e outro infectado, e os que permitem este contágio [28].

Por exemplo, um contato sexual de uma pessoa soropositiva ao HIV com outra pessoa qualquer (esteja ela na classe suscetível ou infectada) é um contato adequado. Já um contato social destas mesmas pessoas não é considerado adequado, pois o vírus não pode ser transmitido².

²As três principais vias de transmissão do HIV são: contato sexual; exposição a fluidos contaminados; e de mãe para filho durante o parto ou por meio da amamentação [46].

Pode-se haver contatos adequados com indivíduos imunes, suscetíveis, latentes, removidos e outros infectados. Contudo, apenas no caso do contato adequado entre um indivíduo infectado com um outro suscetível, há a possibilidade da ocorrência do contágio.

No texto que segue, sempre que se utilizar o termo **contato**, pretende-se falar de contato adequado.

2.3.2 Incidência

O número médio de contatos de um indivíduo infeccioso em um certo período de tempo é designado por β , que se refere aos contatos do indivíduo infeccioso com indivíduos de todas as categorias (M, S, E, I, R, C). Considera-se que todas estas classes estão *homogeneamente misturadas* em uma população muito grande. Sendo assim, a proporção de contatos entre o indivíduo infectado com indivíduos suscetíveis será equivalente à proporção de suscetíveis na população ($s = S/N$). Logo, o número médio de contatos com indivíduos suscetíveis durante este período de tempo será βs .

A **incidência** de uma doença é dada pelo número médio total de suscetíveis infectados por período de tempo. Se *um* infectado gera βs novos infectados por determinado período de tempo, para obter o valor da incidência basta multiplicar βs pelo número total de infectados, I . Assim será:

$$\beta s I = \beta S I / N = \beta i S$$

2.3.3 Força de infecção

Sendo a quantidade $\beta i S$ o número de suscetíveis convertidos em infectados durante o tempo a que β se refere, entende-se que βi representa a força exercida pela infecção sobre os suscetíveis, tentando infectá-los. Deste modo, pode-se designar βi por **força de infecção**, aqui representada por λ .

A força de infecção pode também ser entendida como a probabilidade, por unidade de tempo, de que um suscetível seja contaminado, passando para a classe dos infectados [28]. A probabilidade pode ser estimada a partir de dados epidemiológicos reais.

2.3.4 Período médio de infecção

O **período médio de infecção** (pmi) é o tempo médio durante a qual um indivíduo permanece na categoria dos infecciosos.

Há duas razões pelas quais a categoria I perde indivíduos: porque os indivíduos se recuperam da infecção ou porque morrem. Se por exemplo, a taxa de mortalidade (d) ou de recuperação da doença (c) for muito grande, os indivíduos não permanecerão por muito tempo dentro desta categoria, ou seja, o período médio de permanência no estado infeccioso será curto. Simetricamente, se a mortalidade e recuperação da doença forem muito fracas, o tempo médio dos indivíduos na classe I será prolongado.

O efeito conjunto destas taxas é $c + d$, e portanto, o **período médio de infecção** é o seu inverso:

$$pmi = \frac{1}{c + d}$$

2.3.5 Medida da capacidade máxima da doença

É a medida da capacidade máxima da doença de aumentar o número de infectados [30]. Supondo-se um indivíduo infectado sendo introduzido em uma grande população inteiramente suscetível. R_0 é o número médio de infecções secundárias causadas pela infecção primária em uma população inteiramente suscetível [28].

Sendo β o número médio de contatos, por infeccioso, em uma unidade de tempo, para saber qual o número médio de contatos de um infeccioso pode-se multiplicar β pelo pmi (se ambos estiverem medidos na mesma unidade de tempo). O R_0 é então:

$$R_0 = \beta \frac{1}{c + d}$$

O R_0 e o λ são medidas complementares para modelar o processo de transmissão. Enquanto o R_0 é o número de contatos que o infeccioso tem enquanto esta infectado, λ é a probabilidade de um indivíduo suscetível ser infectado em uma unidade de tempo.

2.3.6 Número de substitutos

A partir de um certo período de tempo, há tantos indivíduos não-suscetíveis na população, que só uma pequena parte dos contatos ocorrem com suscetíveis, diminuindo assim a força da propagação da doença.

Para que a infecção possa se propagar em uma população, é necessário que cada infeccioso infecte, em média, mais de um suscetível, antes de deixar o compartimento I, ou seja, ele deve deixar mais de um substituto.

Se a média de substitutos por infectado for menor que um, significa que a epidemia está perdendo força.

O número de substitutos é dado por R_0s , ou seja, o número de indivíduos infectados por infeccioso.

2.3.7 Anatomia de uma epidemia

Assumindo que R_0 é constante, pode-se entender a anatomia de uma epidemia como:

- a princípio, o número de novos casos, ainda baixo, começa a subir exponencialmente ($R_0s > 1$);
- após ter infectado uma grande quantidade de indivíduos, o número de novos casos diminui e, a seguir, o número de infectados atinge um ponto máximo ($R_0s = 1$);
- por conseguinte, decresce ($R_0s < 1$), removendo indivíduos da classe dos infectados a uma taxa maior que o número de substitutos.

A Figura 2.5 ilustra a evolução do número de substitutos durante uma epidemia.

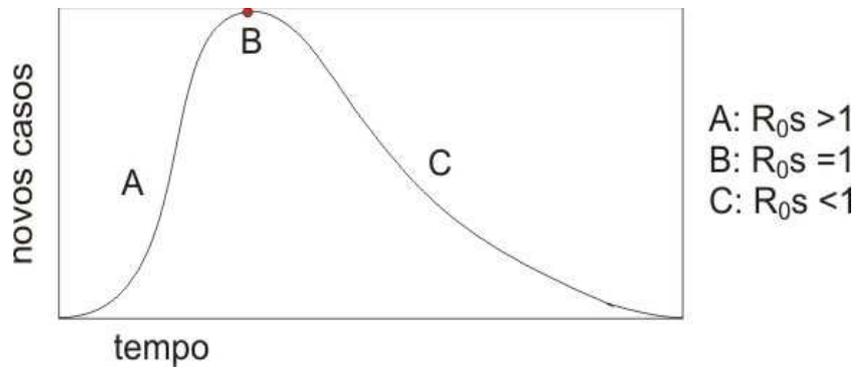


Figura 2.5: Evolução do número de substitutos em uma epidemia

2.4 Considerações finais

Uma vez abordado o tema Epidemiologia Matemática, detalhando as principais características dos modelos epidemiológicos e explicando o processo de transmissão de uma epidemia, pode-se pesquisar o estado da arte quando se fala em Epidemiologia Computacional, estudando os métodos, técnicas e abordagens mais utilizadas para simulações epidemiológicas.

O objetivo da revisão bibliográfica feita neste capítulo, é agregar os subsídios necessários ao completo entendimento dos capítulos posteriores.

Capítulo 3

Abordagens Matemáticas e Computacionais na Modelagem de Epidemias

A modelagem matemática e computacional de fenômenos epidemiológicos agrega conhecimentos, procedimentos e técnicas que permitem contruir um modelo a partir do fenômeno real, unindo a Matemática, a Computação e a Biologia. Sob hipóteses e considerações físicas, fisiológicas e biológicas, pode-se afirmar que o modelo computacional fornece uma aproximação da realidade.

A seguir, são apresentadas algumas das abordagens computacionais que podem ser empregadas na modelagem de fenômenos epidemiológicos.

3.1 Baseado em equações

O estudo da dinâmica de uma doença transmissível consiste essencialmente em esclarecer como a quantidade de indivíduos pertencentes a cada um dos compartimentos M, S, E, I, C e R , varia conforme o tempo passa.

Matematicamente, a variação de uma variável y em função de outra, x , é medida pela derivada da primeira em relação à segunda. Se, por exemplo, o número de suscetíveis S na população for dado em função do tempo (t), ou seja,

$$S = S(t)$$

então a variação de S à medida que t varia será dada por

$$\frac{dS}{dt}$$

e isto se aplica a qualquer compartimento da epidemia [43]. Portanto, todos os modelos epidemiológicos podem ser representados por sistemas de equações diferenciais.

O processo básico consiste em elaborar equações apropriadas para cada tipo de modelo epidemiológico. A modelagem destas classes epidemiológicas gera os modelos do tipo compartimental [22].

Assim, estes modelos empregados no estudo da dinâmica de epidemias do tipo compartimental são determinísticos, assumindo intrinsecamente que os contatos entre indivíduos de diferentes comunidades é uniforme, sem considerar que tais contatos podem ter fatores relacionais, distintos ou comuns, gerando representação não condizente com o real.

Por exemplo, os modelos compartimentais não representam a persistência de doenças infecciosas, já que ele considera homogênea a distribuição espacial e temporal dos indivíduos. Sendo assim, é frequente a crítica a estes tipos de modelos, já que a aproximação proposta não representa a realidade de uma sociedade.

Porém, uma modelagem determinística mais realística invariavelmente resulta em um modelo matematicamente intratável, não existindo uma solução analítica e, na maioria das vezes, mesmo sua análise matemática somente pode ser elaborada para situações particulares ou considerando-se uma série de simplificações.

Soluções decorrentes do uso de métodos de modelagem de solução, cujas abordagens são discretas, probabilísticas ou heurísticas tratam cada elemento componente da comunidade de modo individual e, quando for o caso, com características e estruturas próprias. Estudos mostram que tais abordagens são suficientemente flexíveis para permitir o emprego detalhado de parâmetros com maior significado biológico do que os modelos determinísticos.

Não obstante a estas restrições ao emprego de modelos matemáticos baseados em equações, ordinárias ou parciais, em geral, a simulação de processos epidemiológicos via modelos baseados em indivíduos, ou outros modelos probabilísticos, utiliza como base na sua modelagem os possíveis estados de um indivíduo, empregando entre as regras de transição, taxas ou probabilidades, os tradicionais blocos de modelos compartimentais.

Assim sendo, as ideias subjacentes à modelagem compartimental são utilizadas mesmo

nesses modelos estocásticos e baseados em indivíduos. Também os resultados das análises matemáticas realizadas com simples modelos do tipo comportamental, como o exemplificado SIR, são empregados para estabelecer parâmetros, especificações ou mesmo, para certa análise quantitativa e qualitativa da solução desses tipos de modelos.

Diversos modelos epidemiológicos foram propostos utilizando diferentes abordagens para a simulação da dinâmica de doenças, alguns destes modelos são descritos nas próximas seções.

Exemplos de trabalhos relacionados com epidemiologia matemática e computacional são [2][43]. Neles os autores realizaram estudos numérico-computacionais de modelos matemáticos epidemiológicos. Há ainda modelagens matemáticas sobre a dengue, apresentadas por Tran e Raffy em [52] e a proposta de um modelo discreto da propagação feita por Dietrich e Muhammad [14]. Convém ainda citar o trabalho de Hethcote [30], que é entendido como item de estudo obrigatório na área de epidemiologia matemática.

3.2 Baseado em autômatos celulares

Introduzidos por Von Neumann e Ulam [31], os autômatos celulares (AC) são usados para estudar processos de crescimento e auto-reprodução. Qualquer sistema com um grande número de elementos idênticos interagindo localmente pode ser modelado usando essa técnica[16][22].

Um autômato celular consiste basicamente em [44][11]:

- um espaço euclidiano dividido em uma malha de células idênticas;
- uma vizinhança, que corresponde a uma seleção de células próximas;
- um conjunto discreto de estados de células;
- um conjunto de regras que determinam o estado de cada célula em função de sua vizinhança;
- tempo discreto e finito.

Quando esta abordagem é usada na simulação de epidemias, cada célula representa uma área (pode ser uma quadra, bairro, cidade, dependendo do mundo a ser modelado), que pode se encontrar em um dos compartimentos do modelo (M, S, E, I ou R, dependendo do modelo

compartimental escolhido para a simulação). A probabilidade de uma célula livre de infecção passar ao compartimento dos infecciosos (significa que algum indivíduo presente naquela área apresenta a doença) depende da função de vizinhança [22][16].

A Figura 3.1 mostra a transmissão de um estado entre células para as mais importantes funções de vizinhança. Em amarelo, pode-se ver o resultado da função de vizinhança de Von Neumann, onde a propagação da epidemia se dá na célula atual, e nas células que se encontram nas posições norte, sul, leste e oeste desta, e em azul o resultado da vizinhança de Moore, formada pela célula atual juntamente com as oito células mais próximas a ela [56].

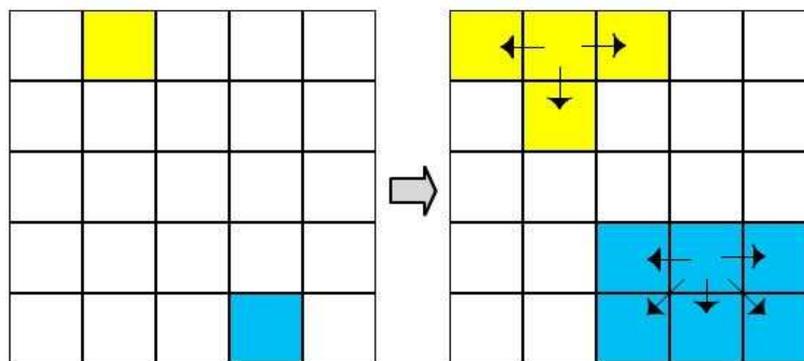


Figura 3.1: Transmissão em Autômatos Celulares

Um caso de estudo sobre a propagação da gripe aviária na Ásia utilizando AC foi feito por Situngkir [49]. Uma ferramenta capaz de simular o modelo compartimental SIS foi desenvolvida por White *et al* em [56]. Vale citar o trabalho de Merchant [38] onde foi implementado um protótipo de *software* utilizando AC para simulação da propagação da malária na África considerando fatores como vacinação e a informação da população.

3.3 Método de Monte Carlo

O método de Monte Carlo desempenha um papel muito importante em simulações, pois é utilizado para aproximar, de forma tão precisa quanto possível, as propriedades de um sistema dentro de uma determinada distribuição estatística. Essa técnica faz uso intenso de números aleatórios para estabelecer distribuições probabilísticas uniformes [1].

Em geral, a versão deste algoritmo utilizada em simulações epidemiológicas pode ser resumida nos seguintes passos:

1. Escolhe-se um indivíduo da população aleatoriamente.
2. Se este indivíduo pertencer ao compartimento dos suscetíveis, verifica-se se o número de vizinhos infectados é maior que zero.
3. Se o número de indivíduos infectados na vizinhança for maior que zero o próximo passo é executado, caso contrário, repete-se o passo 1.
4. Calcula-se a probabilidade de infecção do indivíduo.
5. Sorteia-se um número randômico.
6. Se o número randômico for maior ou igual a probabilidade de infecção do indivíduo, este passa para a classe dos infectados, caso contrário, continua como suscetível.
7. Repete-se estes passos até que a quantidade de ciclos definidos previamente seja alcançada.

Como pode-se observar, este algoritmo aplica suas regras na vizinhança de um único indivíduo a cada iteração. Ao comparar este modelo com uma sociedade real, percebe-se que é como se uma única pessoa tivesse autorização para se mover por vez enquanto todo o resto da comunidade permanece parada, e outra pessoa começa a se mover assim que a primeira estiver parada, ou seja, não representa corretamente a realidade dinâmica de uma sociedade.

Karsten *et al* [34][35] usou este método para desenvolver um modelo de simulação que descreve o propagação da febre suína em criações. Este algoritmo também foi usado por Jorgensen [33] para simulação de epidemias em populações suínas, comparando as diferentes doenças com os métodos de prevenção e verificando a melhor ação frente a um surto. Além disso, um protótipo para simulações epidêmicas em populações humanas utilizando este método foi desenvolvido por Ferreira [17], modelando a dinâmica da tuberculose.

3.4 Baseado em Lógica *Fuzzy*

A Lógica *Fuzzy*, também conhecida como Lógica Nebulosa ou Difusa, foi publicada em 1965 pelo professor Lotfi Asker Zadeh, em um artigo intitulado “*Fuzzy Sets*” [61]. Baseada na Teoria dos Conjuntos Clássica [8], a lógica *Fuzzy* foi criada com o objetivo de traduzir informações vagas, imprecisas e ambíguas em valores numéricos [42]. O fato de inferir conclusões e gerar respostas a partir de dados imprecisos confere aos sistemas *Fuzzy* uma habilidade seme-

lhante à faculdade do raciocínio, tradicionalmente humana [51].

Na Teoria dos Conjuntos Clássica, cujos conjuntos são denominados *crisp*, um determinado elemento do universo pertence ou não pertence a um dado conjunto. De modo análogo, uma proposição lógica possui dois extremos: totalmente verdadeira ou totalmente falsa [51]. Em contrapartida, na lógica *Fuzzy*, cujos conjuntos são denominados nebulosos, existe um grau de pertinência de cada elemento a um determinado conjunto. Assim, uma informação pode ser parcialmente verdadeira ou parcialmente falsa, variando seu grau de verdade entre 0 e 1 [42]. Tais como os conjuntos *crisp*, os conjuntos nebulosos possuem as operações de união, intersecção e negação (ou complemento) [48].

No controle *Fuzzy*, a idéia básica é modelar as ações a partir de conhecimento especialista em vez de modelar o processo em si, por meio de modelagem matemática. Um controlador nebuloso é um sistema especialista simplificado, onde a consequência de uma regra não é aplicada como antecedente de outra [48].

Os sistemas *Fuzzy* são fáceis de compreender e utilizar na prática. A lógica *Fuzzy* possui aplicações em diversas áreas, tais como Inteligência Artificial, processamento de imagens, otimização, reconhecimento de padrões, tomada de decisões, entre outras [48].

Alguns esforços relacionando a lógica *Fuzzy* com a epidemiologia podem ser listados, como a utilização da teoria nebulosa no modelo epidemiológico SI [5] e SIR [41]. Um estudo unindo as técnicas de AC com a lógica nebulosa para analisar o espalhamento geográfico de doenças do modelo SIR foi desenvolvido por Peixoto [44].

3.5 Redes complexas

As redes, ou grafos, são estruturas formadas por um conjunto de nós e um conjunto de arestas que conectam estes nós, e podem ser utilizadas para modelar praticamente qualquer estrutura discreta.

Uma das formas utilizadas na literatura para modelar contatos heterogêneos é o uso de redes complexas, que podem representar uma sociedade usando grafos onde os vértices são as pessoas e as arestas representam os contatos pelos quais a doença se propaga. As redes complexas são comumente usadas para representar um grupo de indivíduos que interage ou se relaciona de alguma maneira [32].

Como pode ser visto na Figura 3.2, redes complexas podem ser categorizadas em três tipos: redes de mundo pequeno, redes de escala livre e redes randômicas. Uma diagramação destes modelos de rede, que vêm sendo utilizadas na simulação da dinâmica de epidemias, podem ser vistos na Figura 3.2.

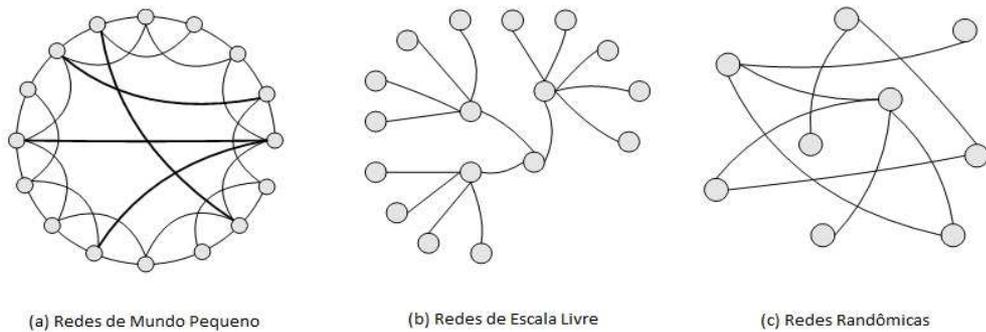


Figura 3.2: Tipos de redes complexas

3.5.1 Redes de mundo pequeno

Redes de mundo pequeno tem base na observação de que existe na natureza uma variedade de sistemas cujos elementos estão relativamente próximos uns dos outros, apesar do tamanho significativamente grande destes sistemas (por exemplo, uma sociedade humana). Por esse motivo, são bastante populares em simulações sociais [3][39].

Na simulação de epidemias, a aplicação de redes de mundo pequeno faz com que o nível de infecciosidade dependa da conexão topológica do grafo. A alta clusterização destas redes faz com que a doença se propague localmente em pouco tempo, enquanto que o curto caminho da rede permite que a epidemia alcance grandes distâncias no grafo [22][32].

Huang *et al* [32] criaram um ambiente de simulação de propagação de doenças epidemiológicas ou valores culturais utilizando redes complexas de mundo pequeno. Para Zhou [62] a teoria de mundo pequeno é útil para analisar este tipo de propagação, considerando que essa forma de organização de sociedade realmente existe em populações humanas e influencia no processo de propagação.

3.5.2 Redes de escala livre

No início de uma simulação com redes de escala livre, o grafo possui um pequeno número de nós. Conforme o tempo passa, novos nós são conectados aos existentes, sendo que, quanto maior for o número de conexões de um nó, maior será sua probabilidade de ganhar a conexão com o novo nó [32][54]. Essa característica resulta em redes com poucos vértices altamente conectados, denominados *hubs*, e muito vértices com poucas conexões. Esta dinamicidade de conexões elimina a possibilidade de se falar em escala ou em número médio de arestas.

3.5.3 Redes randômicas

São geradas adicionando-se conexões em pares de nós randomicamente escolhidos. A baixa clusterização, faz este tipo de rede complexa ser pouco aplicável a simulações de sociedades reais [32][39][54].

3.6 Considerações finais

Este capítulo estudou o estado da arte em epidemiologia computacional, com seus métodos, técnicas e abordagens mais utilizadas para simulação epidemiológica. O propósito deste estudo é documentar o que está sendo feito atualmente na área.

A técnica de Agentes Computacionais, utilizada no desenvolvimento deste trabalho, é estudada no Capítulo 4.

Capítulo 4

Agentes Computacionais

Este capítulo provê uma visão geral sobre Agentes Computacionais, enfatizando aspectos que são utilizados no decorrer deste trabalho.

4.1 Definições para Agentes Computacionais

Não há ainda uma definição unânime sobre o que são agentes computacionais. Diferentes definições dadas por autores, cujas obras foram consultadas durante o desenvolvimento deste trabalho, são as seguintes:

- para a FIPA (*Foundation for Intelligent Physical Agent*) um agente computacional é uma entidade de software que encapsula seu estado, comportamento, processo de controle e as habilidades de interagir e se comunicar com outras entidades [18]. Essa definição trata agentes como um objeto, elevando portanto, o nível de abstração;
- a definição dada por Russel & Norvig em [47] assume que um agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e de agir sobre esse ambiente por intermédio de atuadores;
- Genesereth somente considera uma entidade como um agente se esta for capaz de se comunicar com outras entidades [27];
- e para Franklin um agente é um sistema que faz parte de um ambiente, sentindo-o e agindo sobre ele ao longo do tempo, cumprindo suas tarefas [19].

Neste trabalho é aceita a definição dada por Michael Wooldridge: Um agente autônomo é um sistema de computador, que é situado em algum ambiente, e é capaz de decidir o que fazer e quando fazer, a fim de alcançar seus objetivos. Deste modo, um agente autônomo tem por característica principal a capacidade de escolher entre diversas possibilidades de comportamento [59][9].

4.2 Propriedades dos agentes

As propriedades dos agentes estão diretamente relacionadas com o tipo do agente implementado e com as suas funcionalidades. As propriedades mais comumente implementadas são listadas a seguir:

- **Autonomia:** um agente interaje com o ambiente sem orientações ou intervenções, sejam elas humanas ou não [57][47];
- **Comunicabilidade:** um agente tem a capacidade de se comunicar com outros agentes visando completar suas tarefas ou ajudar aos demais [58]. É interessante ressaltar que a concepção de comunicação entre agentes é referente à troca de mensagens entre os mesmos, onde o conhecimento contido nas mensagens deve estar escrito em algum formalismo de representação de conhecimento entendido por todos os envolvidos na comunicação [20]. Ou seja, o conceito de comunicação entre agentes difere do conceito de comunicação utilizado em outras áreas da computação.
- **Cooperação:** os agentes são capazes de trabalhar em conjunto a fim de concluírem tarefas de interesse comum [59];
- **Inteligência:** essa propriedade é verificada nos agentes inteligentes [60]. Estes agentes podem decidir a melhor ação a ser tomada, ou, podem ter a capacidade de aprender [47];
- **Reatividade:** um agente é capaz de perceber alguma alteração em seu ambiente e de reagir adequadamente às mudanças ocorridas [47].

Um agente não precisa possuir todas essas características ou pode possuir outras características além destas (como veracidade, benevolência ou continuidade temporal, por exemplo). As suas habilidades estão intimamente ligadas às propriedades que possui.

4.3 Classificação geral dos agentes

Como diferentes agentes podem possuir diferentes conjuntos de propriedades, tem-se então agentes com comportamentos diferentes, os quais são classificados de diferentes maneiras. As duas classificações mais comuns estão descritas a seguir.

4.3.1 Agentes cognitivos

São baseados em modelos organizacionais humanos, onde cada agente é um sistema complexo e computacionalmente pesado. As principais características desses agentes são [6]:

- Podem manter um histórico das iterações passadas, isso é, têm memória;
- A comunicação entre agentes é feita de modo direto, pelo envio e recebimento de mensagens, e seus sistemas de percepção e comunicação são distintos;
- Tais agentes raciocinam e decidem quais objetivos querem alcançar e quais ações devem ser executadas a seguir para alcançá-los;
- Uma sociedade de agentes cognitivos geralmente possui poucos agentes;
- Há uma representação explícita de ambiente e de membros da sociedade.

4.3.2 Agentes reativos

Baseados em modelos de organização biológica, como, por exemplo, as sociedades de formigas ou cupins.

Embora uma formiga sozinha não possa ser considerada uma entidade inteligente, o formigueiro como um todo apresenta um comportamento visivelmente inteligente, no sentido em que existe uma busca de alimentos e posterior estocagem, uma organização da reprodução, com berçários e enfermeiras, etc.

Características do modelo de funcionamento de um agente reativo são listados abaixo [6]:

- Não há representação explícita de conhecimento, sendo que a idéia de inteligência emerge da interação no grupo;

- O agente não possui memória, de modo que, o resultado de uma ação já ocorrida não possui influência nenhuma sobre o seu comportamento em ações futuras;
- Normalmente, as sociedades de agentes reativos são numerosas;
- Os agentes reativos não se comunicam entre si;
- Não há planejamento de ações futuras;
- Cada agente possui um conjunto de regras muito simples;
- Todas as informações relativas ao seu comportamento estão no ambiente e suas reações dependem unicamente da percepção deste ambiente.

Tomando por base essas características, conclui-se que os agentes reativos são simples. Mas apesar desta simplicidade torná-los pouco úteis isoladamente, os agentes reativos tem grande força trabalhando em grupo.

Com relação ao ambiente aqui proposto, a intenção é analisar o comportamento de uma epidemia quando inserida em um grande grupo de agentes. Para esse fim, é adequado o uso de agentes reativos.

4.4 Arquitetura

Uma arquitetura de software pode ser descrita como sendo a configuração dos componentes que constituem um sistema. Sendo assim, quando se fala de arquitetura de um sistema de agentes, não é somente a configuração do agente em si que importa, mas o modo de organização dos agentes dentro do sistema e como estão estruturados seus relacionamentos e interações [60].

A seguir pode ser vista uma classificação para arquiteturas de agentes [47]:

- **Agentes Reativos Simples:** Utilizam um conjunto de regras pré-estabelecidas, e selecionam ações unicamente com base na percepção atual.
- **Agentes Reativos Baseados em Modelo:** Possuem uma representação do estado interno e um histórico de percepções. A reatividade está condicionada a experiências anteriores.

- **Agentes Baseados em Objetivos:** Para atuar, utilizam informações sobre os seus objetivos. Neste caso, o agente considera a melhor ação a ser tomada. O agente pode apresentar diferentes comportamentos para o mesmo estado do mundo, dependendo do seu objetivo.
- **Agentes Baseados na Utilidade:** Neste caso a *utilidade* representa uma função que mapeia o *grau de satisfação* do agente. Se um estado do mundo é mais desejável que outro, então este terá maior utilidade para o agente. Os agentes baseados em utilidade conseguem ser mais racionais, pois possuem capacidades para avaliar a utilidade da execução de determinada ação.

Concluindo esta seção, convém dizer que, para o ambiente de interação entre agentes ao qual este trabalho propõe a implementação, é utilizada a arquitetura de Agentes Reativos Simples, principalmente por este ambiente ter seu foco de aplicação ao estudo do comportamento da sociedade de agentes, e não no funcionamento de um único, ou de cada agente da comunidade.

4.5 Ambiente

De um modo bastante simplificado, um agente pode ser entendido como um componente de um sistema que *atua* sobre outras partes. Por atuar, entende-se a capacidade de efetuar leitura e escrita dos dados pertencentes aos demais componentes, que estão sofrendo reflexos da atuação do agente.

Um *ambiente*, por sua vez, pode ser compreendido como o **componente passivo que tem como única função disponibilizar seus dados à ação do agente**. Ou seja, o ambiente consiste de toda e qualquer fonte de estímulo sensorial aos agentes, sendo passível de sua atuação.

A seguir são apresentados alguns critérios pelos quais os ambientes podem ser categorizados [47]:

- **Acessível versus Inacessível:** Um ambiente *acessível* permite que um agente acesse todas as suas informações, ou seja, os sensores dos seus agentes podem detectar todos os aspectos que são relevantes à escolha de suas próximas ações. Se isso não se verificar, o ambiente é então chamado *inacessível*. O uso de um ambiente acessível é conveniente, pois o agente não necessita armazenar nenhuma informação referente a este, ou seja,

quanto mais acessível um ambiente for, mais simples será projetar e contruir agentes para nele operarem.

- **Determinístico versus Estocástico:** Um ambiente é considerado *determinístico* caso seu próximo estado seja, do ponto de vista do agente, completamente determinável a partir do estado atual. Se isso não for verificado, o ambiente é então chamado *estocástico*.
- **Estático versus Dinâmico:** Um ambiente *estático*, ao contrário do que acontece em um ambiente *dinâmico*, permanece inalterado durante o tempo em que o agente está processando a próxima ação. Deste modo, é fácil perceber que os ambientes reais são todos dinâmicos. O ambiente estático é inerentemente mais simples, pois o agente recolhe uma determinada informação uma única vez, sem se preocupar se haverá algum tipo de mudança no ambiente.
- **Discreto versus Contínuo:** Tendo em vista a continuidade, um ambiente é classificado como *discreto* caso suas variáveis principais assim o sejam. Em outras palavras, para um ambiente de jogo de xadrez, por exemplo, sendo o tempo relevante à ação de cada agente determinado pelas jogadas, e o número de estados distintos finito, o ambiente é claramente discreto. Em outras situações, entretanto, em que as variáveis são contínuas (como em um agente robótico que explora um território, cuja extensão é caracterizada por infinitas posições), o ambiente é dito *contínuo*.

Analisando as propriedades acima, pode-se afirmar que os ambientes menos complexos são os ambientes acessíveis, estocásticos, estáticos e discretos. Sendo assim, o ambiente da ferramenta desenvolvida tem o compromisso de ser o mais acessível, estocástico, estático e discreto possível, a fim de ser capaz de “abrigar” o grande número de agentes reativos proposto.

4.6 Considerações finais

Este capítulo abordou as características principais de Agentes Computacionais, detalhou Agentes Reativos e Cognitivos, apresentou definições para Ambiente de Agentes, Sistemas Multiagente e Solução Distribuída de Problemas e, ao fim, conceituou Organização de Agentes. O objetivo desta revisão bibliográfica foi o de fornecer subsídios aos capítulos posteriores, a fim de que os conceitos vistos possam ser usados sem que haja dificuldade de entendimento.

Tendo em mente que, no capítulo seguinte será apresentada uma modelagem da ferramenta para simulação de eventos epidemiológicos desenvolvida, pode-se identificar, da teoria de sistemas multiagentes apresentada, quais conceitos são úteis em tal tarefa.

O fato da proposta do simulador epidemiológico ter o foco em epidemias inseridas em sociedades de seres humanos, faz o paradigma de sistemas multiagentes um bom candidato a orientador de sua implementação. De fato, é intuitivo o pensamento de que **cada pessoa da sociedade será um agente**.

Além disso, tomando por base as informações referentes à epidemiologia dadas no Capítulo 2, pode-se concluir que, para o propósito de disseminação de doenças em uma sociedade, um indivíduo não necessita de qualquer raciocínio lógico (necessitando dele, na verdade, para evitar essa disseminação). Tal característica indica que os agentes contruídos para simula-los podem ter uma **arquitetura basicamente reativa**.

Capítulo 5

Modelagem da Ferramenta

Depois do estudo da Epidemiologia Matemática, seguida pelas abordagens matemáticas e computacionais mais utilizadas na modelagem de epidemias, as considerações apresentadas no capítulo anterior sobre Sistemas Multiagentes completam a teoria necessária à tarefa de modelagem de uma sociedade artificial que seja capaz de simular os modelos epidemiológicos mais comuns.

Deste modo, o objetivo deste capítulo é apresentar a modelagem da ferramenta desenvolvida. Na primeira seção está a descrição da modelagem do sistema multiagente como um todo. Na Seção 5.2 é apresentada a modelagem do agente e suas regras, e na Seção 5.3 apresenta a modelagem do ambiente onde os agentes estão interagindo.

5.1 Modelagem do Sistema Multiagente

Sistemas multiagente são uma forma natural de modelar sociedades devido à facilidade de representação dos indivíduos ativos originais como agentes do modelo [36]. Além disso, na própria definição de sistemas multiagente ¹ fica clara sua vocação para modelar a simulação da evolução de uma epidemia em uma sociedade.

Nestes termos, tendo em mente a construção de um sistema multiagente que simule o comportamento de uma epidemia inserida em uma sociedade de seres humanos reais, é intuitiva a idéia de que, basicamente, **cada ser humano é um agente** do sistema e interage com os demais (outros seres humanos) e com o ambiente. **O ambiente, por sua vez, é modelado como uma matriz**, por onde os seres humanos podem transitar, ocupando uma de suas posições.

¹Um grupo organizado de componentes autônomos que atuam na solução de um problema [9]

O resumo das analogias dos principais conceitos utilizados na modelagem da ferramenta desenvolvida pode ser observado na tabela 5.1.

Tabela 5.1: Conceitos e analogias utilizados na modelagem do sistema multiagentes da ferramenta desenvolvida

Conceito	Analogia
Pessoa	Agente itinerante capaz de percorrer toda a matriz-ambiente
Ambiente	Matriz M composta de 700 linhas e 1000 colunas formando um plano cartesiano com 700 mil posições para possível alocação de pessoas ou barreiras, onde $0 \leq x < 1000$ e $0 \leq y < 700$.
Posição	Local na matriz. Indexado pelo número da linha (y) e da coluna(x), representado da seguinte maneira: $M[y][x]$. Esta posição tem o exato tamanho de um agente e pode, em determinado momento durante a simulação, alocar um agente qualquer.
Barreira	Faz parte do ambiente como um de seus componentes. Não possui a capacidade de locomoção e a sua função é evitar a passagem de agentes-pessoa pela posição em que se encontra na matriz-ambiente.

A Figura 5.1 apresenta uma ilustração dos conceitos utilizados na modelagem da ferramenta.

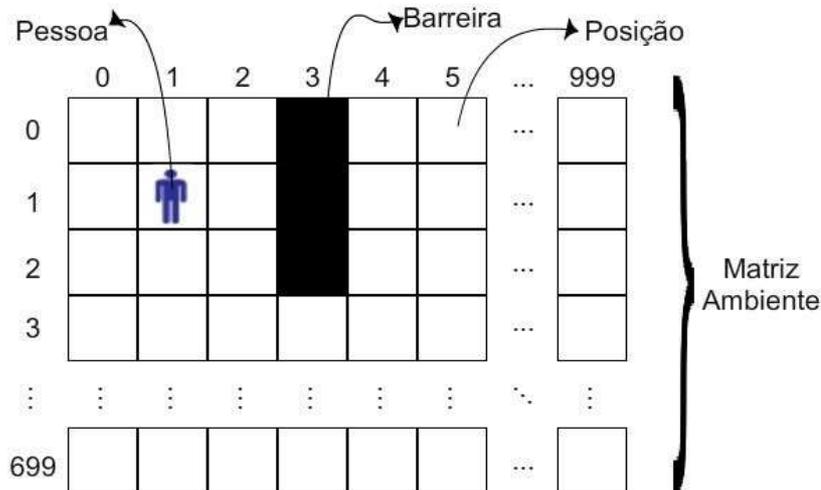


Figura 5.1: Ilustração dos conceitos usados na modelagem do sistema multiagentes da ferramenta de simulação epidemiológica

Com o intuito de apresentar de fato a modelagem adotada para o sistema multiagentes, de modo que ele simule a transmissão de uma doença em uma sociedade humana, os dois conceitos principais de sistemas multiagente (ambiente e agentes) foram modelados individualmente tal como apresentado nas seções a seguir.

5.2 Modelagem dos Agentes

Sobre a modelagem dos agentes do sistema multiagente em questão, a metáfora mais clara é a de que **agentes correspondem a pessoas**. Entretanto, outro componente menos evidente e muito importante para o desenvolvimento da simulação é a **barreira** que atua limitando a movimentação dos agentes-pessoa. Assim, as seguintes subseções descrevem a modelagem dos agentes pessoa e dos componentes barreira.

5.2.1 Pessoa

Voltando-se a atenção exclusivamente às pessoas, ainda que a modelagem de agentes computacionais possa ser feita de modo que cada agentes possua características específicas, diferenciando-os uns dos outros, a decisão tomada neste trabalho é a de que todos os agentes-pessoa serão **exatamente iguais**, tanto no comportamento, quanto nas capacidades de percepção e atuação.

A capacidade de percepção de um agente-pessoa pode-se dar da seguinte maneira:

- O agente pode **perceber a presença de outro agente ou de uma barreira** dentro de sua vizinhança. O tipo de vizinhança adotada é a vizinhança de Moore, formada pelas oito células mais próximas da posição em que o agente se encontra [56], representada em azul na Figura 5.2.
- O agente pode **perceber o seu estado atual** em relação ao modelo compartimental da doença que está sendo simulada. Os modelos compartimentais simulados na ferramenta desenvolvida são SI, SIS, SEI, SEIS, SIR, SIRS, SEIR e SEIRS. Devido à grade complexidade dos modelos compartimentais com a classe C, estes não foram implementados, sendo assim, tomados e indicados para trabalhos futuros.

Com relação à capacidade de atuação, um agente-pessoa pode atuar no ambiente das seguintes maneiras:

- O agente pode **modificar a sua posição atual** para uma outra posição da sua vizinhança, simulando a movimentação de pessoas.

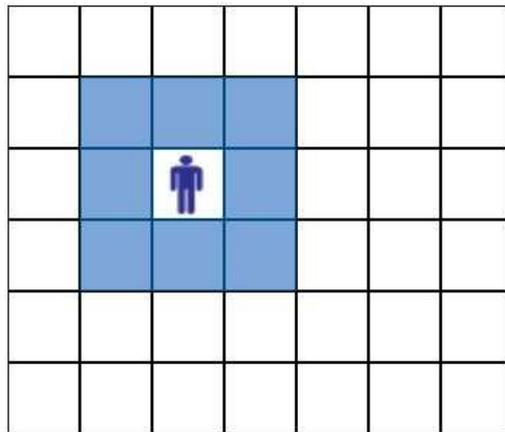


Figura 5.2: Vizinhança de Moore representada pelas células em azul

- O agente pode **modificar o seu estado interno** quando
 - a) estiver no compartimento dos indivíduos suscetíveis e um agente infectado se encontrar em sua vizinhança (de acordo com a taxa de transmissão da doença);
 - b) ocorrer o fim de um período de latência, de infecção ou de imunidade parcial.

Quanto aos estados que um agente pode assumir, estes são referentes aos compartimentos mais comuns utilizados nos modelos compartimentais de epidemias: suscetível, latente, infectado e imune (parcialmente ou não).

Referindo-se ainda a tais estados, pode-se explicá-los da seguinte maneira: todo o agente-pessoa criado no ambiente começa a simulação no estado suscetível. São inseridos alguns indivíduos infectados no ambiente a fim de proliferar a epidemia. Dependendo do modelo compartimental da doença simulada, quando cada um destes indivíduos infectados entra em contato com um indivíduo suscetível (há um indivíduo infectado dentro da vizinhança de um indivíduo suscetível) pode haver a transmissão da doença (de acordo com a taxa de transmissão da doença simulada) passando o indivíduo suscetível ao estado de latente ou infectado.

Como já foi comentado anteriormente, um indivíduo que se encontra no estado de latência é portador da doença mas não a transmite [30]. Ao fim do período de latência (que é dado respeitando o modelo da doença) o agente-pessoa passa para o estado infectado, e neste estado, ele é transmissor da doença.

Em alguns modelos compartimentais o indivíduo permanece neste estado até o fim de sua

vida (são as doenças que não possuem cura, como o HIV). Nas outras doenças (as que possuem cura), ao fim do período infeccioso o indivíduo pode passar para o estado de imune ou voltar ao estado suscetível.

Se o modelo compartimental da doença simulada possuir imunidade parcial, o agente-pessoa despenderá algum tempo neste estado e em seguida voltará ao estado suscetível. Se o modelo possuir imunidade definitiva, os indivíduos que alcançarem este estado permanecem nele até o fim da simulação.

Na Figura 5.3 pode-se ver a máquina estados resultante de todas estas interações citadas, onde os estado infectado (I) e imune (R) podem ser estados finais.

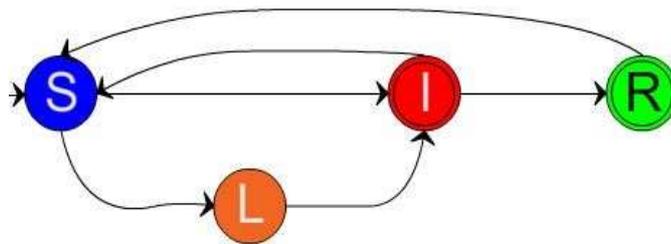


Figura 5.3: Máquina de estados utilizada na modelagem do agente-pessoa

Voltando-se a atenção à função do agente-pessoa, lembrando que cada agente é desprovido de qualquer raciocínio lógico, este deve ser reativo, implicando no fato de sua função possuir como parâmetro aos agentes presentes em sua vizinhança.

Em termos mais concretos, a função de agente pode ser codificada em um programa de agente, cuja execução incorre na determinação de uma ação específica para cada modo de atuação anteriormente identificado: uma ação de mudança de posição corrente ou uma ação de modificação do estado interno.

Com relação à ação específica de mudança de posição corrente o agente escolhe uma direção a seguir de acordo com a sua vizinhança, tendo assim oito opções: norte, sul, leste, oeste, nordeste, noroeste, sudeste, sudoeste, além da opção de permanecer na posição em que se encontra (Figura 5.4a). Depois de feita esta escolha, o agente deve decidir o número de posições que ele deseja avançar naquela direção (Figura 5.4b). Enquanto o agente está percorrendo a trajetória escolhida ele percebe somente o que está acontecendo em sua vizinhança, não sabendo o que está por encontrar no seu caminho (Figura 5.4c). Se houver uma barreira ou um outro

agente-pessoa na trajetória do agente, este deverá parar quando esta entrar em sua vizinhança e escolher um novo trajeto (Figura 5.4d).

5.2.2 Barreira

Apesar de não ser um agente computacional, faz parte da matriz-ambiente como um de seus componentes. Sua função é unicamente **evitar a passagem dos agentes-pessoa**. Isto se dá pela simples existência de uma barreira em determinada posição da matriz-ambiente, pois o agente-pessoa não pode ocupar uma célula da matriz que já esteja ocupada por outro agente ou por uma barreira.

5.3 Modelagem do Ambiente

Como já comentado anteriormente, a modelagem adotada para o ambiente do sistema multiagente é a de uma matriz com setecentas (700) linhas e mil (1000) colunas ($M[700][1000]$), onde os índices das linhas e das colunas indicam as posições da matriz, formando assim um mapa cartesiano com 700 mil posições.

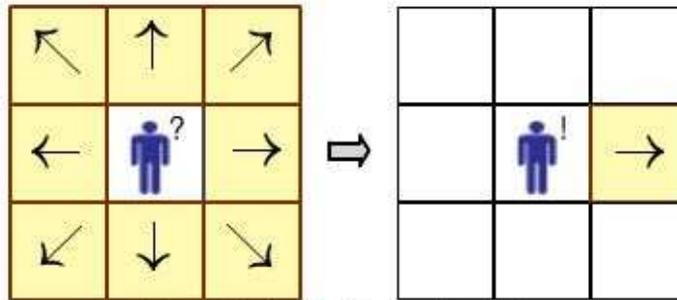
Os agentes podem se movimentar no ambiente respeitando os limites de sua composição, evitando as barreiras que são alocadas metodicamente a fim de delimitar a formação de comunidades no ambiente.

A comunicação entre as comunidades é dada por aberturas nas barreiras que as separam, representando, por exemplo, as vias principais entre bairros de um município, pelas quais os agentes-pessoa se movimentam de forma mais direcionada. Neste trabalho, é através destas conexões que os indivíduos mantêm contato uns com os outros, sendo por meio desses contatos que a doença se propaga de uma comunidade para outra.

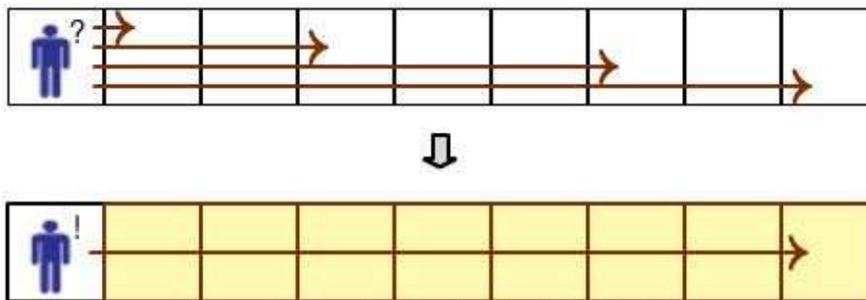
A Figura 5.5 representa um pequeno ambiente dividido em três comunidades, uma à oeste no mapa (comunidade A) e outras duas à nordeste (comunidade B) e sudoeste (comunidade C).

5.4 Considerações finais

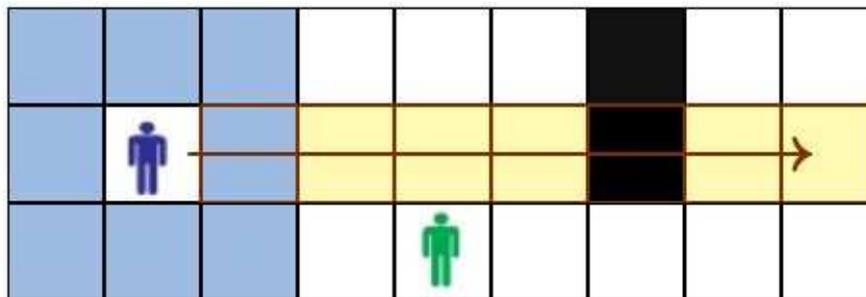
Terminada a apresentação da modelagem da ferramenta para simulação epidemiológica, a implementação em uma linguagem de programação pode ser feita, de modo que os seus resul-



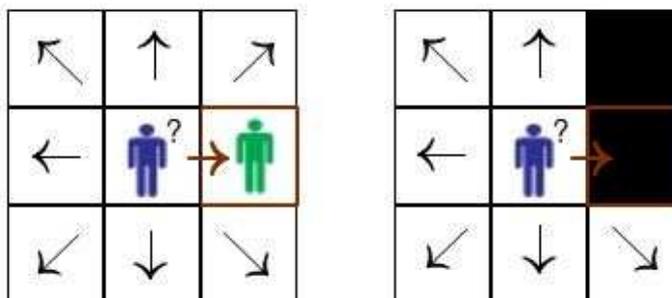
a) O agente deve decidir que direção tomar



b) A seguir o agente deve decidir o tamanho da sua trajetória.



c) Enquanto está seguindo a trajetória escolhida, o agente somente sabe (percebe do ambiente) o que está acontecendo em sua vizinhança.



d) Se o agente encontrar um outro agente pessoa ou uma barreira em sua trajetória, este deverá escolher outro trajeto para seguir

Figura 5.4: Ação referente à mudança de posição corrente

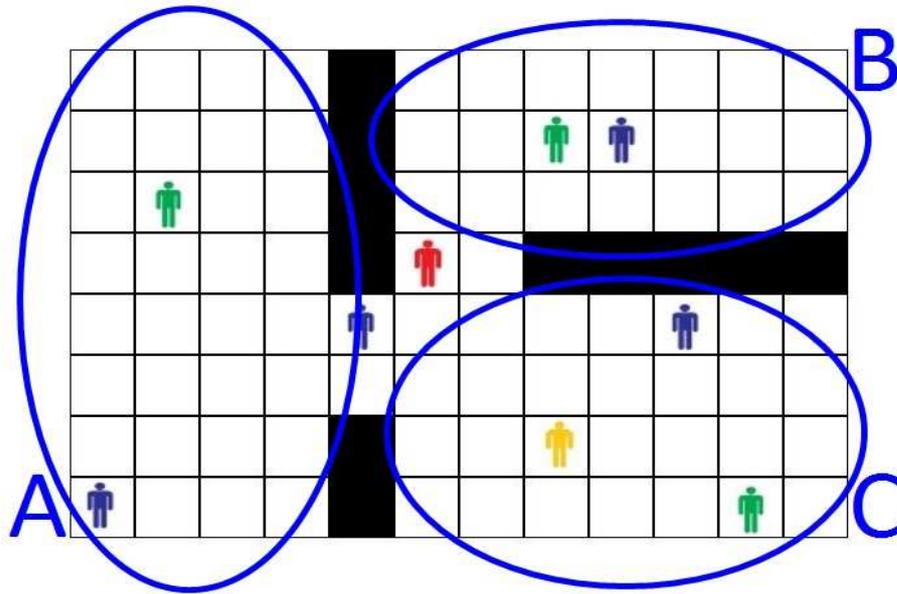


Figura 5.5: Ambiente dividido em comunidades

tados possam ser analisados, bem como comparados com outras implementações existentes na literatura.

Antes de fazê-lo, entretanto, algumas considerações devem ser feitas. Voltando-se a atenção ao ambiente, e tomando por base a sua classificação [47], o ambiente modelado através de uma matriz pode ser enquadrado da seguinte forma:

- quanto à dimensão de observação, o ambiente-matriz é totalmente *acessível*, já que, o agente-pessoa percebe todos os dados necessários para a decisão de sua atuação, sem a necessidade de um modelo do mundo interno;
- em respeito ao determinismo, o ambiente é *estocástico*, dado que seu próximo estado não pode ser determinado do ponto de vista de um agente, a partir do estado atual;
- com relação ao dinamismo, o ambiente em questão é *estático*, já que este é mutável somente sob a ação de algum agente;
- quanto à continuidade, o ambiente é *discreto* por se tratar de uma matriz bem delimitada com 700 mil posições para alocação de agentes.

Sobre a interindependência dos agentes-pessoa, durante sua implementação, o recurso ideal

a ser adotado é vê-los como processos independentes, os quais disputam recursos (tanto computacionais quanto recursos do ambiente) que precisam ser acessados concorrentemente. Como são muitas as linguagens que não proporcionam a possibilidade de implementar tal propriedade, uma maneira aplicável de emular tal comportamento é sincronizar a atuação dos agentes-pessoa da seguinte maneira: no ciclo das etapas (1) caminhar, (2) verificar vizinhança e (3) mudar (ou não) estado interno, um agente só deve executar uma das etapas depois que todos os demais terminaram sua execução individual da etapa anterior. Portanto, para linguagens de programação que não possuem a capacidade de implementar algoritmos com múltiplas linhas de execução (não é o caso da implementação gerada por este trabalho, como pode-se verificar no capítulo a seguir), cada um destes passos deve ser dado assim que todos os agentes do ambiente tiverem terminado de executar o passo anterior.

Capítulo 6

A Implementação do Simulador

De posse da modelagem do sistema multiagentes para simulação epidemiológica apresentado no capítulo anterior uma implementação pôde então ser feita. Esta é o objeto de estudo deste capítulo.

Todavia, antes de abordar de fato a implementação, faz-se necessário apresentar e justificar algumas escolhas tecnológicas para tal atividade, como o paradigma e a linguagem de programação adotados.

Um agente pode ser entendido como uma extensão de um objeto, que ganha características adicionais como autonomia, a interatividade e a adaptabilidade. Deste modo, tal informação sugere que o **paradigma de programação orientado a objetos** seja, dentre vários paradigmas existentes, o mais adequado para implementar um sistema multiagentes [26].

Pode-se ainda dizer que a orientação a objetos corresponde a uma tentativa de modelagem do mundo real para sistemas de computação em que as entidades são representadas por invólucros de código independentes chamados *objetos*. Estes objetos descritos através de seus *atributos*, que armazenam as principais características de uma determinada aplicação, e de seus métodos, que dizem respeito ao comportamento esperado para o objeto, implementando o que ele é capaz de fazer, podem apresentar semelhanças em maior ou menor nível de abstração, sendo portanto passíveis de classificação [12].

Da classificação de objetos surgem as *classes*, que se relacionam entre si através de conceitos como a *generalização*, a *especialização*, a *composição* e a *agregação*. Por exemplo, enquanto a generalização compreende o processo de identificar similaridades entre dois tipos de atributos e métodos de duas ou mais classes - dando origem a uma superclasse ou classe pai - a especialização evidencia as diferenças, produzindo novas subclasses ou classes filhas. Quanto

à composição e agregação por sua vez, traz a idéia de que um objeto de uma determinada classe faça parte de outro como um de seus atributos [12].

Sobre a linguagem de programação utilizada, dada a decisão pelo paradigma orientado a objetos, a escolha feita foi a favor da **plataforma Java** que é composta por uma linguagem de programação associada a uma máquina virtual¹ capaz de interpretá-la.

Dentre as várias vantagens que tal plataforma oferece, as mais evidentes são o fato de sua linguagem de programação ser totalmente orientada a objetos, além de outras propriedades como a portabilidade, já que existem máquinas virtuais Java para as mais variadas plataformas de sistemas operacionais, e a grande disponibilidade de ferramentas gratuitas de suporte ao desenvolvimento de código, como o NetBeans IDE² (também utilizado para desenvolvimento da ferramenta de simulação epidemiológica).

Devido ao grande número de informações geradas pela ferramenta desenvolvida (número de indivíduos em cada um dos compartimentos da epidemia simulada a cada passo do sistema) foram desenvolvidos dois sistemas de saída de dados: documento de texto com os valores dos resultados obtidos e gráfico plotado destes mesmos valores.

No documento de texto estão listados os valores correspondentes a dias, número de indivíduos suscetíveis, número de indivíduos latentes (se houver latência no modelo compartimental da doença simulada), número de indivíduos infectados e número de indivíduos imunes (se houver imunidade no modelo compartimental da epidemia simulada). Na Figura 6.1 pode-se observar o conteúdo de um documento de texto contendo o resultado de uma simulação do modelo compartimental SEIR, com 3 dias de infecção e 3 dias de latência, durante 10 dias, em uma população de 2000 indivíduos onde 3 destes indivíduos se encontram infectados no início da simulação.

Para geração dos gráficos contendo os resultados das simulação foi escolhida a Google Charts API (*Application Programming Interface*)³, que permite gerar gráficos dinamicamente, tendo como único requisito a necessidade de uma conexão com a internet.

A API do Google Chart retorna uma imagem em formato PNG (*Portable Network*

¹A JVM (*Java Virtual Machine*) ou em português Máquina Virtual Java, é um mecanismo que permite executar códigos Java em diversas plataformas. Mais informações podem ser obtidas em <<http://java.sun.com/docs/books/jvms/>>

²A ferramenta NetBeans IDE pode ser obtida gratuitamente no endereço <<http://www.netbeans.org>>

³O Guia do Desenvolvedor da API do Google Chart está disponível no endereço <<http://code.google.com/apis/chart/>>

```

0 1993 4 3 0
1 1991 6 3 0
2 1989 8 3 0
3 1985 8 4 3
4 1980 11 6 3
5 1976 13 8 3
6 1969 16 8 7
7 1966 14 11 9
8 1954 22 13 11
9 1946 23 16 15
10 1938 28 14 20

```

Figura 6.1: Conteúdo do documento de texto da simulação de uma epidemia SEIR. A primeira coluna refere-se ao dia do sistema, a segunda coluna refere-se ao número de indivíduos suscetíveis em cada dia do sistema. Na terceira coluna pode-se ver o número de indivíduos latentes, na quarta coluna o número de indivíduos infectados e na quinta coluna o número de indivíduos no compartimento dos imunes.

Graphics)⁴ em resposta a uma URL (*Universal Resource Identifier*)⁵. É possível gerar diversos tipos de imagens, incluindo gráficos de linha, de barras e de setores, simplesmente especificando atributos como tamanho, cores e rótulos para cada tipo de imagem. Na Figura 6.2 pode-se ver o gráfico gerado para uma simulação do modelo compartimental SEIR, com 3 dias de infecção e 3 dias de latência, durante 100 dias, em uma população de 2000 indivíduos onde 3 destes indivíduos se encontram infectados no início da simulação (mesma doença simulada na representação da Figura 6.1).

Apresentadas as escolhas tecnológicas, a implementação pode então ser apresentada. De modo a organizar o código produzido, as classes foram geradas em termos mais gerais, unidas em dois pacotes, a saber: o **pacote componentes**, que contém as classes dos objetos agente-pessoa e das barreiras e o **pacote view**, que contém a implementação das interfaces gráficas do sistema, incluindo a classe Ambiente, que foi implementada estendendo a classe JPanel⁶. A Figura 6.3 mostra o diagrama de pacotes (e suas classes) para o aplicativo desenvolvido. Nas seções a seguir, cada um destes pacotes é melhor explicado.

O conceito de *pacote* é próprio da programação orientada a objetos e diz respeito a um conjunto de *classes* que são agrupadas segundo algum critério específico. A definição de *diagrama de pacotes* faz parte da linguagem UML (*Unified Modeling Language*). Dada a popu-

⁴Formato de dados livre utilizado para imagens.

⁵Sintaxe do sistema de endereçamento de uma rede.

⁶Um tipo de *Container* que define uma área de composição de componentes contida em alguma janela. Mais informações sobre a classe JPanel podem ser encontradas no endereço <<http://java.sun.com/javase/7/docs/api/javax/swing/JPanel.html>>

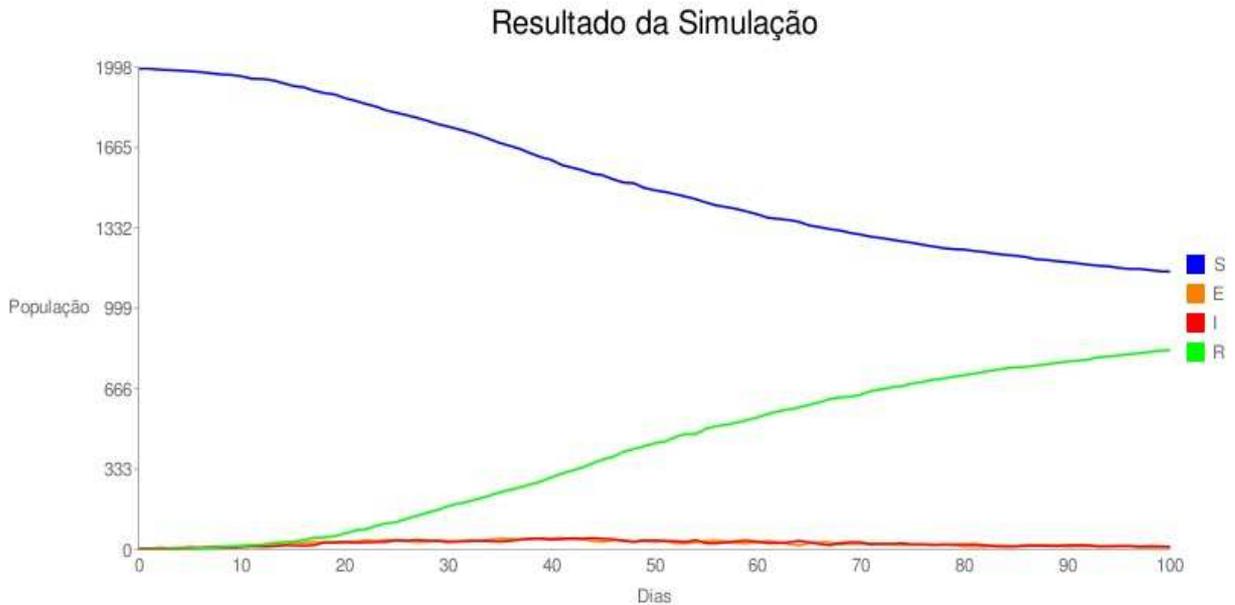


Figura 6.2: Gráfico gerado pela API Google Chart referente à simulação de uma epidemia SEIR

laridade alcançada pela tecnologia orientada a objetos (OO), diversos métodos para representar projetos OO foram criados nos anos oitenta e noventa, resultando em uma miscelânea de diagramas e notações [12]. Em uma tentativa de unificar os métodos criados, Booch, Jacobson e Rumbaugh desenvolveram a UML⁷, que conta com vários diagramas para representar os aspectos estruturais, comportamentais e funcionais de sistemas OO. Desta forma, um diagrama de pacotes expressa os pacotes que coexistem em um nível de abstração específico para um dado sistema [7].

6.1 O pacote componentes

Como dito anteriormente, o pacote componentes contém as classes dos agentes (que populam de fato o sistema multiagente) e das barreiras. Estas classes compõem⁸ a classe Ambiente. As seguintes subseções explicam detalhadamente estas duas classes.

⁷A documentação da última versão da UML pode ser obtida em <http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML>

⁸A composição é um relacionamento entre duas classes onde uma das classes representa o todo (Ambiente) e a outra representa os elementos que compõem o todo (Barreiras e Agentes), onde as partes só podem existir com o todo, e são criadas e destruídas com ele [7].

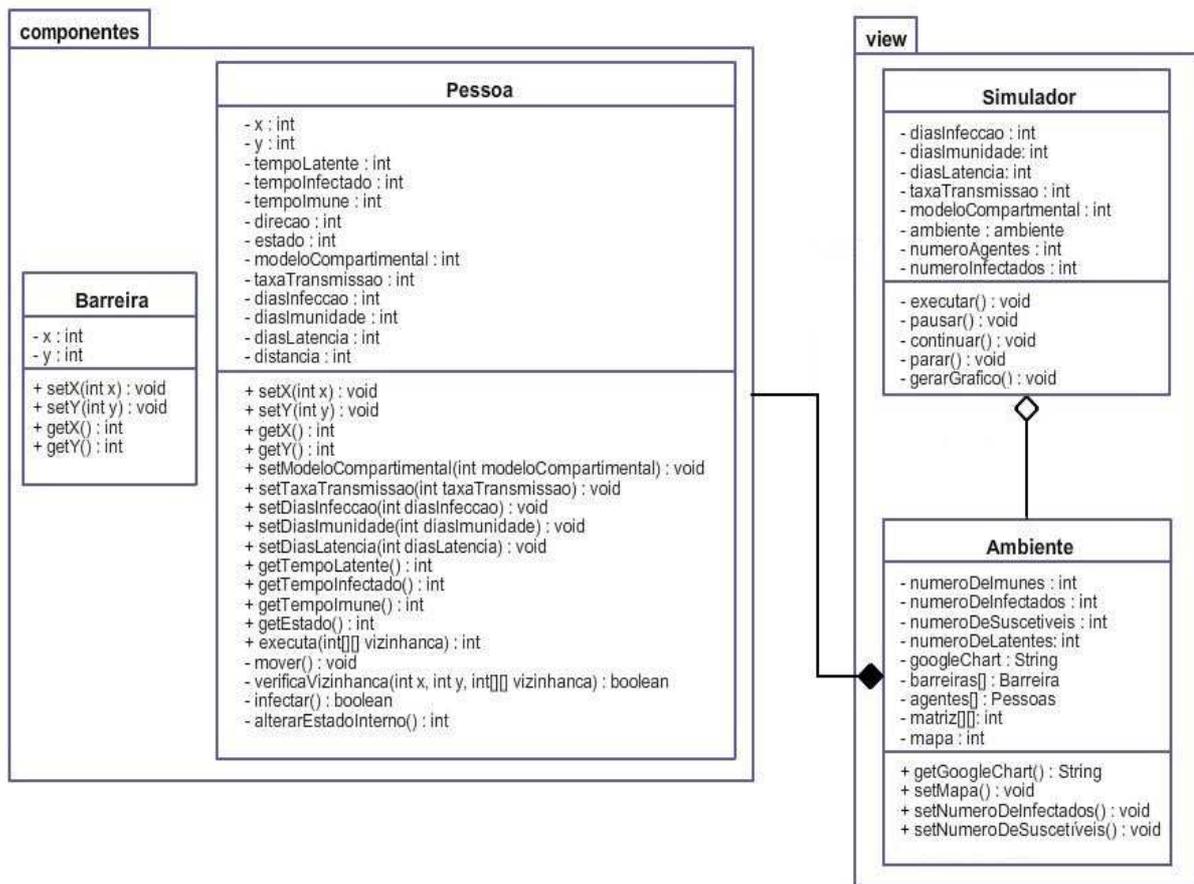


Figura 6.3: Diagrama de pacotes mais geral do aplicativo desenvolvido

6.1.1 Classe Barreira

Como dito na Seção 5.2.2, a única função das barreiras é bloquear a passagem dos agentes pessoa, e para cumprir com este objetivo ele se utiliza da sua simples existência, pois um agente-pessoa não pode ocupar uma célula da matriz-ambiente que já está ocupada por outro agente-pessoa ou por uma barreira.

Para esta finalidade foi implementada a classe Barreira possuindo dois atributos inteiros x e y , referentes à sua localização na matriz-ambiente, e quatro operações, referentes aos seus *getters* e *setters*⁹. Estes poucos atributos e métodos são suficientes para alcançar os objetivos da classe.

⁹São métodos seletores (*getters*) e modificadores (*setters*) dos atributos de sua classe. Através deles aplica-se uma regra da OO chamada *encapsulamento*, que consiste em evitar o acesso aos atributos de uma classe diretamente.

6.1.2 Classe Pessoa

Implementada extendendo a classe *Thread*¹⁰, representa o agente-pessoa, modelado na Seção 5.2.1. Os objetos da classe Pessoa, além dos atributos inteiros x e y , referentes à sua localização na matriz-ambiente, possuem outros 11 (onze) atributos, detalhados na Tabela 6.1.

Tabela 6.1: Atributos da classe Pessoa

Atributo	Descrição
tempoLatente	Número inteiro que funciona como um contador que recebe o número de dias de latência (diasLatencia) que a doença que está sendo simulada tem, e é decrementado a cada dia do sistema (um dia é igual a 700 iterações, em cada iteração cada agente do sistema se movimenta por uma posição). Ao chegar ao valor 0 (zero), o estado do agente é alterado para infectado.
tempoInfectado	Número inteiro que funciona como um contador que recebe o número de dias de infecção (diasInfeccao) que a doença que está sendo simulada tem, e é decrementado a cada dia do sistema. Ao chegar ao valor 0 (zero), o estado do agente é alterado.
tempoImune	Número inteiro que funciona como um contador que recebe o número de dias de imunidade parcial (diasImunidade) que a doença que está sendo simulada tem, e é decrementado a cada dia do sistema. Ao chegar ao valor 0 (zero), o estado do agente se torna suscetível.
direcao	Número inteiro de 0 a 8. Indica a direção em que o agente irá se movimentar. Onde: 0 significa que o agente irá caminhar para o norte , mantendo o mesmo valor em x e decrementando o valor de y ; 1 significa nordeste , incrementando o x e decrementando o y ; 2 significa leste , incrementando o x e mantendo o mesmo valor em y ; 3 significa sudeste , incrementando o x e o y ; 4 significa sul , mantendo o mesmo valor em x e incrementando o Y ; 5 significa sudoeste , decrementando o x e incrementando o Y ; 6 significa oeste , decrementando o x e mantendo o mesmo valor em Y ; 7 significa noroeste , decrementando o x e o Y ; e 8 significa que o agente irá manter-se na mesma posição . A posição do ambiente onde ambos x e y são iguais a 0 (zero) é o ponto mais a noroeste no mapa.

¹⁰Uma *thread* é uma linha de execução em um programa. A JVM permite que uma aplicação use múltiplas linhas de execução ao mesmo tempo, concorrentemente. Mais informações sobre a classe *Thread* podem ser obtidas no guia do desenvolvedor Java, disponível em <<http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Thread.html>>.

estado	Representa o compartimento do modelo compartimental da epidemia simulada em que o agente-pessoa objeto desta classe se encontra, onde: 0 significa que o agente se encontra no estado suscetível ; 1 representa o estado de latência ; 2 representa o estado infectado ; e 3 significa que o agente está no estado de imunidade .
modeloCompartimental	Representa o modelo compartimental a ser simulado. Foram selecionados oito modelos compartimentais para serem simulados nesta ferramenta. A seguir são apresentados um número inteiro representando um valor que pode ser utilizado na variável <i>modeloCompartimental</i> e o modelo compartimental de epidemia representado por ele: 1 , SI; 2 , SEI; 3 , SIS; 4 , SEIS; 5 , SIR; 6 , SEIR; 7 , SIRS; e 8 , SEIRS.
taxaTransmissao	Referente à taxa de transmissão da doença que está sendo simulada. É um número inteiro de 0 a 100 onde 0 significa que não há transmissão da doença e 100 significa que em qualquer contato adequado¹¹ a doença será transmitida . Em outras palavras, é a probabilidade de transmissão de uma doença em um contato adequado.
diasInfeccao	Diz respeito ao número de dias do período infeccioso da doença simulada.
diasImunidade	Referente ao número de dias de imunidade parcial conferida pela doença simulada.
diasLatencia	Representa o número de dias do período de latência da epidemia simulada.
distancia	Diz respeito ao número de posições que o agente vai avançar na direção escolhida, sendo o limite máximo 250 posições.

Como dito na Seção 5.2.1 um agente-pessoa pode atuar de duas maneiras no ambiente em que habita: modificando sua posição atual e modificando seu estado interno. Para isso a classe Pessoa se utiliza de quatro métodos: o método **mover** age modificando a posição atual do agente; o método **verificaVizinhanca** que verifica a necessidade de alteração na rota e retorna **true** se existir algum indivíduo infectado em sua vizinhança; o método **infectar** pode modificar o estado do agente de suscetível para os estado infectado ou latente (dependendo do modelo compartimental da doença simulada); e o método **alterarEstadoInterno** que decremente o tempo de estadia em um estado a cada iteração, e altera o estado do agente caso o tempo de determinado estado tenha acabado. Estes métodos são descritos nas sub-seções a seguir.

A classe Pessoa faz a chamada a cada um destes métodos através de sua função **executa**,

¹¹O conceito de contato adequado é dado na Seção 2.3.1.

que é chamada a cada iteração do sistema. A linha de execução desta função pode ser vista no fluxograma presente na Figura 6.4.

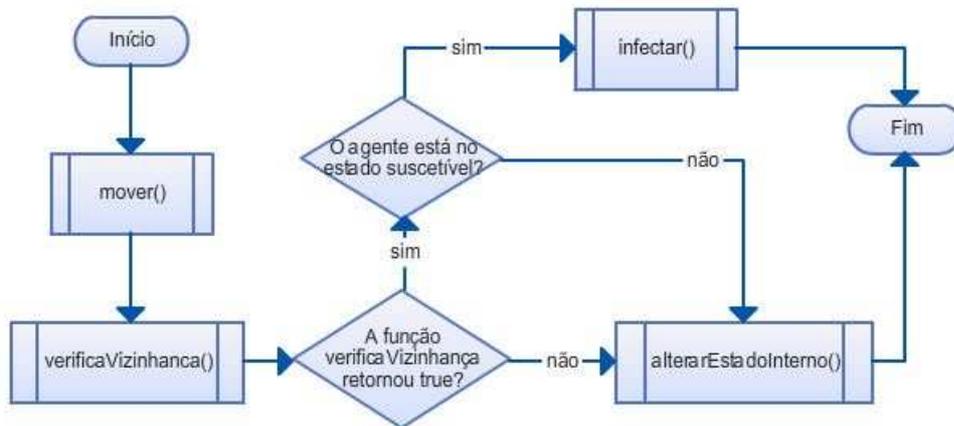


Figura 6.4: Fluxograma da função principal da classe Pessoa

É dita uma **iteração do sistema** o tempo suficiente para que todos os agentes-pessoa que populam o ambiente possam efetuar a função *executar* por completo.

Método mover

Este método é reponsável pela ação de agente de *alteração de posição atual*. Para isso, age alterando a posição do agente na matriz-ambiente. O fluxograma presente na Figura 6.5 apresenta o funcionamento desta função em alto nível.



Figura 6.5: Fluxograma da função mover da classe Pessoa

Método verificaVizinhanca

Este método verifica se existe algum indivíduo no caminho do agente, setando a variável *distancia* para 0 caso positivo, para que na próxima execução da função *executa* o agente-pessoa mude sua rota. Além disso, verifica se existe algum indivíduo infectado em sua vizinhança, retornando *true* se houver. A Figura 6.6 mostra o funcionamento deste método.

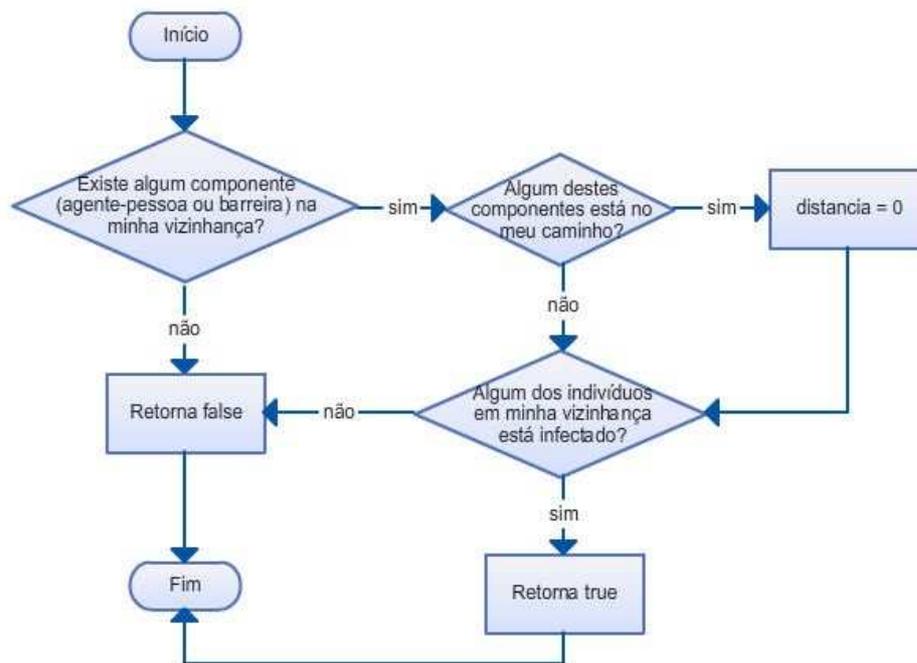


Figura 6.6: Fluxograma da função *verificaVizinhanca* da classe Pessoa

Método infectar

Este método é chamado caso o agente esteja no estado suscetível e exista em sua vizinhança algum indivíduo infectado. Nele, é verificada a taxa de transmissão da doença simulada, sorteando um número aleatório de 0 a 100 e verificando se este valor é menor do que a taxa de transmissão. Caso positivo, o agente será infectado pela epidemia, podendo passar para o estado latente ou indo diretamente para o estado infeccioso, isso depende do modelo compartimental da doença. A Figura 6.7 apresenta o fluxograma do funcionamento desta função.

Ao alterar o seu estado interno, o agente deve alimentar suas variáveis que controlam o tempo de permanência no novo estado. Estas variáveis são *tempoLatente*, *tempoInfectado* e *tempoImune*, e foram detalhadas na Tabela 6.1.

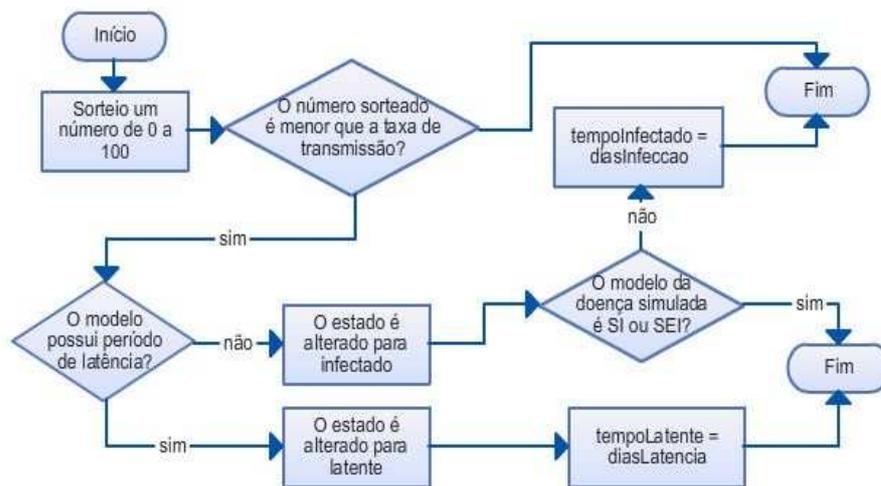


Figura 6.7: Fluxograma da função *infectar* da classe Pessoa

Método alterarEstadoInterno

Na Figura 6.8 pode-se ver o fluxograma do funcionamento desta função, que é acionada para verificar se o tempo de estadia em algum estado terminou, e alterar o estado do agente neste caso.

Nos modelos compartimentais implementados que possuem período de latência (SEI, SEIS, SEIR, SEIRS), o agente permanece no estado *latente* por uma certa quantidade de dias¹², ao fim desta o agente é movido para a classe dos infectados.

Uma vez na classe I, o agente pode: permanecer neste estado pelo resto da simulação, nos modelos SI e SEI; ser movido para o compartimento dos indivíduos imunes, nos modelos SIR, SEIR, SIRS e SEIRS; ou ter seu estado alterado para suscetível, nos modelos SIS e SEIS.

Nos modelos SIR e SEIR o indivíduo permanece na classe dos imunes até o fim da simulação, e nos modelos SIRS e SEIRS a imunidade conferida pela doença é temporária, e o indivíduo será transferido para a classe dos suscetíveis assim que este tempo terminar.

O indivíduo somente é removido do compartimento dos suscetíveis se este contrair a epidemia, e esta ocorrência é assegurada pelo método *infectar* apresentado anteriormente.

¹²Como dito anteriormente, um dia é representado por 700 iterações do sistema.

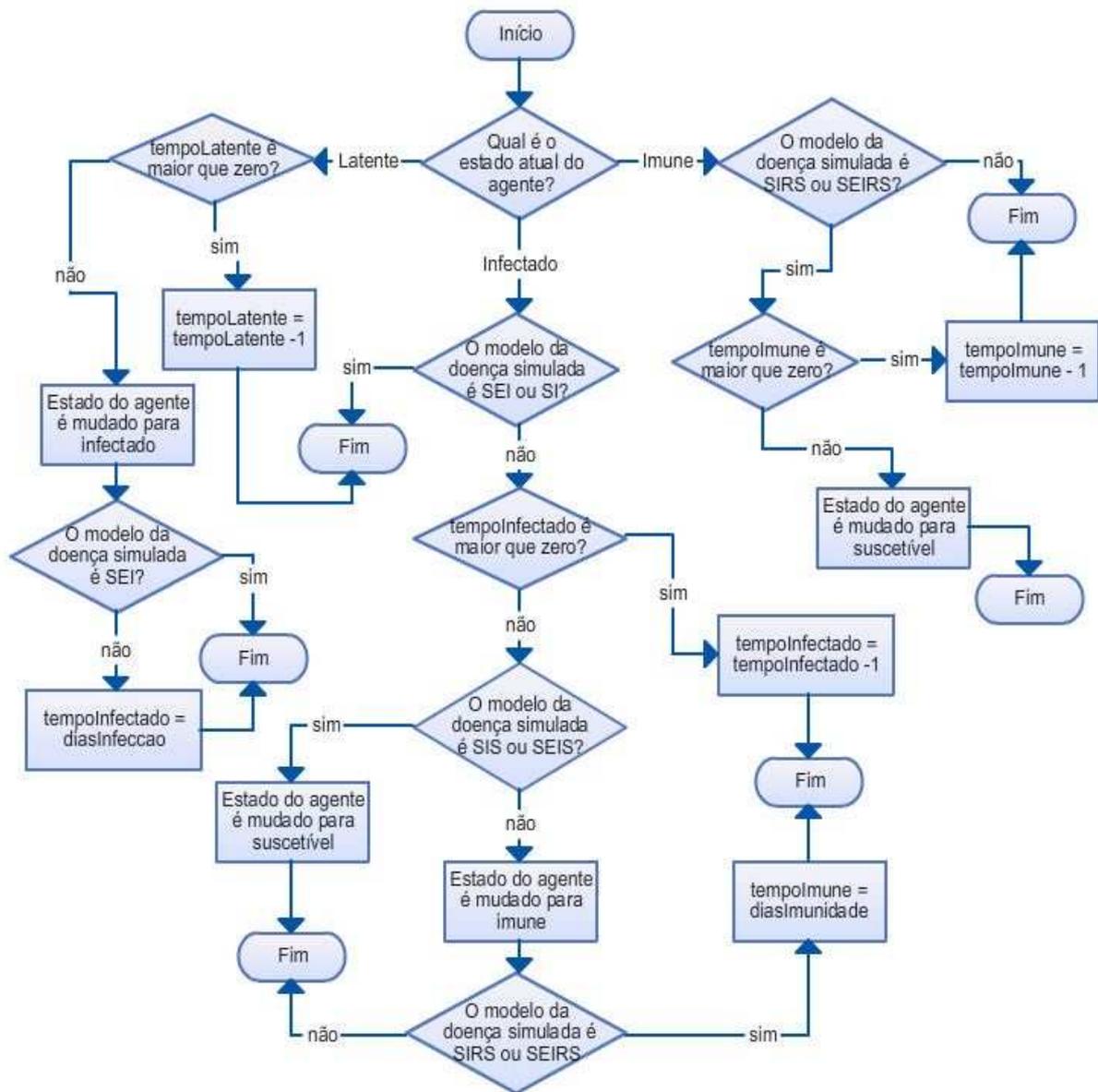


Figura 6.8: Fluxograma da função *alterarEstadoInterno* da classe *Pessoa*

6.2 O pacote *view*

Terminada a apresentação do pacote componentes, que explicita a implementação dos componentes que populam o ambiente do sistema multiagentes modelado no capítulo anterior, pode-se então abordar o pacote *view* que, como comentado anteriormente, contém as classes responsáveis pela implementação das interfaces gráficas da ferramenta desenvolvida.

Fazem parte da implementação da plataforma *Java* pacotes específicos que trazem prontas

classes de objetos projetados especialmente para a composição de interfaces gráficas. Destes pacotes, o mais utilizado é o *javax.swing*, que com o passar do tempo tornou-se um padrão para a construção de telas de usuários em aplicações *Java* [12]. Desta forma, todos os componentes gráficos que constituem as telas apresentadas na sequência foram utilizados do pacote *javax.swing* supracitado.

Para reuni-las, usou-se a ferramenta de desenvolvimento NetBeans IDE¹³, que permite a construção de interfaces gráficas no estilo WYSIWYG¹⁴, reduzindo o tempo necessário para montagem de telas do um aplicativo.

A Figura 6.9¹⁵ mostra a tela principal da ferramenta desenvolvida. A grosso modo, esta é composta da tela de representação do ambiente de agentes (à esquerda) e uma área de configuração da simulação (à esquerda). A área de configuração da ferramenta apresenta as opções descritas na Tabela 6.2.

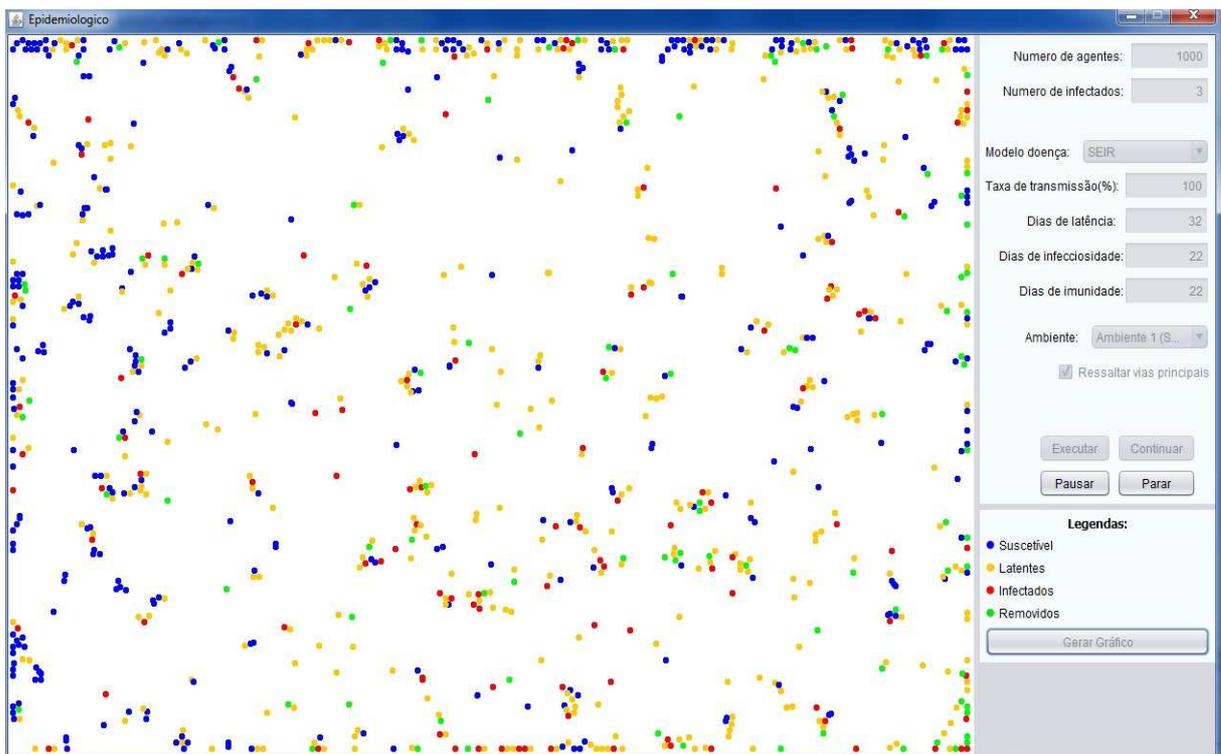


Figura 6.9: Tela principal da ferramenta desenvolvida

¹³A versão da ferramenta de desenvolvimento em *Java NetBeans IDE* utilizada foi a 6.7.

¹⁴WYSIWYG, do inglês *What You See Is What You Get*, diz respeito a uma categoria de ferramentas computacionais de edição e programação que usam recursos gráficos visuais para geração automática de código.

¹⁵A simulação apresentada na figura utiliza uma configuração que não representa uma doença real.

Tabela 6.2: Opções de configuração da ferramenta de simulação epidemiológica

Opção	Descrição
Número de agentes	Neste campo de texto deve-se indicar o número de agentes-pessoa total da sociedade de agentes onde a doença será inserida. Na simulação apresentada na Figura 6.9, foram utilizados 1000 (mil) agentes-pessoa.
Número de infectados	Referente ao número de infectados inseridos no início da simulação. Na simulação da Figura 6.9 foram inseridos 3 (três) indivíduos infectados em uma sociedade de 1000 (mil) agentes, logo, no princípio da simulação existiam 997 (novecentos e noventa e sete) agentes suscetíveis e 3 (três) agente infectados.
Modelo da doença	Referente ao modelo compartimental da infecção a ser simulada. Os modelos implementados foram SI, SIS, SIR, SIRS, SEI, SEIS, SEIR, SEIRS. Na simulação apresentada na Figura 6.9 usou-se o modelo SEIR.
Taxa de transmissão	Diz respeito à probabilidade da doença ser transmitida em um contato adequado. Como pode-se observar na Figura 6.9, nesta simulação usou-se 100% (cem por cento), o que significa que em todo contato adequado entre um indivíduo suscetível e um indivíduo infectado ocorrerá a transmissão do vírus.
Dias de latência	Referente ao número de dias de latência a doença a ser simulada apresenta. Se o modelo compartimental da doença não apresenta período de latência, esse valor é ignorado durante a simulação. Como pode-se ver na Figura 6.9, utilizou-se 32 (trinta e dois) dias de latência para esta simulação.
Dias de infecciosidade	Representa o número de dias de infecciosidade a doença a ser simulada possui. Na simulação da Figura 6.9, utilizou-se o valor hipotético de 22 (vinte e dois) dias de infecciosidade.
Dias de imunidade	Diz respeito ao número de dias de imunidade conferida pela doença, caso o modelo compartimental possua imunidade temporária, caso contrário, esse valor é ignorado. Na simulação apresentada na Figura 6.9 o modelo compartimental da doença simulada (SEIR) não conferia imunidade temporária.

Ambientes	Esta ferramenta apresenta 7 diferentes ambientes de agentes (diferenciados unicamente pela distribuição das barreiras). O primeiro ambiente (utilizado na simulação da Figura 6.9) não possui barreiras. O ambiente 2 foi construído com 9 comunidades com alta conectividade. O ambiente 3 também possui 9 comunidades, porém apresenta conectividade média. O ambiente 4 apresenta as mesmas comunidades com baixa conectividade e o ambiente 5 possui 9 comunidades sem conectividade. O ambiente 6 possui 5 comunidades, onde estas possuem diferentes níveis de conectividade entre si. E, por fim, o ambiente 7, onde as comunidades representam os bairros da cidade de Cascavel do estado do Paraná e a conectividade se dá de uma forma onde a comunicação entre os bairros periféricos é menor entre si, e vai aumentando conforme a comunidade esteja posicionada mais ao centro do município. Os ambientes 2, 3, 4, 5, 6 e 7 podem ser vistos na Figura 6.10.
Ressaltar vias principais	Este <i>checkbox</i> é utilizado para usar a opção de ressaltar ou não as vias principais do mapa quando o usuário estiver simulando uma epidemia no ambiente que representa o município da cidade de Cascavel, no Paraná.
Botões de controle	O botão <i>Executar</i> dá início à simulação. O botão <i>Pausar</i> pára a simulação e o botão <i>Continuar</i> dá continuidade. O botão <i>Parar</i> dá fim à simulação.
Gerar gráfico	Utilizando a <i>API Google Charts</i> , este botão gera uma URL que apresenta um gráfico referente à uma simulação previamente executada.

6.2.1 A classe *Ambiente*

Sobre a área do ambiente de agentes, que é representado por uma matriz (como previsto na Seção 5.3), utilizou-se o componente *JPanel*, que define uma área de composição de componentes contida em alguma janela.

Este componente aliado aos métodos da classe *Graphics*¹⁶ permite a representação da matriz em forma de imagem, onde cada pixel do ambiente representa uma posição da matriz.

Além disso a classe *Ambiente* possui nove atributos. Uma descrição mais detalhada destes pode ser vista na Tabela 6.3.

¹⁶Esta classe fornece uma abstração que permite que a aplicação possa apresentar formatos dinâmicos em seus componentes (como um *JPanel*). Mais informações sobre esta classe podem ser obtidas em <<http://java.sun.com/j2se/1.4.2/docs/api/java/awt/Graphics.html>>.

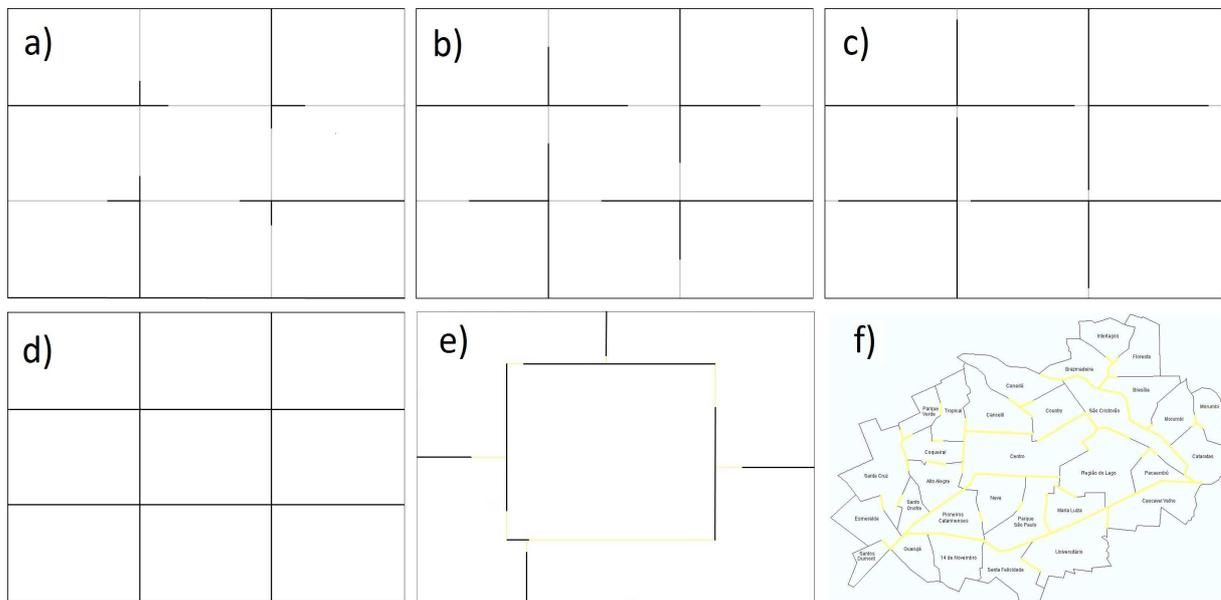


Figura 6.10: Ambientes desenvolvidos. a) nove comunidades altamente conectadas; b) nove comunidade com conectividade média; c) nove comunidade com baixa conectividade; d) nove comunidades sem conectividade; e) cinco comunidades com diferentes conexões entre si; f) ambiente representando os bairros do município de Cascavel.

Tabela 6.3: Atributos da classe *Ambiente*

Atributo	Descrição
numeroDeImunes	Contador que armazena o número de indivíduos imunes durante a simulação.
numeroDeInfectados	Contador que armazena o número de indivíduos infectados durante a simulação.
numeroDeSuscetíveis	Contador que armazena o número de indivíduos suscetíveis durante a simulação.
numeroDeLatentes	Contador que armazena o número de indivíduos latentes durante a simulação.
googleChart	<i>String</i> que armazena a URL do gráfico gerado pela simulação.
barreiras[]	Vetor que armazena as barreiras usadas no ambiente.
agentes[]	Vetor que armazena os agente-pessoas que populam o ambiente.
matriz[][]	Representação abstrata do ambiente. Nela são alocadas as barreiras e os agente através de números inteiros que os representam.
mapa	Número inteiro de 0 a 6. Indica qual o modelo do ambiente está sendo utilizado.

Dos seus métodos, os *setters* são: **setMapa**, que define o ambiente a ser utilizado na simulação; **setNumeroDeInfectados**, que informa à classe qual o número de indivíduos infectados no início da simulação; e **setNumeroDeSuscetiveis**, que informa o número de indivíduos suscetíveis existem no início da simulação, sendo este o número total de agentes subtraído do número de agentes infectados. O método **getGoogleChart** retorna uma *String* com a URL a ser acessada a fim de se obter o gráfico gerado na simulação.

6.2.2 A classe Simulador

Os atributos desta classe são dados obtidos através de sua interação com o usuário. Estes dados são referentes à epidemia a ser simulada. Seus métodos são, em sua maioria, funções de controle do ambiente de simulação: **executar** dá início à simulação; **pausar** congela a simulação; **continuar** retoma-a; e **parar** finaliza a simulação. Exceção à esta regra é o método **gerarGrafico** que chama a função *getGoogleChart* da classe *Ambiente* e dispara um evento para iniciar um navegador de *internet* que, se conectado em rede, acessa a URL obtida.

6.3 Considerações finais

Em termos gerais, a ferramenta cuja implementação foi apresentada neste capítulo, representa de fato uma concretização da modelagem proposta no Capítulo 5 para um sistema multiagentes para simulação de eventos epidemiológicos.

Deste modo, tendo em vista que o programa é funcional, a adoção do uso de agentes computacionais contribuiu bastante para este fato, dado que não somente as escolhas tecnológicas (de paradigma e de linguagem de programação) foram feitas ao se pensar no aplicativo como um sistema multiagentes, mas também o planejamento das classes e da maneira como seus objetos deveriam interagir de modo a mimetizar pessoas de uma sociedade foram influenciados por tal decisão.

Ainda assim, com a intenção de verificar se o programa é realmente um simulador válido para eventos epidemiológicos, experimentos devem ser feitos no sentido de verificar a qualidade de suas respostas, bem como o impacto dos parâmetros de configuração de doenças sobre a sua capacidade de resolver o problema para qual os algoritmos foram escritos.

Capítulo 7

Resultados Obtidos

De posse da **ferramenta para simulação de eventos epidemiológicos** que, como apresentado no capítulo anterior, é uma implementação fidedigna à modelagem descrita no Capítulo 5, análises de resultados podem ser feitas no sentido de:

- 1) Certificar-se que o **aplicativo produzido é de fato um simulador válido para eventos epidemiológicos**, e que **seus resultados são coerentes com os dados encontrados na literatura**.
- 2) Verificar a **influência dos parâmetros de configuração sobre a capacidade dos algoritmos propostos**.
- 3) Verificar a **o comportamento da ferramenta desenvolvida em simulações realizadas em ambientes divididos em comunidades**.

Nestes termos, de modo a atender a estes três objetivos gerais, foi necessário alcançar primeiramente diversos objetivos específicos. Durante os estudos desenvolvidos a fim de se atender à meta definida para este trabalho, foram gerados e publicados dois textos com resultados parciais.

O primeiro, um resumo estendido nomeado “**A Utilização de Agentes Computacionais na Modelagem não Determinística de Eventos Epidemiológicos**” [40] publicado no II ECA (Encontro em Computação Aplicada). O segundo, nomeado “**Simulações de Eventos Epidemiológicos Baseadas em Redes de Contatos e Agentes Computacionais**” [24] foi apresentado na Mostra de Trabalhos e Protótipos do IV C3N (Congresso da Academia Trinacional de

Ciências). Detalhes sobre estes trabalhos e os resultados obtidos são apresentados nas seções a seguir.

7.1 Resultados Parciais - ECA

Neste trabalho são apresentadas a interface inicial da ferramenta utilizando o ambiente que representa os bairros do município de Cascavel no Paraná (Figura 7.1), as regras mais básicas dos agentes gerados, e a simulação no contexto do modelo compartimental SIR, o primeiro implementado.

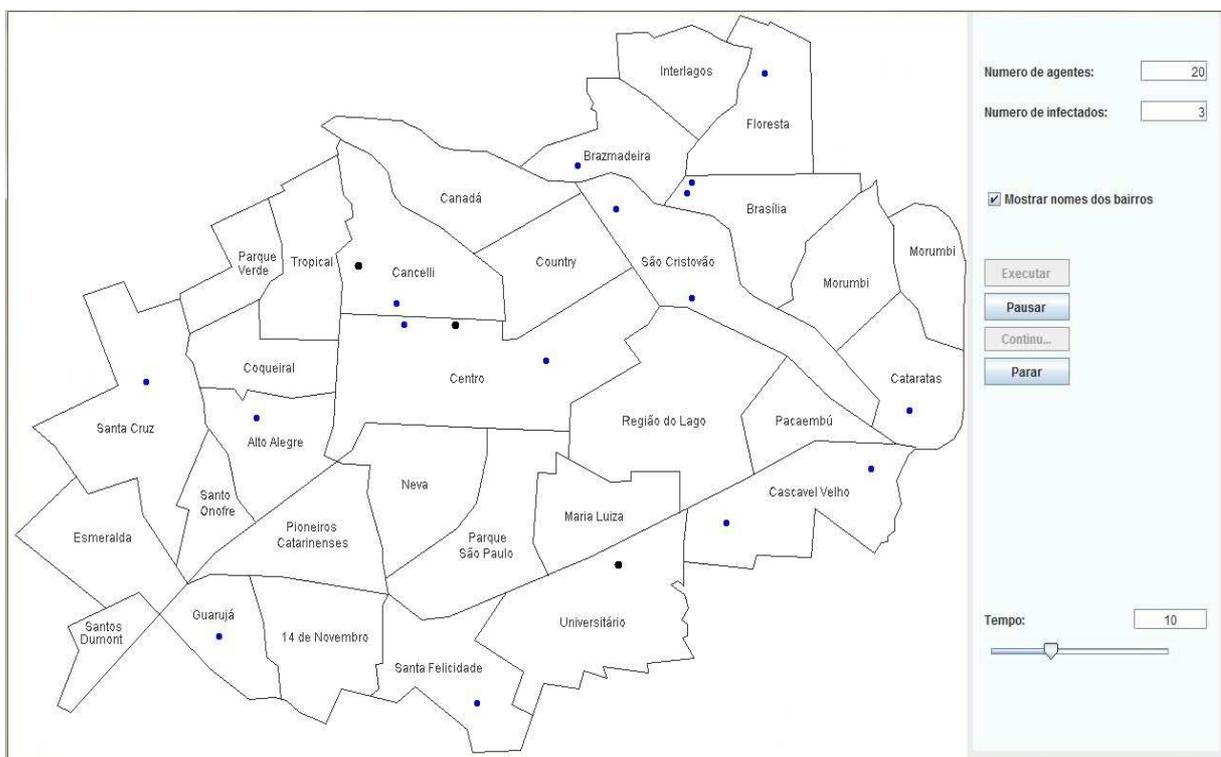


Figura 7.1: Tela inicial do experimento desenvolvido

Os experimentos realizados para este ambiente consideraram como domínio de teste a região cuja fronteira é o perímetro urbano da cidade de Cascavel, sem considerar a região rural. As vias de contato por onde a doença é disseminada, são representadas com pequenas aberturas, ditas vias de conexão, que interligam um bairro ao outro.

Apenas alguns testes foram realizados utilizando dados fictícios, de modo a testar o funcionamento do ambiente de simulação. Mas, mesmo que antecipadamente à fase de testes, pode-se

observar que o uso de agentes computacionais é uma técnica que pode ser aplicada para a simulação de surtos epidemiológicos com sucesso, pois pode simular de maneira conveniente os processos dos contatos entre os indivíduos.

7.2 Resultados Parciais - C3N

Este trabalho apresenta duas abordagens computacionais para simulação do modelo compartimental SEIR, redes de contato e agentes computacionais. Ambas as ferramentas foram testadas em ambientes similares, com 6500 indivíduos divididos em 5 comunidades, e os resultados obtidos foram comparados entre si e com a literatura.

Na modelagem baseada em redes de contato são empregadas as redes sem escala com estrutura de comunidades para representar a população, onde um indivíduo é representado por um vértice e seus contatos com outros indivíduos são as arestas do vértice.

A simulação executada usando a modelagem de agentes computacionais foi feita utilizando o ambiente que pode ser visto na Figura 7.2. Neste ambiente, uma população de 6500 indivíduos é aleatoriamente distribuída em cinco comunidades, onde a comunidade A ocupa 16% do espaço do ambiente, o compartimento B ocupa 18%, o C ocupa 12%, o D ocupa 22% e o compartimento E, que representa o centro da sociedade, ocupa a área referente a 32% da área total.

As taxas de mortalidade e natalidade são consideradas constantes, onde todos os recém-nascidos se tornam imediatamente suscetíveis, ignorando-se a imunidade proporcionada por anticorpos maternos, ou seja, não há transmissão vertical nem imunidade passiva.

A Figura 7.3 mostra os resultados obtidos nas simulações de uma doença fictícia, similar a uma gripe, com latência de 3 dias, período de infecciosidade de 4 dias, taxa de transmissão de 6%, e ao término do período de infecção confere imunidade temporária de 100 dias.

Em uma comunidade, a epidemia de gripe geralmente atinge um ponto máximo em 2 a 3 semanas após o surto inicial da doença, e dura entre 5 a 10 semanas [10]. Simulou-se um período de 60 dias, tempo médio da duração de um surto de gripe. Pode-se observar na figura que o pico de infecção ocorreu por volta do dia 24, o que é aproximado ao ciclo natural da doença. Além disso, pode-se observar que a curva de latência tem um ciclo que atinge o pico aproximadamente após 21 dias. A duração natural de um surto de gripe dura entre 35 e 70 dias. Os resultados da simulação realizada mostram que após 50 dias o surto encerrou-se ficando na

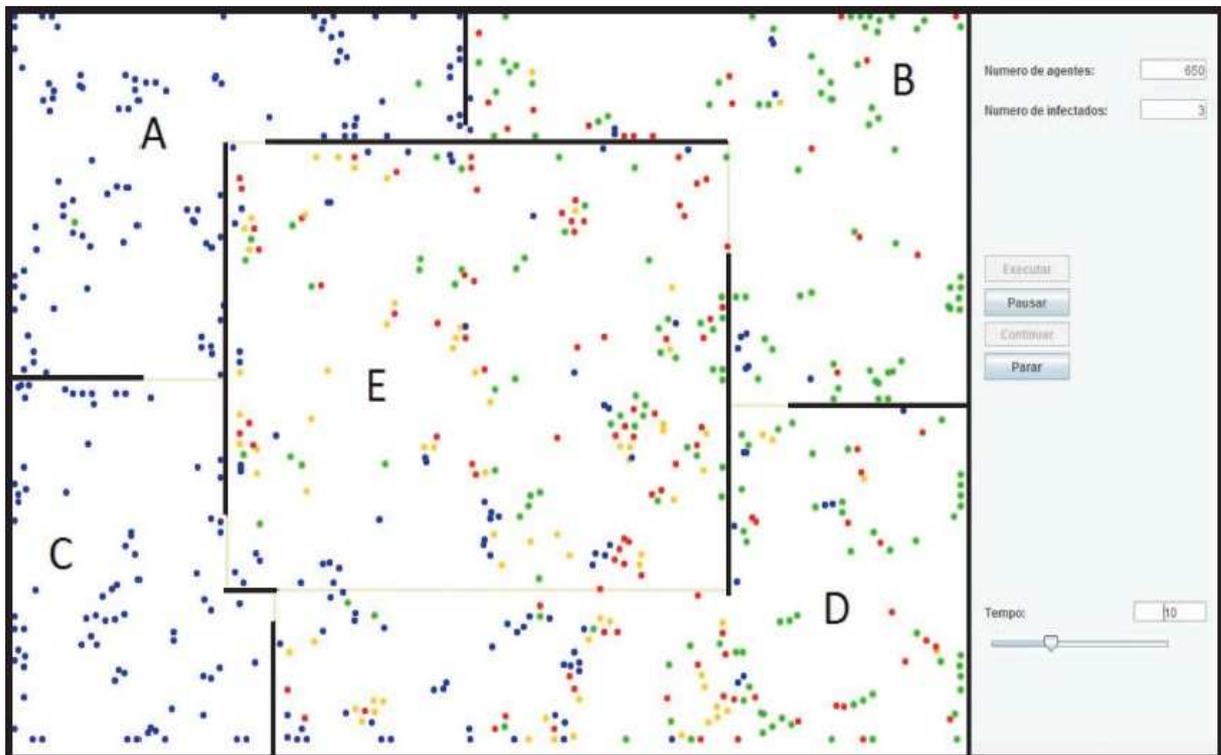


Figura 7.2: Tela da ferramenta parcial utilizando o ambiente dividido em cinco comunidades

média dos intervalos citados.

7.3 Resultados Finais

De modo a atender o primeiro objetivo, que visa verificar a validade das respostas produzidas pela ferramenta, na sequência, análises comparativas são estabelecidas entre os resultados obtidos com o aplicativo desenvolvido e algumas respostas de trabalhos relacionados descritos na literatura.

De forma semelhante, logo em seguida, com a intenção de atender o segundo objetivo deste trabalho que diz respeito à análise da influência dos parâmetros de configuração, descrições de experimentos feitos com o intuito de evidenciar a relação da variação da qualidade das respostas produzidas de acordo com a variação dos valores de configuração são devidamente realizadas.

Por fim, apresentam-se experimentos executados sobre o ambiente que mimetiza a divisão de bairros do município de Cascavel, Paraná, a fim de atender ao terceiro objetivo, que diz respeito à análise do comportamento da ferramenta desenvolvida em simulações realizadas em

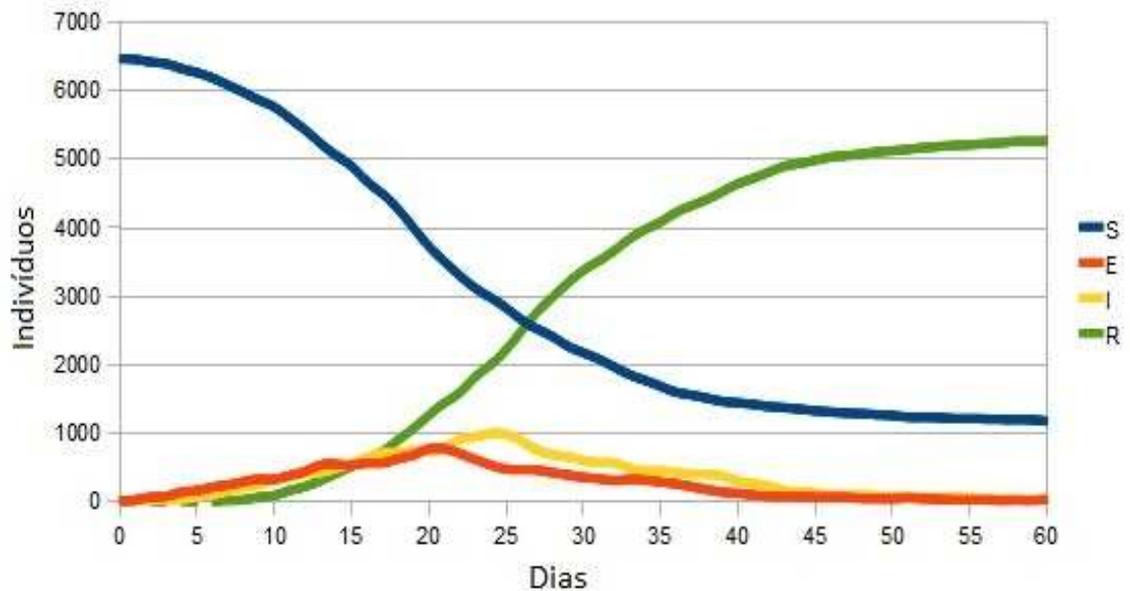


Figura 7.3: Resultado da simulação de uma doença do modelo compartimental SEIR usando agentes computacionais inseridos em um ambiente dividido em cinco comunidades

ambientes divididos em comunidades

7.3.1 Análises comparativas

Como fora comentado anteriormente, com o intuito de se certificar que o aplicativo desenvolvido é, de fato, um simulador válido dos modelos compartimentais propostos, comparações podem ser estabelecidas entre seus resultados e os resultados obtidos em outros trabalhos existentes na literatura, a exemplo de White em [56], simulando o modelo compartimental SIS utilizando autômatos celulares e Huang em [32], utilizando redes de mundo pequeno para verificar o espalhamento de uma epidemia SIR. Na sequência, o aplicativo desenvolvido é formalmente comparado com cada um dos demais trabalhos.

Análise comparativa com os experimentos de White

Em seu artigo intitulado *Modeling epidemics using cellular automata*, White, del Rey e Sánchez descrevem o desenvolvimento de uma ferramenta capaz de simular o modelo compartimental SIS através de autômatos celulares. Seus resultados podem ser comparados com as respostas produzidas pela aplicação desenvolvida neste trabalho para simulação de epidemias.

Utilizando uma população de vinte e cinco mil indivíduos em seus testes, White *et al* sim-

ularam uma epidemia fictícia com taxa de transmissão igual a 6%, em um autômato celular utilizando a vizinhança de Moore. O número de indivíduos infectados ao início da simulação é igual a 30 e a doença possui período de infecção de cerca de 3 dias. O resultado obtido por meio desta simulação pode ser visto na Figura 7.4.

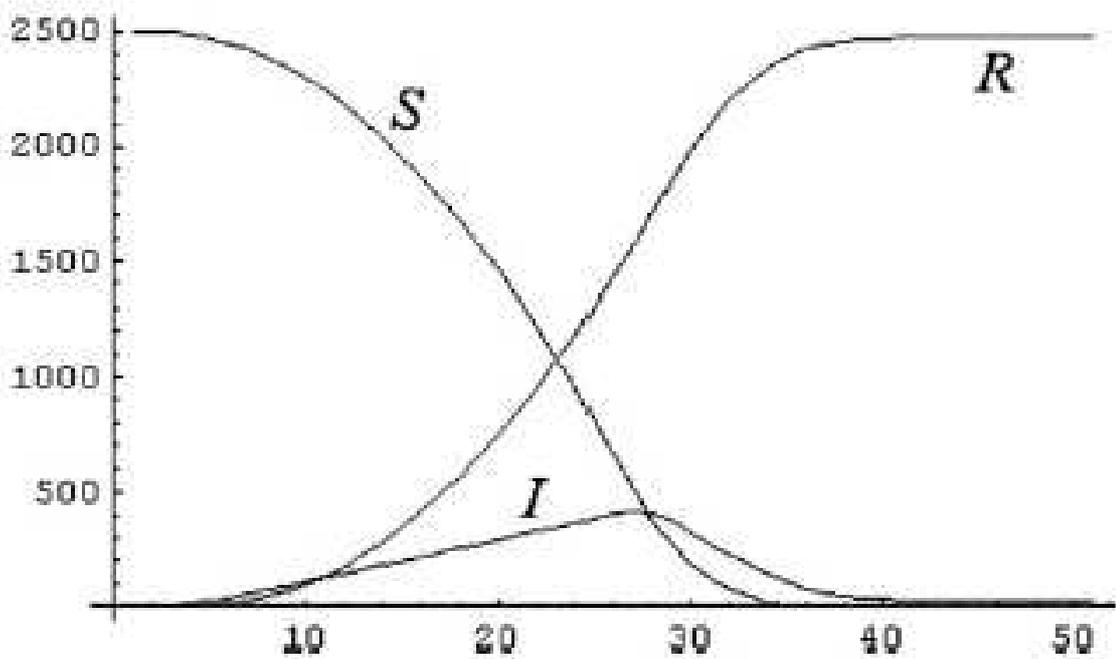


Figura 7.4: Resultados dos experimentos de White. Na linha vertical, o números de células da malha dos autômatos, onde cada célula representa 100 indivíduos. Na linha horizontal as iterações do sistema.

Uma simulação utilizando estes parâmetros foi executada na ferramenta desenvolvida. Como pode-se observar na Figura 7.5, os resultados foram bastante aproximados, o que quer dizer que, para esta simulação, estas ferramentas produzem resultados semelhantes.

Deste modo, considerando-se a fidelidade dos resultados dos aplicativo desenvolvido aos resultados obtidos por White, a ferramenta desenvolvida pode ser considerada, do ponto de vista dos experimentos de White, como válida.

Análise comparativa com os experimentos de Huang

O sistema desenvolvido por Huang *et al* utiliza redes complexas de mundo pequeno ¹. O autor, assim como White, modela doenças epidemiológicas de modelo compartimental SIR.

¹A definição de redes complexas de mundo pequeno foi dada na Seção 3.5.1.

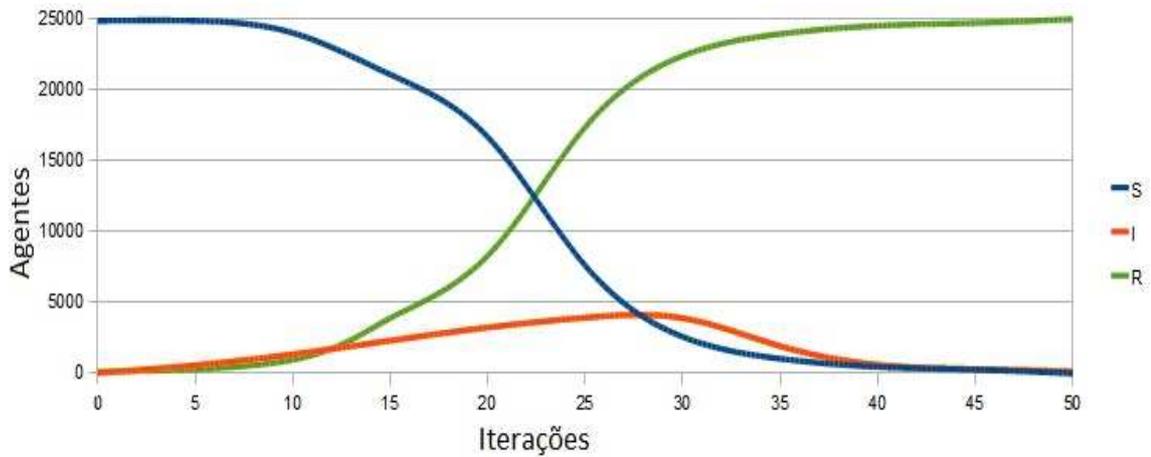


Figura 7.5: Gráfico gerado pela simulação utilizando os parâmetros de White

Entre outros experimentos, ele analisa o número acumulado de infectados durante a simulação. Para diferenciar do experimento anterior, decidiu-se comparar estes resultados de Huang com os resultados obtidos para este caso na ferramenta desenvolvida.

Para esta simulação, utilizou uma população de dez mil indivíduos em seus testes, simulado uma epidemia fictícia com taxa de transmissão igual a 3%, em uma rede de mundo pequeno altamente conectada, inserindo um indivíduo infectado no início da simulação com um dia de infecção. O resultado obtido por meio desta simulação pode ser visto na Figura 7.6.

Os resultados obtidos simulando as mesmas condições propostas por Huang podem ser vistos na Figura 7.7. Como pode-se observar, os resultados foram qualitativamente semelhantes.

De posse de tal comparação, pode-se considerar as respostas produzidas pela ferramenta desenvolvida como aceitáveis, dado que são similares aos obtidos no trabalho de Huang [32].

7.3.2 Análise da influência dos parâmetros de configuração

Terminadas as comparações dos resultados produzidos pelo programa desenvolvido com os resultados de outros trabalhos relacionados encontrados na literatura, uma análise do impacto dos parâmetros de configuração sobre a qualidade da resposta pode ser realizada.

O principal parâmetro de configuração que interfere na transmissão da epidemia e nos resultados da simulação é a *taxa de transmissão*. Para analisar este parâmetro, utilizou-se a simulação feita para comparação com os resultados de White (Figura 7.5), com taxa de transmissão de 6%.

Na simulação representada no gráfico da Figura 7.8, utilizou-se as mesmas configurações

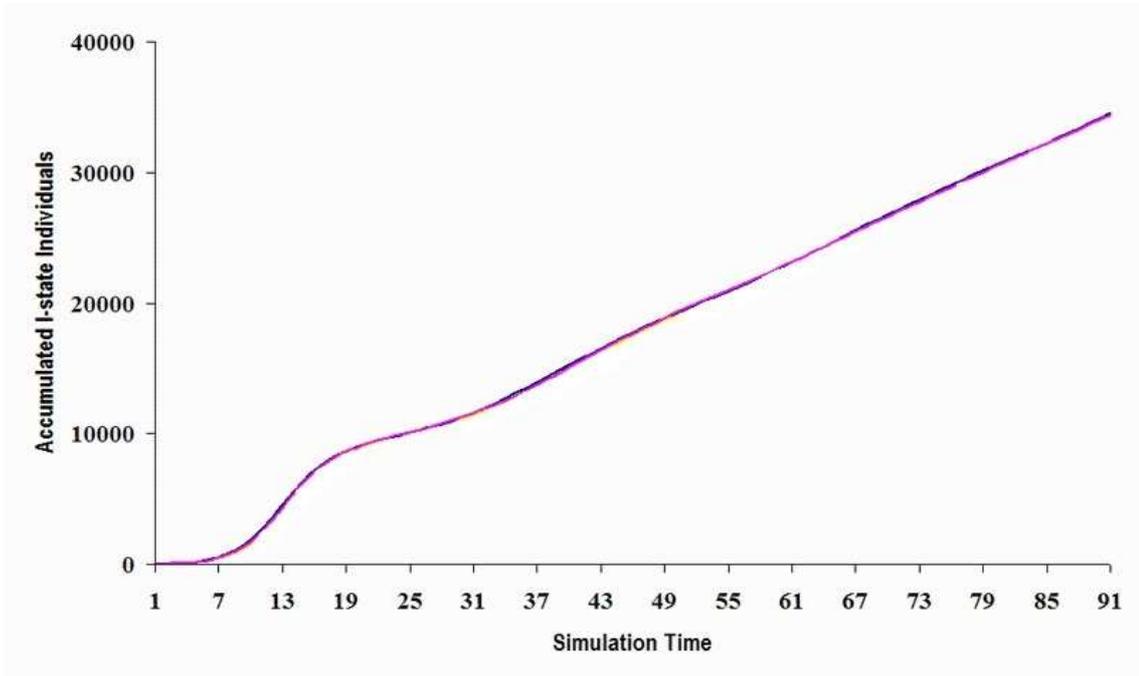


Figura 7.6: Número acumulado de indivíduos infectados segundo o experimento de Huang.

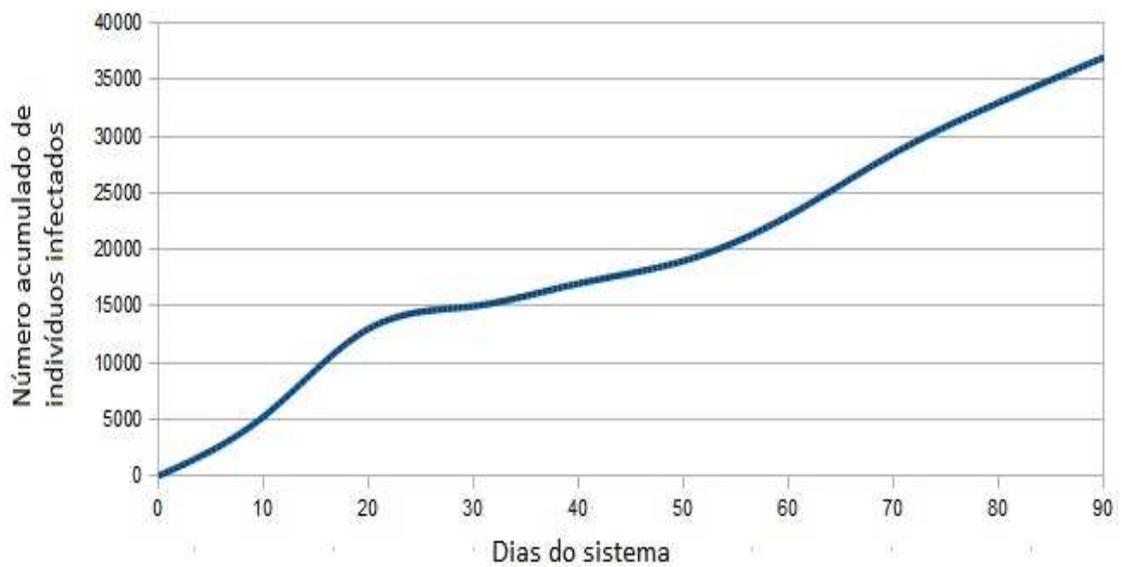


Figura 7.7: Número acumulado de indivíduos infectados obtidos durante simulação de 90 dias com a ferramenta desenvolvida

de White, porém, a taxa de transmissão da doença é 100%. Como pode-se observar, a doença se disseminou mais rapidamente.

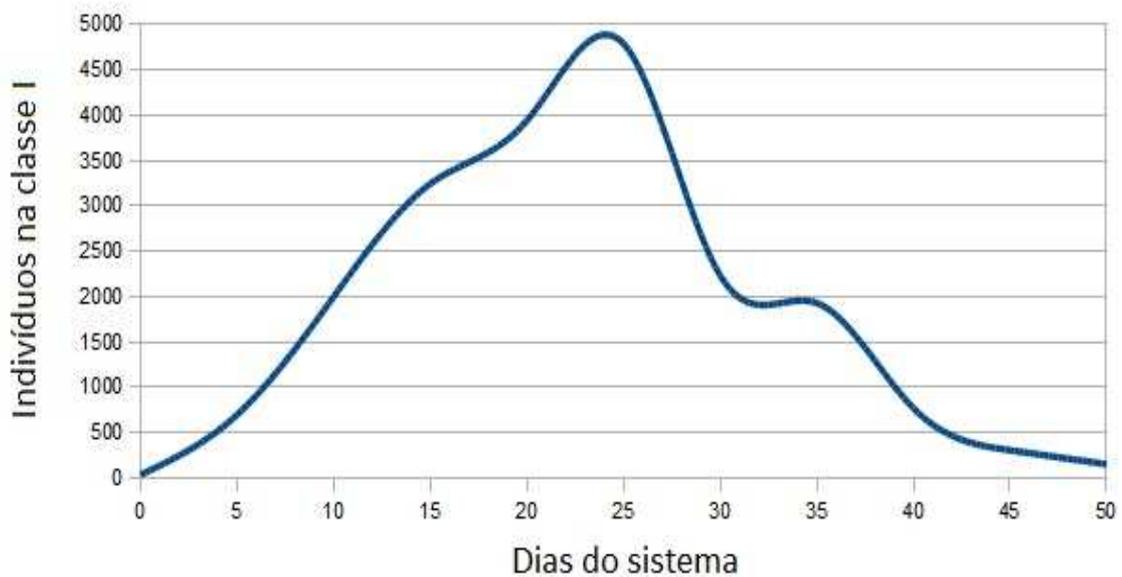


Figura 7.8: Número de indivíduos infectados na simulação de 50 dias com taxa de transmissão igual a 100%

Como era de se esperar a disseminação da epidemia ocorreu mais rapidamente, alcançando seu pico com 10 dias de antecedência, já que a cada contato adequado entre um indivíduo infectado e um indivíduo suscetível a doença é transmitida. Sendo assim, pôde-se perceber que a configuração do parâmetro de *taxa de transmissão* influencia no resultado da simulação, devendo ser corretamente ajustada a fim de resultar em uma simulação mais correta.

7.3.3 Análise do comportamento da ferramenta desenvolvida em simulações realizadas em ambientes divididos em comunidades

A existência da estrutura de comunidades pode afetar substancialmente as dinâmicas epidemiológicas. Desta forma, para a definição do ambiente de simulação que representa o município de Cascavel, foi utilizado um mapa que representa os bairros da cidade de Cascavel, onde cada bairro pode ser considerado uma comunidade. Este mapa pode ser visto na Figura 7.9.

Tal como acontece no modelo real existem conexões entre bairros que representam as vias principais entre bairros do município, através das quais pessoas (agentes computacionais) se movimentam de forma mais direcionada. Através destas conexões, tanto as pessoas (no modelo real) quanto os agentes computacionais (no aplicativo desenvolvido) mantém contato uns com os outros, e é através desses contatos que pode ocorrer a propagação de uma determinada

doença.

Pode-se observar na Figura 7.9 as conexões entre os bairros destacadas em amarelo. Estas conexões foram estabelecidas de acordo com o maior fluxo de indivíduos entre bairros da cidade, orientando-se com maior frequência dos bairros para o centro da cidade, e com menor frequência entre os bairros periféricos.



Figura 7.9: Em amarelo estão as conexões entre bairros do município de Cascavel

A fim de verificar a influência das comunidades em um ambiente de simulação, foram feitas duas simulações sob o ambiente que representa o município de Cascavel, com a mesma epidemia fictícia.

Na primeira simulação, foi inserido um indivíduo infectado no centro da cidade, enquanto na segunda simulação, o indivíduo infeccioso foi posicionado em um bairro periférico, o bairro Floresta (ao norte no mapa).

Quanto à epidemia simulada, foi escolhida uma doença do modelos compartimental SI, que deixa bastante claro no ambiente de simulação o local onde surgem novos casos. A taxa de

transmissão é de 6% e a simulação durou 50 dias do sistema. O número de agentes na simulação é 2000 onde um destes está infectado desde o início da simulação.

Na Figura 7.10 pode-se ver o espalhamento da epidemia sobre o mapa do ambiente quando inserido um indivíduo infectado no centro da cidade. Como pode-se notar, em cinquenta dias do sistema, a epidemia se espalhou por quase toda a cidade, e apenas em alguns dos bairros mais afastados do centro não existem ocorrências.

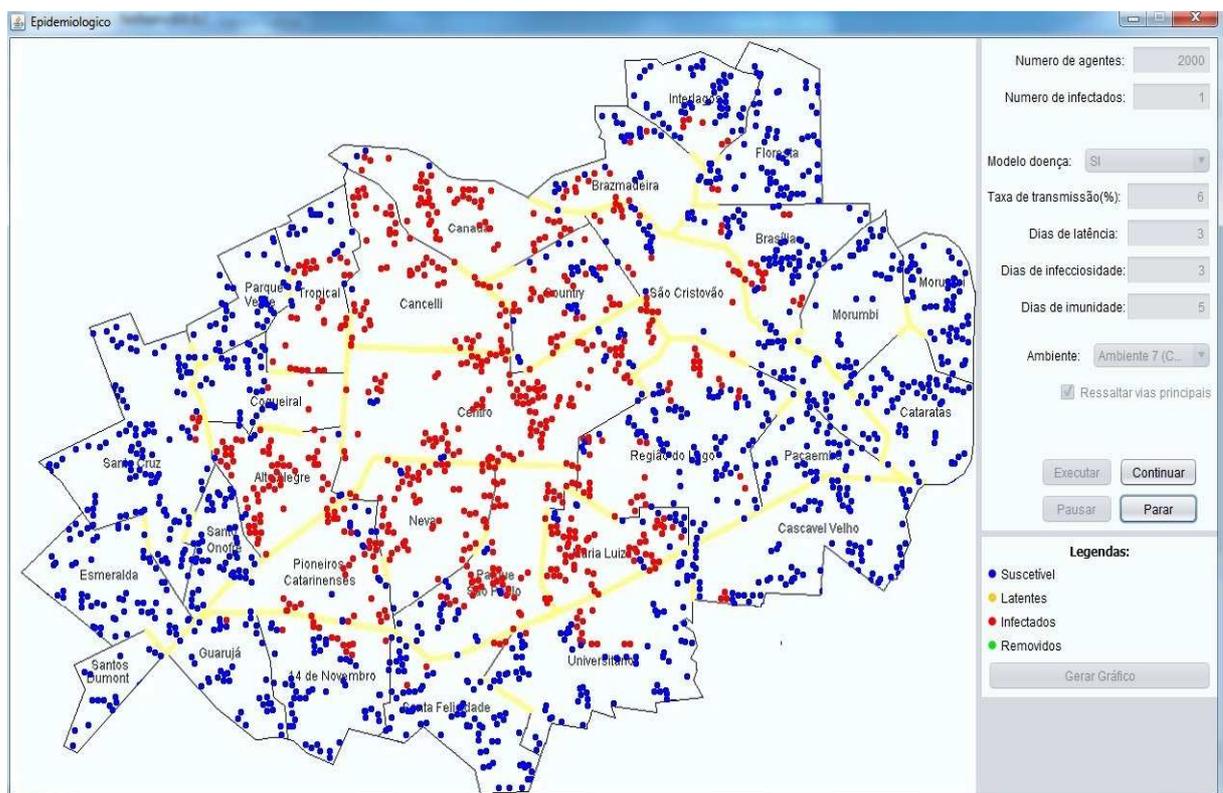


Figura 7.10: Espalhamento da epidemia inserida no centro da cidade

Porém, na simulação onde a epidemia foi inserida em um bairro periférico, o bairro Floresta, esta espalhou-se menos, atingindo um menor número de bairros. A Figura 7.11 apresenta o ambiente desta simulação após cinquenta dias do sistema.

Nas Figuras 7.12 e 7.13 pode-se ver os gráficos gerados nas simulações supracitadas. Como era de se esperar, a disseminação da epidemia atingiu mais indivíduos na simulação em que surgiu no centro da cidade do que na simulação em que a disseminação foi iniciada em um bairro da periferia.

Deste modo, considerando-se que há uma maior aglomeração de indivíduos no centro do

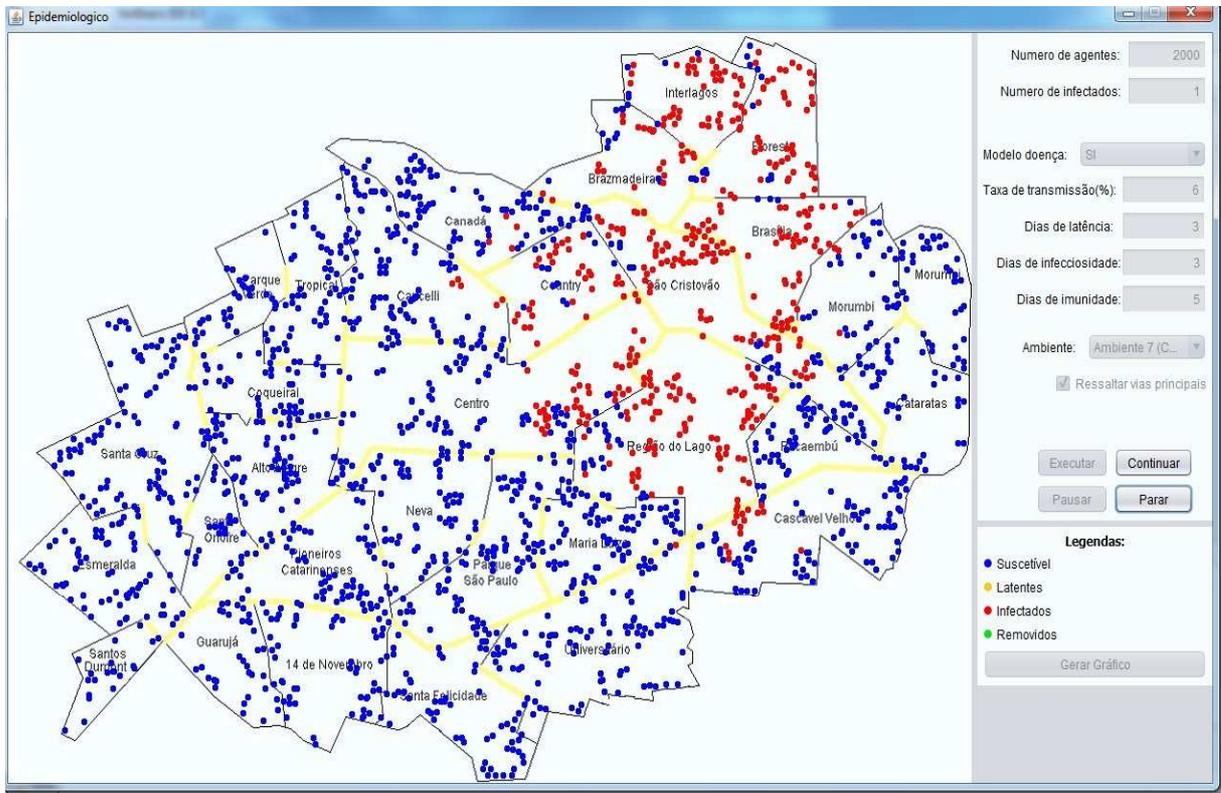


Figura 7.11: Espalhamento da epidemia inserida no bairro Floresta

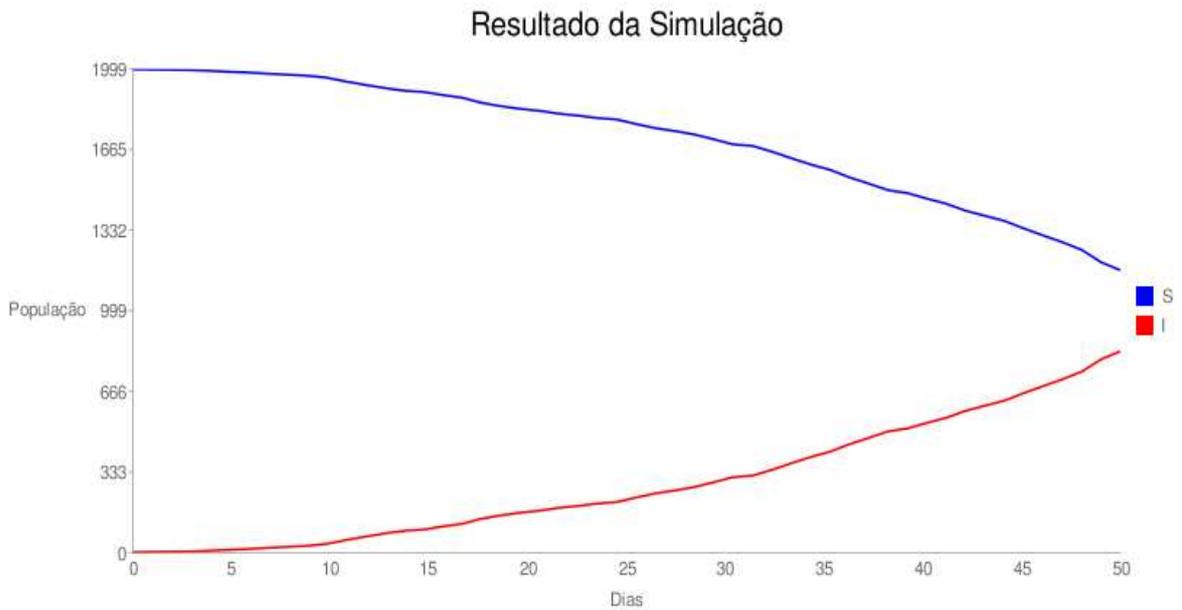


Figura 7.12: Gráfico da simulação onde o indivíduo infectado é inserido no centro da cidade

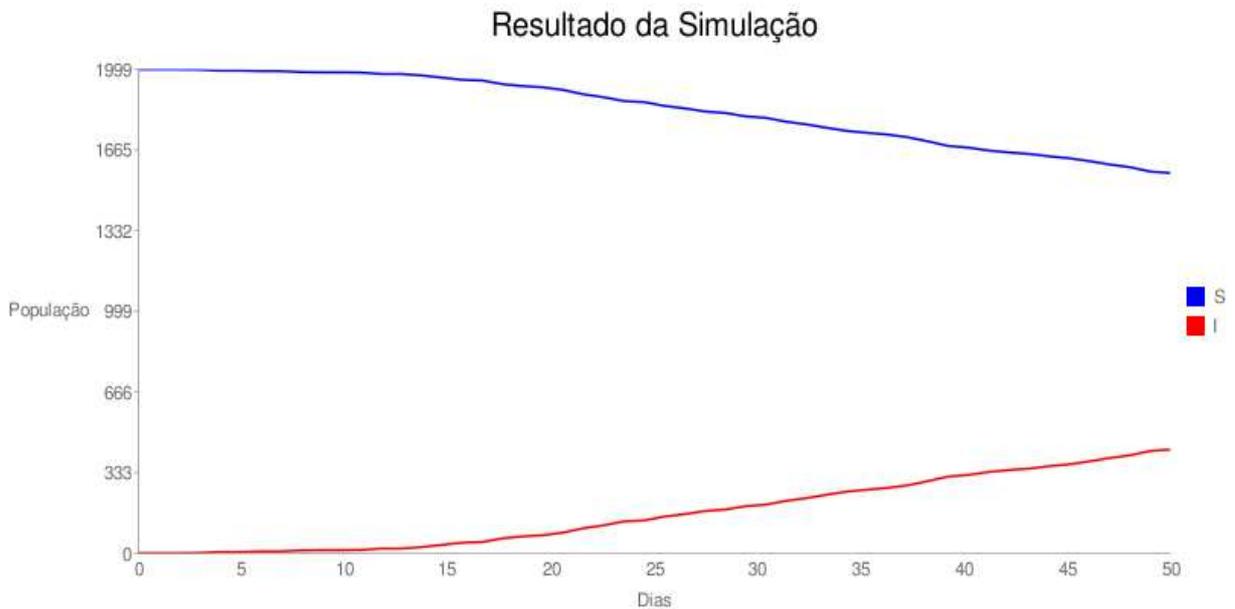


Figura 7.13: Gráfico da simulação onde o indivíduo infectado é inserido no bairro Floresta

município de Cascavel, e que essa aglomeração contribui na disseminação de doenças infecciosas, pode-se afirmar que a **divisão do ambiente em comunidades colabora para que o aplicativo desenvolvido represente de maneira mais realística as iterações que levam a um surto epidemiológico.**

7.4 Considerações Finais

Pelos experimentos realizados nesta seção, pôde-se perceber que a modelagem utilizada na implementação da ferramenta é **capaz de simular fenômenos epidemiológicos**. Além disso, dado o objetivo deste trabalho de entender os conceitos de agentes computacionais e epidemiologia, bem como demonstrar **de que forma o comportamento de uma epidemia pode ser traduzida em programas funcionais com o uso de agentes computacionais**, as experiências apresentadas podem ser consideradas como suficientes, mesmo não tendo sido feitas avaliações mais profundas acerca do desempenho da implementação proposta quanto à utilização de recursos computacionais.

Capítulo 8

Conclusões

Concretamente, este trabalho apresenta como produto final um **programa de simulação de eventos epidemiológicos** capaz de simular oito modelos compartimentais de doenças infecciosas. Tal aplicativo pode ser considerado válido e eficiente quando comparado a outros trabalhos e programas existentes na literatura que se propõem a resolver problemas similares utilizando diversas técnicas da computação, já que nos experimentos realizados apresentou resultados equivalente a tais produções.

Lembrando que o objetivo mais geral deste documento é desenvolver um trabalho de pesquisa relacionando as áreas de epidemiologia e agentes computacionais, a fim de gerar uma implementação de uma ferramenta para simulação epidemiológica empregando a abordagem de agentes computacionais, frente aos resultados teóricos e concretos obtidos, pode-se admitir que este foi plenamente alcançado.

Todavia, além do programa produzido, outras partes deste estudo merecem destaque, como a pesquisa bibliográfica, a reunião de teorias e a contribuição acadêmica com os campos de agentes computacionais e epidemiologia computacional. Isto por que, do ponto de vista teórico, a ferramenta desenvolvida representa a **concretização de uma modelagem do comportamento de uma epidemia inserida em uma sociedade de agentes computacionais**. De fato, o modelo obtido - produzido inteiramente segundo a teoria de sistemas multiagentes - simula o espalhamento de uma epidemia em uma comunidade, baseando-se em interações locais entre seus agentes.

Além disso, o estudo acerca do tema epidemiologia foi de suma importância, não só por inspirar uma representação mais fiel à realidade na concepção dos agentes, mas também identificar os processos comportamentais de diversos modelos compartimentais de epidemias.

Como dito no Capítulo 7, os estudos realizados durante o desenvolvimento deste trabalho resultaram ainda no desenvolvimento de dois artigos, publicados em eventos regionais:

- ECA - Encontro em Computação Aplicada.
- C3N - Congresso da Academia Trinacional de Ciências.

De fato, os resultados obtidos durante as experimentações (disponíveis na Seção 7.3) foram coerentes com os resultados de simulações feitas em trabalhos encontrados na literatura, sendo consideradas suficientes para validar o correto funcionamento do simulador desenvolvido.

Este trabalho não teve como objetivo ou pretensão simular todos os modelos compartimentais ou resolver todos os problemas apresentados relacionados com a modelagem determinística de epidemias. Deixa portanto, como sugestões para possíveis trabalhos futuros:

- A implementação da simulação de outros modelos compartimentais, como o SIRC, entre outros;
- A implementação de outros ambientes mais realísticos, como o que simula os bairros da cidade de Cascavel, Paraná;
- O desenvolvimento de funções de interface de usuário mais interativas, como a inserção de agentes no ambiente através de cliques no mesmo durante a simulação;
- Refinar as relações entre os agentes computacionais, inserindo características específicas que os diferenciem uns dos outros.

Referências Bibliográficas

- [1] AMAR, J. G. The monte carlo method in science and engineering. **Computing in Science and Engineering**, [S.l.], v.8, n.2, p.9–19, 2006.
- [2] ANGELOSI, K. H.; RIZZI, R. L. Modelos computacionais em epidemiologia: Estudos de casos. In: ECA - ENCONTRO DE COMPUTAÇÃO APLICADA, 2008. **Proceedings...** Cascavel, PR: [s.n.], 2008.
- [3] ANTIQUEIRA, L. **Desenvolvimento de técnicas baseadas em redes complexas para sumarização extrativa de textos**. São Carlos - SP: USP – Universidade de São Paulo, Março, 2007. Dissertação.
- [4] ANTONIAZI, D. **Propagação de Epidemias em Redes Complexas Estruturadas**. Cascavel - PR: UNIOESTE – Universidade Estadual do Oeste do Paraná, Dezembro, 2008. Monografia.
- [5] BARROS, L. C.; LEITE, M. B. F.; BASSANEZI, R. C. Epidemiologia e teoria fuzzy. In: BIOMATEMÁTICA IX, 1999. [s.n.], 1999. p.40–47.
- [6] BITTENCOURT, G. **Inteligência Artificial Distribuída**. Apostila. Departamento de Automação de Sistemas - Universidade Federal de Santa Catarina.
- [7] BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML: guia do usuário**. 1. ed. Rio de Janeiro: Campus, 2000.
- [8] CANTOR, G. Über eine eigenschaft des inbegriffes aller reellen algebraischen zahlen. **Journal für die reine und angewandte Mathematik**, [S.l.], v.77, p.258–262, 1874.

- [9] CARABELEA, C.; BOISSER, O.; FLOREA, A. Autonomy in multi-agent systems: A classification attempt. **Agents and Computational Solutions**, Berlin, p.103–113, February, 2004.
- [10] CASAGRANDI, R. et al. The SIRC model and influenza A. **Mathematical Biosciences**, [S.l.], v.200, n.2, p.152 – 169, 2006.
- [11] CÂMARA, G.; MONTEIRO, A. M. V. Geocomputation techniques for spatial analysis: are they relevant to health data? **Caderno de Saúde Pública**, Rio de Janeiro, RJ, v.17, n.5, p.1059–1081, Setembro-outubro, 2001.
- [12] DEITEL, H. M.; DEITEL, P. J. **Java: Como Programar**. 6. ed. Prentice-Hall, 2005.
- [13] DEWITTE, S. N.; WOOD, J. W. Selectivity of black death mortality with respect to preexisting health. **Proc Natl Acad Sci U S A**, Albany, NY, v.105, n.5, p.1436–1441, February, 2008.
- [14] DIETRICH, S.; MUHAMMAD, S. Discrete simulation of the dynamics of spread of extreme opinions in a society. **Physica A**, [S.l.], v.364, p.537–543, 2006.
- [15] DE OLIVEIRA, P. M. C. **Complexidade e Caos**, capítulo Autômatos Celulares. UFRJ/COPEA, Rio de Janeiro, RJ, 1. ed., 1999.
- [16] DA SILVA, A. P.; PEREIRA, J. C. C.; CORREIA, M. A. **Modelagem computacional sobre a expansão epidemiológica da esquistossomose**. Artigo. UFRPE - Universidade Federal Rural de Pernambuco, DEINFO - Departamento de Estatística e Informática.
- [17] FERREIRA, E. F. et al. Sistema para simulação e análise computacional de modelos epidêmicos. In: X ENCONTRO LATINO AMERICANO DE INICIAÇÃO CIENTÍFICA E VI ENCONTRO LATINO AMERICANO DE PÓS-GRADUAÇÃO, In: X ENCONTRO LATINO AMERICANO DE INICIAÇÃO CIENTÍFICA E VI ENCONTRO LATINO AMERICANO DE PÓS-GRADUAÇÃO. **Proceedings...** Universidade do Vale do Paraíba: [s.n.], [S.d.]. p.105–108.

- [18] **FIPA - Foundation for Intelligent Physical Agent.** Disponível em: <http://www.fipa.org>. Acessado em julho de 2009.
- [19] FRANKLIN, S.; GRAESSER, A. Is it an agent or just a program? a taxonomy from autonomous agents. In: **THIRD INTERNATIONAL WORKSHOP ON AGENT THEORIES, ARCHITECTURES AND LANGUAGES: INTELLIGENT AGENTS III**, 1996. [s.n.], 1996. p.21–36.
- [20] FREITAS, F. L. G. D.; BITTENCOURT, G. Comunicação entre agentes em ambientes distribuídos abertos: o modelo “peer-to-peer”. In: **REIC - REVISTA ELETRÔNICA DE INICIAÇÃO CIENTÍFICA**, 2002. **Proceedings...** São Paulo: [s.n.], 2002. v.2.
- [21] GAIDET, N. et al. Evidence of infection by h5n2 highly pathogenic avian influenza viruses in healthy wild waterfowl. **PLoS Pathog**, Legnaro, Italy, v.4, n.5, Agosto, 2008.
- [22] GALANTE, G. **Epidemiologia matemática e computacional.** Cascavel, PR: Universidade Estadual do Oeste do Paraná, 2008. Semestral.
- [23] GALANTE, G. Simulador epidemiológico baseado em redes de escala livre. In: **ECA - ENCONTRO DE COMPUTAÇÃO APLICADA**, 2008. **Proceedings...** Cascavel - PR: [s.n.], 2008.
- [24] GALANTE, G. et al. Simulações de eventos epidemiológicos baseadas em redes de contatos e agentes computacionais. In: **C3N - CONGRESSO DA ACADEMIA TRINACIONAL DE CIÊNCIAS**, 2009. **Proceedings...** Foz do Iguaçu, PR: [s.n.], 2009.
- [25] GALANTE, G.; RIZZI, C. B.; RIZZI, R. Simulating epidemiological processes using scale-free networks with community structures. Maio, 2009.
- [26] GARCIA, A. F. **Objetos e Agentes: Uma Abordagem Orientada a Aspectos.** Rio de Janeiro, RJ: Pontifícia Universidade Católica do Rio de Janeiro, Abril, 2004. Tese.

- [27] GENESERETH, M. R.; KETCHPEL, S. P. Software agents. **Communications of the ACM**, [S.l.], v.38, n.7, p.48–53, 1994.
- [28] GOMES, M. C. Modelação da transmissão de doenças. Lisboa: Universidade de Lisboa, 2003. Relatório técnico.
- [29] GOTTFREDSSON, M. et al. Lessons from the past: Familial aggregation analysis of fatal pandemic influenza (spanish flu) in iceland in 1918. **Proc Natl Acad Sci U S A**, Islândia, v.105, n.4, p.1303–1308, January, 2008.
- [30] HETHCOTE, H. W. The mathematics of infectious diseases. **SIAM Review**, Philadelphia, PA, USA, v.42, n.4, p.599–653, 2000.
- [31] HOPCROFT, J. E.; MOTWANI, R.; ULLMAN, J. D. **Introdução à Teoria de Autômatos, Linguagens e Computação**. 1. ed. Campus, 2002.
- [32] HUANG, C.-Y.; SUN, C.-T.; LIN, H.-C. Influence of local information on social simulations in small-world network models. **Journal of Artificial Societies and Social Simulation**, [S.l.], v.8, n.4, p.8, 2005.
- [33] JORGENSEN, E. Calibration of a monte carlo simulation model of disease spread in slaughter pig units. **Computers and Electronics in Agriculture**, [S.l.], v.25, p.245–259, 2000.
- [34] KARSTEN, S.; RAVE, G.; KRIETER, J. Monte carlo simulation model of classical swine fever epidemics and control. i. general concepts and description of the model. **Veterinary Microbiology**, [S.l.], v.108, p.187–198, 2005.
- [35] KARSTEN, S.; RAVE, G.; KRIETER, J. Monte carlo simulation model of classical swine fever epidemics and control. ii. validation of the model. **IEEE Transactions on Power Systems**, [S.l.], v.108, p.199–205, 2005.
- [36] KLÜGL, F. **Simulated Ant Colonies as a Framework for Evolutionary Models**. 2002, 12p. Universität Würzburg: Dept. of Artificial Intelligence and Applied Computer Science Julius-Maximilians.

- [37] LAROUSE; ÁTICA. **Dicionário da Língua Portuguesa**. Ática, São Paulo, SP, 2001.
- [38] MERCHANT, F. **Simulating the spread of Malaria: A Cellular Automaton Based Mathematical Model & a Prototype Software Implementation**. Blacksburg, Virginia: Virginia Polytechnic Institute and State University, February, 2007. Tese.
- [39] NEWMAN, M. E. J. Models of the small world: A review. **Journal of Statistical Physics**, [S.l.], v.101, p.819–841, May, 200.
- [40] NUNES, D. V. F. et al. A utilização de agentes computacionais na modelagem não determinísticas de eventos epidemiológicos. In: ECA - ENCONTRO EM COMPUTAÇÃO APLICADA, 2009. **Proceedings...** Cascavel, PR: [s.n.], 2009.
- [41] ORTEGA, N. R. S. **Aplicação da Teoria de Conjuntos Fuzzy a Problemas da Biomedicina**. São Paulo, SP: Universidade de São Paulo, 2001. Tese.
- [42] PAGLIOSA, A. L. **Obtenção das funções de pertinência de um sistema neuro-fuzzy modificado pela rede de Kohonen**. Centro de Ciências Tecnológicas - CCT / Departamento de Engenharia Elétrica - DEE: Universidade do Estado de Santa Catarina - UDESC, Dezembro, 2003. "dissertação de mestrado".
- [43] PAN, C. Z.; RIZZI, R. L. Epidemiologia matemática: Conceitos, processos e modelos. In: ECA - ENCONTRO DE COMPUTAÇÃO APLICADA, 2008. **Proceedings...** Cascavel, PR: [s.n.], 2008.
- [44] PEIXOTO, M. S.; BARROS, L. C. Um estudo de autômatos celulares para o espalhamento geográfico de epidemias com parâmetro fuzzy. **TEMA Tend. Mat. Apl. Comput.**, Campinas, SP, v.5, n.1, p.125–133, 2004.
- [45] RIZZI, C. B. et al. Epidemio - sistema computacional web integrado sobre eventos epidemiológicos. In: ECA - ENCONTRO DE COMPUTAÇÃO APLICADA, 2008. **Proceedings...** Cascavel, PR: [s.n.], 2008.

- [46] ROSSI, M. **Modelo matemático da resposta imune à infecção pelo vírus HIV-1**. São Paulo, SP: USP - Universidade de São Paulo, 2008. Tese.
- [47] RUSSEL, S.; NORVIG, P. **Inteligência Artificial**. 4. ed. São Paulo, SP: Editora Campus, 2004.
- [48] SANDRI, S.; CORREA, C. Lógica nebulosa. In: V ESCOLA DE REDES NEURAIAS, 1999. **Proceedings...** ITA - São José dos Campos - SP: [s.n.], 1999. p.c073–c090.
- [49] SITUNGKIR, H. Epidemiology through cellular automata. case of study: Avian influenza in indonesia. **Board of Science**, Bandung Fe, January, 2004.
- [50] SUN, Y.; CHENG, L. A survey on agent-based modeling and equation-based modeling. 2006. **Proceedings...** Atlanta: [s.n.], 2006.
- [51] TÔRRES, J. J. M. **Lógica Fuzzy**. Disponível em <<http://users.femanet.com.br/fabri/fuzzy.htm>>. Acesso em: maio. 29.
- [52] TRAN, A.; RAFFY, M. The dynamics os dengue epidemics from large-scale information. **Theoretical Population Byology**, [S.l.], v.69, n.1, p.3–12, 2006.
- [53] WALDMAN, E. A. **Vigilância em Saúde Pública**, v.7. São Paulo, SP: Fundação Petrópolis, 1998.
- [54] WANG, X. F.; CHEN, G. Complex networks: small-world, scale-free and beyond. **Circuits and Systems Magazine, IEEE**, [S.l.], v.3, n.1, p.6–20, 2003.
- [55] WANG, T. T.; PALESE, P. Unraveling the mystery of swine influenza virus. **Cell**, New York, NY, v.137, n.6, p.983–985, June, 2009.
- [56] WHITE, S.; DEL REY, A. M.; SÁNCHEZ, G. R. Modelling epidemics using cellular automata. **Applied Mathematics and Computation**, Salamanca, Espanha, v.186, p.193 – 202, junho, 2007.
- [57] WOOLDRIDGE, M.; JENNINGS, N. R. Intelligent agents: Theory and practice. 1994. **Proceedings...** London, United Kingdom: [s.n.], 1994.

- [58] WOOLDRIDGE, M. Agent-based software engineering. 1997. **Proceedings...** London, United Kingdom: [s.n.], 1997.
- [59] WOOLDRIDGE, M.; JENNINGS, N. R. The cooperative problem-solving process. In: JOURNAL OF LOGIC AND COMPUTATION, 1999. **Proceedings...** Londres: [s.n.], 1999. v.9, p.563–592.
- [60] WOOLDRIDGE, M. Intelligent agents: The key concepts. 2002. **Proceedings...** Liverpool, United Kingdom: [s.n.], 2002.
- [61] ZADEH, L. A. Fuzzy sets. **Information and Control**, [S.l.], v.8, n.3, p.338–353, 1965.
- [62] ZHOU, H. Research of the small-world character during rumor's propagation. **Journal of Wuhan University of Science and Engineering**, [S.l.], v.18, n.1, p.113–116, 2005.