

UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
Instituto de Geociências e Ciências Exatas – IGCE
Curso de Bacharelado em Ciências da Computação

RENATO LUCIANO CAGNIN

**UMA ABORDAGEM DE MULTIAGENTES BROWNIANOS PARA A
MODELAGEM E SIMULAÇÃO DA DINÂMICA DAS POPULAÇÕES
NO CONTÁGIO DA ESQUISTOSSOMOSE**

Trabalho realizado sob orientação do Prof. Ivan Rizzo Guilherme, DEMAC/IGCE
Período: 03.2012 a 07.2012

Rio Claro-SP
2012

UMA ABORDAGEM DE MULTIAGENTES BROWNIANOS PARA A MODELAGEM E SIMULAÇÃO DA DINÂMICA DAS POPULAÇÕES NO CONTÁGIO DA ESQUISTOSSOMOSE

Trabalho de Conclusão do Curso, modalidade Trabalho de Graduação, apresentado, no 1º semestre de 2012, à disciplina ES/TG do Curso de Bacharelado em Ciências da Computação, período Integral, do Instituto de Geociências e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, campus de Rio Claro, para apreciação segundo as normas estabelecidas pelo Conselho do Curso, em 27.11.2007.

Aluno: Renato Luciano Cagnin

Orientador: Ivan Rizzo Guilherme
DEMAC / IGCE

Coorientador: Alexandro José Baldassin
DEMAC / IGCE

Rio Claro-SP
2012

SUMÁRIO

1. INTRODUÇÃO	1
2. SIMULAÇÃO MULTIAGENTES E A ESQUISTOSSOMOSE	3
2.1 SISTEMAS MULTIAGENTES	4
2.2 A ESQUISTOSSOMOSE	5
3. A ABORDAGEM MULTIAGENTES BROWNIANOS.....	8
3.1. AGENTES BROWNIANOS E A MODELAGEM DO PROBLEMA	8
3.2. MODELAGEM DO AMBIENTE	13
3.3. MÉTODO NUMÉRICO APLICADO AO PROBLEMA	14
4. MODELAGEM COMPUTACIONAL E SIMULAÇÃO	17
4.1 A METODOLOGIA DE DESENVOLVIMENTO	17
4.2 TECNOLOGIAS UTILIZADAS	18
4.3 ARQUITETURA LÓGICA DO SIMULADOR.	20
4.4 MODELAGEM CONCEITUAL DO PROBLEMA.	22
4.5 FUNCIONAMENTO DO SIMULADOR.	28
5. RESULTADOS E DISCUSSÃO	34
5.1 ANÁLISE DA INFLUÊNCIA DO AMBIENTE SOBRE O CONTÁGIO	35
5.2 ANÁLISE DA INFLUÊNCIA DA POPULAÇÃO DE HUMANOS SOBRE O CONTÁGIO	42
5.3 ANÁLISE DA INFLUÊNCIA DA POPULAÇÃO DE CARAMUJOS SOBRE O CONTÁGIO	45
6. CONCLUSÃO	49
REFERÊNCIAS	51

LISTA DE FIGURAS

Figura 1 – Ciclo resumido de contágio da esquistossomose	6
Figura 2 - Diagrama apresentando a interação entre o campo de informação e os agentes	12
Figura 3 - Dinâmica populacional. Na figura, a seta indica o sentido com que a informação flui entre as populações. Populações do tipo Vetor escreve no campo de informações, enquanto que as populações do tipo Parasita leem a informação transferida ao campo.	13
Figura 4 - Diagrama apresentando a relação e a dependência entre diferentes pacotes que foram construídos para a modelagem do problema	22
Figura 5 - Diagrama apresentando a a relação entre as classes que definem os agentes modelados presentes no pacote model.agents do simulador.	24
Figura 6 : Diagrama apresentando a a relação entre as classes que definem o ambiente geográfico do simulador.	25
Figura 7 - Diagrama apresentando a a relação entre as classes que definem o campo de informações, aspecto chave para a modelagem do problema	26
Figura 8 - Diagrama apresentando a a relação entre as classes presentes nos diferentes pacotes apresentados anteriormente.	26
Figura9 - Diagrama apresentando a a relação entre as camadas implementadas a partir do modelo MVC.	27
Figura 10 : Diagrama de sequência apresentando o fluxo principal do simulador.	30
Figura 11 - Diagrama de sequência apresentando o fluxo proliferação e contágio	31
Figura 12 - Diagrama de sequência apresentando o fluxo principal de simulação	32
Figura 13 - Diagrama de sequência apresentando o fluxo principal de simulação	32
Figura 14 - Dispersão da informação (contágio) nos instantes iniciais da simulação 1.1.	37
Figura 15 - Simulação 1.1 em tempo intermediário. Observa-se uma alta fração de agentes do tipo caramujo contaminados.	38
Figura 16 - Término da simulação 1.1. Observa-se uma alta fração de agentes do tipo caramujo contaminados.....	39
Figura 17 - Estado inicial da simulação 1.2	40
Figura 18 - Simulação 1.2 em tempo intermediário. Observa-se uma alta fração de agentes do tipo caramujo contaminados	40
Figura 19 - Término da simulação 1.2. Observa-se uma alta fração de agentes do tipo caramujo contaminados e um aumento significativo no número de humanos contaminados em comparação com a simulação anterior	41
Figura 20 - Início da simulação 1.3. Observa-se um aumento significativo da ocupação do ambiente geográfico por partes dos agentes caramujos	41
Figura 21 - Término da simulação 1.3	42

Figura 22 - Estado final da simulação 2.1.....	43
Figura23 - Estado final da simulação 2.2.....	44
Figura 24 - Estado da simulação 2.3, após 400 passos.....	44
Figura 25 - Estado da simulação 2.4, após 200 passos.	45
Figura 26 - Estado final da simulação 3.1.	46
Figura 27 - Estado final da simulação 3.2	47
Figura 28 - Estado da simulação 3.3 após 800 passos	47
Figura 29 - Estado da simulação 3.4. após 300 passos	48

1. INTRODUÇÃO

Os benefícios do uso de simulações computacionais se tornaram reconhecidos em diversas áreas do conhecimento. As simulações computacionais têm se tornado úteis na modelagem de vários sistemas físicos, químicos e biológicos, cujo objetivo é identificar o comportamento de um sistema real [1].

Simulações multiagentes são caracterizadas por apresentarem vários agentes agindo sobre um ambiente complexo e interagindo entre si, o que as torna flexíveis o suficiente para a resolução de problemas com características de sistemas complexos [1]. Estas simulações têm demonstrado ao longo dos anos a capacidade de representar com adequação sistemas sociais e interagentes, possibilitando a modelagem computacional de sistemas ecológicos, bem como a interação entre os indivíduos e o ambiente [2, 3, 4]. Outra aplicação de simulações baseadas em agentes se dá no contexto do contágio e disseminação de doenças e na relação entre vetores e parasitas [5].

As simulações multiagentes surgiram da análise de sistemas complexos e adaptativos, cuja capacidade de readaptação e organização surge a partir da análise do sistema como um todo. Em particular, existem quatro características que estão associadas a estes sistemas: capacidade de formação de grupos, não linearidade entre cada elemento constituinte, transferência de recursos e informação, e por fim, diversidade. Estas características tornam problemas de contágio e disseminação de doenças, candidatos para a modelagem a partir de sistemas complexos.

A esquistossomose é uma doença transmitida pelo parasita do gênero *Schistossoma* e tem como vetor intermediário o caramujo do gênero *Biomphalaria* e o homem como hospedeiro definitivo [6]. Existem seis espécies do parasita *Schistossoma* que podem provocar a doença no homem, contudo na região sul-americana, a predominância de indivíduos ocorre na espécie *Schistossoma mansoni*. A esquistossomose mansônica ocorre em regiões onde

existem córregos, açudes e baixo saneamento básico ou inexistente [6, 7].

O mecanismo de transmissão da doença é bastante complexo, e envolve a interação entre diferentes populações de diferentes espécies. Dessa forma, o ciclo de vida do parasita depende da interação entre o homem e as espécies de caramujos, e também de condições propícias para o desenvolvimento do parasita [8, 9].

Um controle sobre a população de vetores intermediários e formas eficazes de tratamento da doença tem sido o alvo do combate da doença ao longo dos anos em diversas localidades do território brasileiro. Por isso, é de interesse para as autoridades responsáveis, um plano de contingência sobre a disseminação da doença bem como o controle sobre as áreas de risco [8, 9]. Uma simulação computacional do problema proposto poderia destacar potenciais áreas de risco e estratégias ótimas para aplicações de medidas profiláticas.

Dessa forma, é objetivo principal deste trabalho de conclusão de curso (TCC) desenvolver um simulador para descrever a interação entre a população de parasitas e a população de vetores (intermediários e principais), determinante para o contágio da esquistossomose. Outro objetivo que deseja-se atingir é a representação adequada do problema do contágio e da dinâmica das populações representadas pelo simulador, bem como a validação dos dados obtidos pelas simulações. Outra informação importante que resulta do projeto é a possibilidade de prever eventos simulando-os computacionalmente.

O trabalho segue estruturado da seguinte forma: no capítulo 2, foi realizado um levantamento bibliográfico sobre os tópicos necessários para a conclusão do trabalho. Neste capítulo, uma breve introdução ao problema do contágio da esquistossomose será apresentada, bem como a descrição de seus fatores determinantes. Ainda no capítulo 2, serão descritos alguns parâmetros importantes, associados às populações de vetores e parasitas, que serão utilizados no simulador, tais como: o tempo de vida das espécies, a distribuição geográfica, o tipo de nutrição, o habitat entre outros. No capítulo 3, segue a descrição das metodologias de implementação e técnicas matemáticas necessárias para a modelagem computacional do problema. O capítulo 4 descreve em detalhes a partir de diagrama UML, a modelagem conceitual, o funcionamento e a arquitetura do simulador. No capítulo 5, serão apresentados os resultados seguido da discussão dos mesmos, obtidos a partir da execução de alguns cenários por parte do simulador. E por fim, o capítulo 6 traz as conclusões do trabalho e sugestões para futuros trabalhos.

2. SIMULAÇÃO MULTIAGENTES E A ESQUITOSSOMOSE

Simulações computacionais têm sido empregadas como processos para suporte a decisão. Uma simulação computacional funciona como um laboratório, onde determinados parâmetros e comportamentos podem ser obtidos e interpretados mediante diferentes condições de entrada [10]. Dessa forma, é possível “prever” comportamentos futuros e compreender melhor assuntos específicos. Uma das aplicações de simulações computacionais ocorrem em sistemas cujo comportamento não pode ser descrito analiticamente por equações diferenciais lineares ou mesmo quando a utilização de modelos determinísticos não representa adequadamente o problema em questão [11, 12]. Em particular, é muito comum o uso de simulações computacionais para o estudo de sistemas complexos.

Um sistema complexo apresenta como característica um comportamento global proveniente da dinâmica de interação entre os diferentes indivíduos que são parte do sistema como um todo [13]. Essa interação entre uma, ou mesmo diferentes populações de indivíduos, torna uma modelagem analítica inviável. Dessa forma, é necessário lançar mão de outras técnicas capazes de representar de maneira adequada o comportamento global do sistema em questão. São sistemas compostos por partículas que interagem fortemente entre si e geralmente de maneira não linear. Em sistemas complexos, a interação dos indivíduos define a dinâmica de todo sistema [14].

Uma das áreas de pesquisa da Inteligência Artificial (IA) que vem demonstrando eficiência na modelagem de sistemas complexos em especial, sistemas que apresentam interação social entre diversos indivíduos, é a área denominada de simulações baseadas em agentes (ABS – Agent Based Simulations) [15, 16], cujo objetivo é representar sistemas complexos através de simulações baseadas em agentes inteligentes. Dessa forma, justifica-se neste trabalho a adoção de uma técnica baseada em sistemas multiagentes, pois a dinâmica de populações biológicas envolvendo um ou mesmo vários tipos de diferentes indivíduos,

pressupõe características de sistemas complexos [17].

2.1 SISTEMAS MULTIAGENTES

Um agente pode ser definido como uma entidade computacional funcionando continuamente e de forma autônoma em um ambiente particular, frequentemente habitado por outros agentes e processos. Todo agente é capaz de perceber seu ambiente por meio de sensores e agir por meio de atuadores. De maneira geral, um agente pode ser pensado como uma metodologia para construção de sistemas que simulem a inteligência humana. Existem vários tipos de agentes, que se diferem quanto ao seu funcionamento interno. Neste trabalho, adotaremos uma abordagem baseada em agentes reativos simples [18].

Um agente reativo simples é um agente que reage a mudanças em seu ambiente ignorando as percepções (mudanças no ambiente) anteriores, dessa forma, um agente reativo simples apresenta inteligência (capacidade de resolução de problemas) muito limitada. Contudo quando um conjunto de agentes é reunido para atuar num mesmo ambiente, surge o conceito de multiagente, onde a existência de um conjunto ou uma população de vários agentes que atuam sobre o sistema por meio de um comportamento social, a partir da interação entre os diferentes indivíduos, simula o aparecimento de inteligência a nível global [19, 20].

Um sistema multiagente é caracterizado pelas fortes interações entre os diversos agentes que participam da dinâmica do sistema. Estes agentes interagem (competem / comunicam-se / auxiliam-se) para que um propósito, ou um comportamento, global seja atingido [21, 22]. Nos sistemas multiagentes, os agentes individuais são a parte central que deve cooperar e interagir para a solução de determinados problemas. Neste tipo de sistema, pressupõe-se que os agentes que o formam são dotados de certa autonomia e inteligência para que, independente da existência de outros agentes, consigam obter um objetivo individual.

Sistemas Multiagentes têm sido adotados com êxito em conjuntos de modelos de simulação (MABS, do inglês *Multiagent-based Simulation*). MABS são usados, sobretudo

quando se deseja modelar problemas complexos onde uma solução não pode ser atingida por um agente individualmente [23].

O estudo de sistemas complexos a partir da abordagem de multiagentes tem demonstrado resultados que caracterizam: evolução social, dados demográficos, interação de grupos sociais, comportamento de populações, etc [24, 25].

MABS são especialmente apropriados quando lida-se com um problema complexo em domínios interdisciplinares. Estes modelos têm sido adotados em função de características inerentes aos agentes, tais como pró-atividade, sociabilidade e autonomia, que facilitam a construção de modelos dinâmicos e complexos [26].

2. 2 A ESQUISTOSSOMOSE

A esquistossomose é uma doença transmissível e parasitária considerada um grave problema de saúde pública. No Brasil, a esquistossomose mansônica é endêmica em vasta extensão do território e considerada ainda um grave problema de saúde pública, pois acomete milhões de pessoas, provocando anualmente, um número expressivo de formas graves e até mesmo óbitos [6, 7].

As espécies contribuintes para a proliferação da doença, classificadas em vetores e parasitas (Figura 1), são determinantes para o ciclo de vida do seu causador, um verme trematódeo do gênero *Schistosoma* [6, 7]. Em particular, a espécie deste verme parasita existente no território brasileiro, é denominada *Schistosoma mansoni* [7].

Os vetores dessa doença são o homem, como hospedeiro definitivo e a espécie de caramujos de água doce do gênero *Biomphalaria*, como hospedeiro intermediário [6, 7, 8]. Ambas as espécies são indispensáveis para o ciclo da doença. Os organismos parasitas são denominados Miracídeos e Cercárias. Os primeiros, Miracídeos, são as formas infectantes do hospedeiro intermediário, os caramujos do gênero *Biomphalaria*, e os outros, Cercárias, são as formas infectantes do hospedeiro definitivo, o homem [7, 8].

O mecanismo de contágio da doença envolve diferentes espécies e condições

propícias. Entretanto, para que ocorra a transmissão da doença, é indispensável a presença do homem na condição de hospedeiro definitivo, e dos caramujos do gênero *Biomphalaria*, que atuam como hospedeiros intermediários, liberando as larvas infectantes (Cercárias) do *Schistosoma mansoni* nos recursos hídricos utilizados pelos seres humanos susceptíveis à doença [7, 8, 9].

O ciclo de contaminação inicia-se quando as fezes ou urina do homem contaminado carregando consigo os ovos do verme caem sobre uma região favorável onde exista uma população de caramujos do gênero *Biomphalaria*. Estes ovos eclodem em larvas (Miracídeos) que passarão a contaminar os caramujos, até que seu período de maturação seja atingido. Após este período, estas larvas se tornam Cercárias, a forma infectante da doença para o hospedeiro definitivo. O ciclo agora se fecha com a penetração das Cercárias a partir da epiderme de seu hospedeiro definitivo (Figura 1) [6, 7].

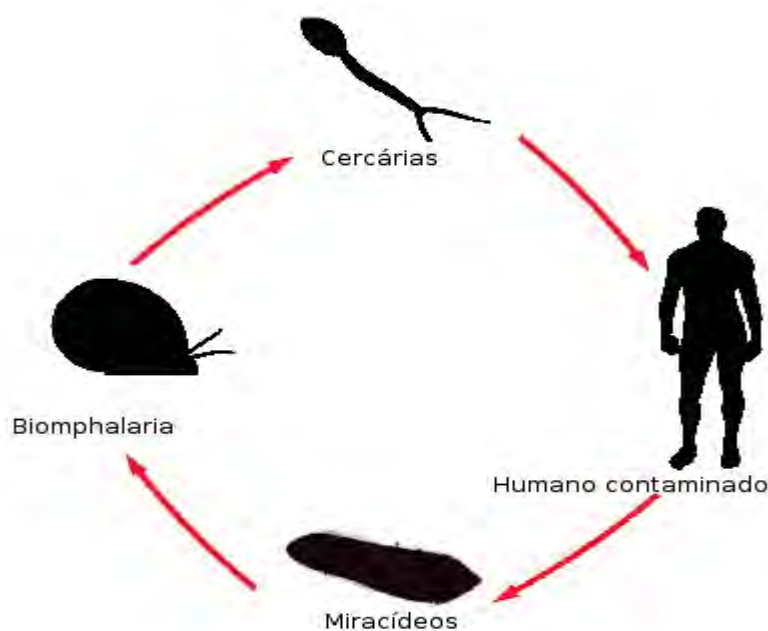


Figura 1 – Ciclo resumido de contágio da esquistossomose.

Os moluscos hospedeiros do gênero *Biomphalaria* podem ser encontrados habitando rios, valas, pequenas poças e até lagoas, sendo o recurso hídrico um fator determinante para sua sobrevivência [27, 28]. As populações humanas que convivem próximo aos locais favoráveis para o desenvolvimento dos caramujos e necessitam da água contaminada,

geralmente acabam por desenvolver a doença esquistossomose pois, a doença ocorre em localidades sem saneamento ou com saneamento básico inadequado [29, 30]. De maneira geral, um recurso hídrico é uma condição necessária para a transmissão e contágio da doença, para ambas populações de hospedeiros. Pois as larvas Miracídeos somente poderão eclodir se estiverem em meio aquático e as larvas Cercárias somente buscarão o hospedeiro definitivo se este também expor sua epiderme no meio aquático [31].

Outra informação importante que será utilizada na modelagem do problema diz respeito à longevidade das populações de vetores e parasitas. As espécies de caramujos *Biomphalaria*, apresentam um tempo de vida máximo, quando não infectados, de 25 a 30 semanas [27, 28]. Este dados dependem de vários fatores associados às condições de vida dos caramujos, e também da espécie considerada. Já quando infectados seu tempo de vida cai para, em média, 12 semanas. Quanto aos parasitas da espécie Cercária e Miracídeos, o tempo de vida médio é de somente 24 a 48 horas, se estes estiverem imersos em água [32, 33].

Uma medida para contenção e controle da doença se deve ao combate da proliferação dos caramujos do gênero *Biomphalaria*, contudo estas medidas afetam diretamente o meio ambiente, e portanto, tais medidas devem ser aprovadas e adequar-se à legislação ambiental. Outra medida contra a emergência da esquistossomose é a educação da população que se utiliza de locais potencialmente contaminados. Até a década de 70, o combate da esquistossomose tinha como objetivo principal o controle da transmissão [6, 7, 8]. Hoje o combate da esquistossomose é feito combinando estratégias quimioterápicas com políticas de melhoria do saneamento, saúde e educação.

3. A ABORDAGEM MULTIAGENTES BROWNIANOS

Neste capítulo, a abordagem de agentes brownianos será apresentada junto aos modelos matemáticos utilizados posteriormente para implementação do simulador. Também será apresentado como ocorreu a adaptação do modelo ao problema do contágio da esquistossomose.

3.1. AGENTES BROWNIANOS E A MODELAGEM DO PROBLEMA.

A metodologia para construção do modelo deste trabalho, será baseada na técnica de agentes brownianos [34], constituindo um sistema multiagentes reativo para modelagem da dinâmica do sistema formado pelas diferentes populações (humano, caramujo e parasitas). A técnica se mostrou eficaz em ensaios anteriores baseados em dinâmica de populações de outros vetores – parasitas.

Esta técnica se baseia em um fenômeno físico que ocorre a nível molecular, denominado movimento browniano. Este fenômeno é o resultado de interações de moléculas de um solvente sobre as moléculas de uma partícula insolúvel [35]. A adaptação deste fenômeno com base no comportamento de sistemas multiagentes, constitui-se hoje a chamada técnica de agentes brownianos.

Um Agente Browniano é caracterizado por um conjunto de variáveis de estado u_i^k , onde $i=1,...,N$ se refere a um agente i individual e k indica suas diferentes variáveis de estado. Estas variáveis podem ser variáveis externas (observáveis) ou graus de liberdade internos (que podem ser observados a partir somente de uma ação).

Duas importantes variáveis externas serão:

$$u_i^1 = \vec{r}_i, \quad (1)$$

a coordenada espacial do agente, e

$$u_i^2 = d \frac{d\vec{r}_i}{dt} = \vec{v}_i, \quad (2)$$

a velocidade individual do agente.

Já quanto aos graus de liberdade interno ao agente, destacam-se duas novas variáveis:

$$u_i^3 = \theta_i, \quad (3)$$

esta variável descreve a informação interna do agente. No caso do problema em questão, é possível associá-lo ao fato do agente estar ou não infectado. Para a simulação baseada em agentes brownianos que será desenvolvida neste trabalho, esta é a variável de maior importância. Esta variável estará presente nos agentes vetores (*Biomphalaria* e Homem).

Quanto à outra variável interna considerada, esta dirá respeito à energia interna do agente, ou seja, quão ativo este agente estará num dado estágio da simulação. Agentes poderão agir desde que apresentem alguma reserva de energia interna, caso contrário serão descartados da simulação [35]. No caso do problema proposto, isso é o equivalente ao metabolismo dos indivíduos das diferentes populações de parasitas. Quando a energia interna de um agente esgota-se, este agente é considerado morto na simulação, pois os parasitas (Miracídeos e Cercárias) sobrevivem temporariamente consumindo sua reserva de nutrientes.

Matematicamente a energia interna é representada a seguir:

$$u_i^4 = e_i. \quad (4)$$

O número de variáveis de estados usadas nas simulações dependerá do sistema a ser modelado. Estas variáveis se relacionarão à precisão desejada na simulação e detalhes específicos, que poderão ser omitidos ou não, caso assim demandar a complexidade do modelo [34]. Neste trabalho, serão adotadas todas as variáveis acima descritas, devido às

características do sistema real a serem representadas durante a simulação.

Uma vez definidas as variáveis de estado, o próximo passo agora é descrever a dinâmica desses agentes. O conceito físico chave usado durante toda simulação é lei da causalidade, que estabelece que para todo efeito existe uma causa, e esta poderá ser estocástica ou determinística [34, 35]. Deste modo, para qualquer variável de estado, a dinâmica será dada por:

$$d \frac{u_i^k}{dt} = f_i^k + F_i^{stoch}, \quad (5)$$

que representa todas as influências do meio sobre o agente browniano, que podem provocar possíveis transições de estados. No problema estudado, estas influências podem ser:

- interações lineares ou não-lineares com outros agentes (interação/competição/predação);
- forças externas resultantes de potenciais externos (auto-preservação/busca por condições de proliferação / condições favoráveis de contágio);
- dinâmica do sistema independente da ação dos agentes (variações climáticas, sazonalidade, alteração do meio).

Para o simulador deste trabalho, assume-se que a influência estocástica se deva à livre busca pelo ambiente geográfico a partir de cada agente, ou seja, assume-se que cada agente se deslocará aleatoriamente em torno de uma posição média. Este movimento estocástico simula o andar errante de um agente em busca de recursos, como por exemplo, alimentos.

A influência determinística já demonstra um comportamento mais complexo condicionado à natureza de cada agente. Para agentes da população de humanos, esta influência determinística se deve às necessidades do uso de recursos hídricos para satisfazer suas necessidades, tais como: beber, banhar-se, trabalhar, etc. Agentes da população de *Biomphalaria*, *Cercárias* e *Miracídeos*; têm como restrição habitar os recursos hídricos, dessa forma, o recurso hídrico pode ser modelado na forma de um potencial atrativo para estes agentes, isso significa que estes agentes estarão presos neste potencial e não serão permitidas em outras regiões no espaço geográfico.

Cada uma das variáveis externas dos agentes variam ao longo do tempo de simulação

de acordo com suas componentes estocásticas e determinísticas, dessa forma, é possível estabelecer a taxa com que cada uma dessas grandezas varia no tempo [34, 35, 36]. Para as variáveis coordenada espacial e velocidade a taxa de variação é dada por:

$$d \frac{u_i^1}{dt} = f_i^1 = u_i^2 \quad (6)$$

$$d \frac{u_i^2}{dt} = f_i^2 + F_i^{Stoch} \quad (7)$$

A variação da coordenada espacial do agente equação (6), é definida como a velocidade individual do agente (segunda variável externa), e a variação da velocidade individual do agente, é obtida a partir das componentes estocásticas (F^{Stoch}) e determinísticas (f_i) sentidas por cada agente.

A energia interna do agente varia de acordo com um fluxo de energia interno ao agente, este é representado pela busca de energia do agente (alimentação) provinda de recursos do ambiente $q(\vec{r}_i, t)$, e o fluxo externo ao agente representando as mudanças no ambiente $p(\vec{r}_i, t)$ e ainda mais um termo dissipativo que representa a taxa metabólica $m_i(t)$. Assim, a variação da energia interna é dada por:

$$d \frac{u_i^4}{dt} = q(\vec{r}_i, t) - p(\vec{r}_i, t) - m_i(t) \quad (8)$$

No atual trabalho, será considerado somente o termo dissipativo da taxa metabólica, dessa forma, a equação 8 torna-se:

$$d \frac{u_i^4}{dt} = -m_i(t) \quad (9)$$

A equação (9) será aplicada somente em agentes da população de parasitas (Miracídeos e Cercárias), pois assume-se que o tempo de vida dos agentes da população de parasitas é muito menor que o tempo de vida da população de vetores. Dessa forma, o número de agentes

do tipo vetores permanece constante ao longo de cada simulação.

A informação interna do agente varia somente de acordo com as condições propícias ao contágio, esta variável tem como função construir uma área de influência numa dada região do ambiente, denominada de campo de informação (Figura 2). Este campo é construído a partir da contribuição individual de cada agente segundo o valor de sua informação interna atual. Dessa forma, quanto maior o número de agentes concentrados numa dada região do espaço, maior será o efeito deste campo sobre os demais agentes que compõem o sistema.

Na verdade, o campo de informação é a forma com que os agentes interagem e se comunicam. Este campo de informação pode ser modelado na forma da chance de contágio entre as populações vetores e parasitas. Admite-se que somente os moluscos não contaminados poderão se contaminar, e que estes produzirão regiões em que terão maiores chances de contágios para a população de Miracídeos. O mesmo ocorre com agente do tipo Humano e Cercárias.

Este campo de informação é um campo dinâmico que se modifica ao longo do tempo a partir das ações dos agentes. Sua dinâmica é definida como:

$$\partial \frac{h_{\theta}(\vec{r}, t)}{\partial t} = \sum_{i=1}^N s_i \delta_{\theta, \theta_i} \delta(\vec{r} - \vec{r}_i) - k_{\theta} h_{\theta}(\vec{r}, t) + D_{\theta} \Delta h_{\theta}(\vec{r}, t), \quad (10)$$

a equação acima descreve os seguintes fatos: a variação do campo é determinada pela informação individual de todos agentes que se encontram na posição considerada e este varia de acordo com o tempo (k_{θ}), pois os agentes se moverão no espaço, alterando as propriedades locais desse campo.

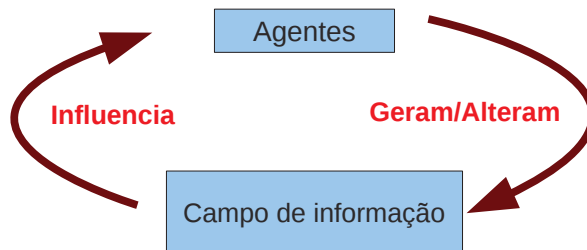


Figura2 : Diagrama apresentando a interação entre o campo de informação e os agentes.

O campo de informação no problema descrito, se deve à condição de contaminação dos parasitas relacionada ao contágio de cada agente não contaminado. De maneira geral, sua finalidade é a interação entre os indivíduos da população.

3.2 MODELAGEM DO AMBIENTE

Para que seja possível a construção do simulador é necessário, além da utilização da técnica de agentes brownianos, a construção de um modelo matemático onde será descrito como ocorrerá a interação entre os agentes e também como o ambiente modificará as variáveis (internas e observáveis) do agente. Ainda, a partir deste modelo será possível abstrair, as características e o comportamento das classes que representarão o problema no paradigma de Orientação a objetos [37]. Inicialmente, é necessário descrever como os agentes de diferentes populações irão interagir ao longo da simulação, bem como o ambiente será observado pelos agentes.

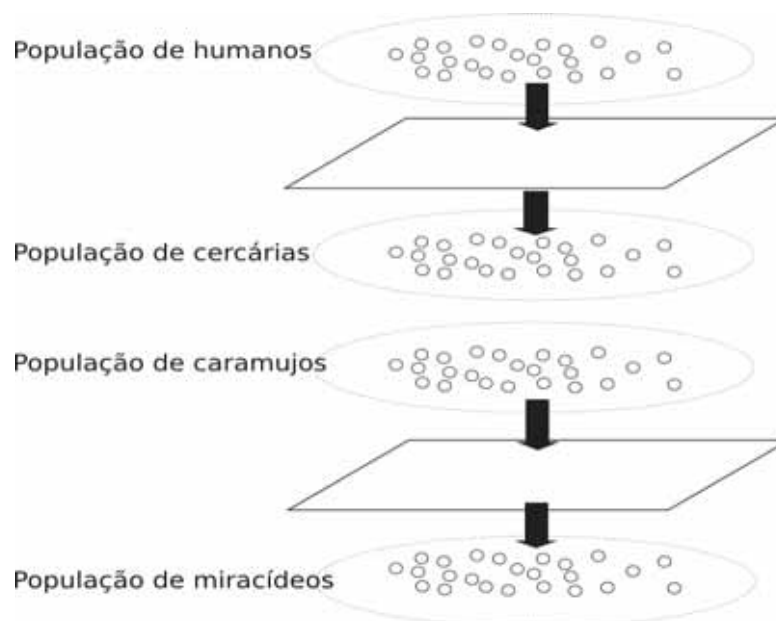


Figura3 : Dinâmica populacional. Na figura, a seta indica o sentido com que a informação flui entre as populações. Populações do tipo Vetor escreve no campo de informações, enquanto que as populações do tipo Parasita leem a informação transferida ao campo.

A interação entre os agentes das diferentes populações, condição necessária para o ciclo do contágio, ocorre a partir dos campos de informação. Uma modificação foi feita no modelo de agentes Brownianos para garantir que as diferentes populações se encontrem ao longo do espaço geográfico representando o contexto da dinâmica parasita – vetor. O campo de informações é acessível tanto aos agentes Vetores como aos agentes Parasitas. Contudo, para simular a necessidade da busca de um hospedeiro, os agentes Parasitas sentem a influência do campo, caracterizada como uma força determinística que os atrai até o(s) agente(s) Vetores. Por outro lado, os agentes Vetores são os geradores do campo de informações contribuindo com o valor do seu estado num dado ponto do espaço. Dessa forma, a dinâmica do campo de informações é definida pela escrita dos agentes Vetores que modificam o comportamento dos agentes Parasitas (Figura 3).

Outro fator importante para a descrição do problema ocorre com os recursos hídricos que são fatores cruciais para o habitat do caramujo *Biomphalaria*, essencial na dinâmica do contágio. Levando em consideração a técnica de agentes Brownianos, um recurso hídrico pode ser pensado como um potencial determinístico para os agentes que representarão os caramujos. Dessa forma, o recurso hídrico como um todo pôde ser modelado a partir de uma função matemática que descreve uma área de influência para os agentes caramujos. A função escolhida para efetuar a modelagem do ambiente é dada a seguir:

$$f(x, y, A) = \frac{A}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \quad (11)$$

3.3 MÉTODO NUMÉRICO APLICADO AO PROBLEMA.

As equações diferenciais apresentadas pelo modelo de agentes Brownianos constituem um desafio quanto sua resolução. Estas equações são a base do modelo adotado e por este motivo devem retornar um valor a cada passo da simulação.

Para a resolução dessas equações propõe-se uma técnica baseada em diferenças finitas

[38]. Assume-se que todas as derivadas a serem integradas, são na verdade diferenças finitas entre pontos na curva considerada. Para isso será levado em conta que o espaço e o tempo da simulação serão discretizados em unidades Δl e Δt respectivamente. Esta é a técnica apresentada na referência citada [39].

As equações apresentadas anteriormente podem agora ser avaliadas computacionalmente aplicando-se a referida técnica. Assim as equações (2) e (7) tornam-se agora:

$$\vec{r}_i(t+\Delta t) = \vec{r}_i(t) + \vec{v}_i(t) \Delta t \quad (12)$$

$$\vec{v}_i(t+\Delta t) = \vec{v}_i(t)[1 - \gamma_0 \Delta t] + \Delta t \alpha_i \nabla h_0(\vec{r}, t) + \sqrt{2 D_0 \Delta t} \text{GRND} \quad (13)$$

Observa-se na equação (13) a dependência de algumas constantes, entre elas e D_0 . A primeira relaciona-se a efeitos dissipativos do meio onde o agente está submetido e a segunda, ao coeficiente difusivo discutido anteriormente.

A equação (13) demonstra que o meio exerce uma resistência no deslocamento do agente, essa resistência é responsável pelo decréscimo na energia interna do mesmo. Também mostra que o campo de informação gerado pelos demais agentes é o seu principal motivador. Na verdade o gradiente do campo está associado à força externa sobre o movimento do agente. E por fim, o movimento sofre influências estocásticas representadas pelo último termo da equação onde GRND é um número aleatório proveniente de uma distribuição normal.

Quanto ao campo de informações, este pode ser representado a seguir:

$$h(\vec{r}, t + \Delta t) = (1 - k_0 \Delta t) h_0(\vec{r}, t) + \Delta t s_0 N_0(\vec{r}, t) - D_0 \frac{\Delta t}{(\Delta l)^2} w(x, y, t), \quad (14)$$

$$w(x, y, t) = 4h_0(\vec{r}, t) - h_0(x + \Delta l, y, t) - h_0(x - \Delta l, y, t) - h_0(x, y + \Delta l, t) - h_0(x, y - \Delta l, t).$$

É necessário informar que para encontrar soluções estáveis, os seguintes requisitos devem ser satisfeitos,

$$k_0 \Delta t \ll 1, \quad (15)$$

e

$$D_0 \frac{\Delta t}{(\Delta l)^2} < \frac{1}{4}, \quad (16)$$

caso contrário, a técnica de simulação estocástica discutida divergirá dos valores esperados.

A modelagem do sistema ocorrerá com base na técnica de multiagentes descrita na sessão anterior, partindo de algumas hipóteses que resultarão nos campos de informações, construído para as diferentes populações em questão.

Cada agente presente no programa deve se comportar como um agente do tipo agente browniano, dessa forma, cada população presente no programa deve apresentar um campo de informações associado ao seu comportamento específico. Será adotado o seguinte critério na construção do comportamento individual de cada agente: todo agente apresenta um comportamento determinístico, baseado em fatores que influenciam o meio sobre o qual está inserido, no caso de populações biológicas, este campo se apresenta na forma de chances de contágio por parte dos indivíduos parasitas; já no caso de populações humanas, o campo de informação refere-se às necessidades biológicas que levam o indivíduo humano à um ambiente de possível contágio.

4. MODELAGEM COMPUTACIONAL E SIMULAÇÃO.

O desenvolvimento de um simulador envolvendo os conceitos anteriormente expostos demonstra ser uma tarefa de grande complexidade, de modo que o sucesso do projeto não se limita somente na implementação. Dessa forma, é necessário a adoção de uma metodologia de desenvolvimento de software que garanta o objetivo desse trabalho. Este capítulo descreve como a metodologia de multiagentes brownianos, já apresentada, pôde ser implementada computacionalmente a partir da metodologia de desenvolvimento chamada de processo unificado.

4.1 A METODOLOGIA DE DESENVOLVIMENTO

A metodologia baseada em processo unificado (PU) [40] pressupõe a existência de fases determinantes no processo de concepção do sistema mencionado. Esta metodologia é um processo iterativo e evolutivo para desenvolvimento de sistemas orientados a objeto. As fases iterativas se baseiam em 4 etapas: concepção, elaboração, construção e transição [41].

A seguir, cada fase do PU será apresentada e também será descrito como ocorreu sua adoção para a conclusão do trabalho.

Na fase de concepção são analisados os requisitos funcionais e não funcionais, ou seja, quais ações e comportamentos globais o sistema deve apresentar. No caso do simulador, dois requisitos funcionais, relacionados ao usuário, são descritos a seguir: um usuário deve solicitar os dados de entrada, e solicitar a inicialização do simulador. As ações em nível computacional que representam o modelo matemático e a metodologia anteriormente descrita, também são requisitos funcionais do sistema. Em particular, o diagrama UML concebido

nesta fase é o diagrama de casos de uso [40], que não será descrito neste trabalho. Como requisito não funcional, é desejável que o simulador tenha disponibilidade para usuário utilizá-lo a partir da WEB.

Na fase de elaboração, são elaboradas a modelagem estática e dinâmica do sistema a partir dos requisitos levantados na fase anterior. Os principais diagramas de interesse, apresentados nas sessões a seguir, são os diagramas de sequência e o diagrama de classes. Nesta fase, são apresentadas as características e comportamentos dos agentes que descrevem o problema e as relações estabelecidas entre si. Outro aspecto chave relevante nesta fase, se deve à elaboração da arquitetura do simulador.

A fase de construção não é apresentada no trabalho, pois constitui a implementação dos demais requisitos.

Por fim, a fase de transição constitui-se em testes e implantação do sistema. Neste trabalho, serão apresentados os resultados obtidos de modo a garantir a validação do simulador.

4.2. TECNOLOGIAS E FRAMEWORKS UTILIZADOS

Com base na complexidade do sistema a ser concebido, o paradigma de programação orientada a objetos foi escolhido por apresentar a capacidade de representar naturalmente objetos e ambientes do mundo real em sua forma computacional. No paradigma de programação orientada a objetos, o programador desenvolve o código a partir da criação de classes, cujas instâncias chamadas objetos, apresentam métodos (ações) e atributos (características). Além disso, possui diversas características que incentivam a reutilização do código, como encapsulamento, herança e polimorfismo. Outra vantagem se deve ao fato de que os objetos podem se relacionar a partir da troca de mensagens a partir de métodos [42].

A linguagem de programação utilizada para desenvolvimento é a linguagem de programação JAVA, por ser uma linguagem orientada a objetos, independente de plataforma e apresentar suporte à tecnologias WEB [43].

A seguir serão descritas as tecnologias que foram utilizadas na construção do simulador.

- **JFreeChart**

É esperado que o sistema gere, ao final de cada simulação, os dados obtidos na forma de gráficos e tabelas, a partir dos parâmetros de entrada utilizados. Para este fim, será utilizado o Framework JFreeChart [44], cujo propósito é a apresentação de dados na forma de elementos gráficos.

O JFreeChart é uma biblioteca livre para construção de gráficos disponível para plataforma Java. Pode ser utilizada para aplicações Desktop, Applets, Servlets e JSP. Pode gerar gráficos do tipo pizza, barras, linhas, pontos, séries temporais, entre outros.

Entre seus recursos disponíveis destaca-se: exportação de arquivos de imagens no formato PNG e JPEG, exportação de gráficos em PDF e SVG, suporte para HTML, é escrita inteiramente na linguagem JAVA, necessitando da plataforma JDK 1.3.1 ou mais recente.

- **A tecnologia JSP**

Para que o requisito não funcional descrito levantado na fase de concepção do PU (aplicação WEB) seja satisfeito, é necessário a adoção de uma tecnologia compatível com aplicações Java e com suporte à WEB.

A tecnologia JSP foi concebida a partir da necessidade das aplicações que fazem uso de requisitos não funcionais associados à tecnologia WEB. Esta tecnologia é uma das várias API's presentes no Java EE [45]. O Java EE (Java Enterprise Edition) consiste de uma série de especificações bem detalhadas, dando uma receita de como deve ser implementado um software que faz cada um desses serviços de infraestrutura. Possui várias especificações, entre elas, algumas específicas para lidar com o desenvolvimento de uma aplicação Web:

- JSP
- Servlets
- JSTL
- JSF

Na plataforma Java, a primeira e principal tecnologia capaz de gerar páginas dinâmicas são as classes e interfaces denominadas Servlets, que são responsáveis pela criação

de páginas WEB dinâmicas a partir da linguagem JAVA. As Servlets apresentam a capacidade de processar requisições a partir do protocolo HTTP e responder na forma de redirecionamentos ou mesmo na renderização de páginas HTML [45].

Para a execução da tecnologia JSP e das Servlets, torna-se necessário a utilização de um Servlet Container, servidor que suporta essas funcionalidades, mas não necessariamente o Java EE completo. É indicado para aplicações que não necessitam do Java EE completo, mas somente das especificações web.

Há alguns servlet containers famosos no mercado. O container utilizado neste trabalho é o Apache Tomcat [46].

4.3. A ARQUITETURA LÓGICA DO SIMULADOR

A modelagem do simulador segue o padrão de projetos MVC - (Model View e Controller) [40], a arquitetura do sistema se baseia no modelo de três camadas que divide o aplicativo de modo que os componentes específicos residam nas três camadas citadas.

Na arquitetura MVC o modelo representa os dados da aplicação e as regras que governam o acesso e a modificação dos dados, também fornece ao controlador a capacidade de acessar as funcionalidades da aplicação encapsuladas pelo próprio modelo. No simulador, a camada de modelo é constituída pelas diversas classes que modelam o problema e a técnica de multiagentes brownianos.

O componente de visualização renderiza o conteúdo de uma parte particular do modelo e encaminha para o controlador as ações do usuário; acessa também os dados do modelo via controlador e define como esses dados devem ser apresentados. No simulador, a camada de visualização é implementada por páginas JSP e classes Servlets [43].

Um controlador define o comportamento da aplicação, é ele que interpreta as ações do usuário e as mapeia para chamadas do modelo. As ações realizadas pelo modelo incluem ativar processos ou alterar o estado do modelo. Com base na ação do usuário e no resultado do processamento do modelo, o controlador seleciona uma visualização a ser exibida como

parte da resposta da solicitação do usuário.

A arquitetura MVC é estruturada a partir dos seguintes elementos:

- Model (Camada de lógica da Aplicação) - Esta camada se preocupa apenas com o armazenamento, manipulação e geração de dados. Responsável pela simulação de cenários e do comportamento dos agentes que estarão interagindo no ambiente geográfico.
- Camada de Controle - determina o fluxo da apresentação servindo como uma camada intermediária entre a camada de visualização e a lógica de aplicação.
- Camada de Visualização – construída na plataforma WEB a partir da tecnologia JSP, renderiza as páginas na linguagem HTML apresentando os dados calculados pelo simulador, na forma de imagens. Esta camada também é responsável pela renderização dos gráficos na forma de imagens, valendo-se do Framework JFreeChart.

A justificativa do uso do modelo MVC para este trabalho reside na necessidade de desacoplar a lógica de aplicação fazendo o papel do simulador das diversas formas de visualização. Isso significa que outras aplicações, não necessariamente apresentando plataforma WEB, podem fazer uso das classes do modelo para simular outros cenários e até mesmo problemas, onde seja possível a aplicação da técnica de agentes brownianos.

O digrama de pacotes (Figura 3) ajuda na compreensão da arquitetura lógica do sistema, destacando a dependência entre determinados pacotes e classes. As três camadas do sistema são construídas a partir dos pacotes “model”, “view” e “control”. A dependência entre os pacotes se deve somente no pacote “control”. Os pacotes “model” e “view” são totalmente independentes entre si.

O pacote “model” é composto por outros pacotes internos com responsabilidade voltada para a representação do problema. Neste pacote são encontrados as classes que representam o ambiente, os agentes, os campos de informações. Ainda existem classes, presentes no pacote “engine”, voltadas para integração das equações diferenciais e modelagem do comportamento dos agentes. Outro pacote de interesse é o pacote “statistics”, responsável pelo tratamento estatístico dos dados calculados pelo simulador (Figura 4).

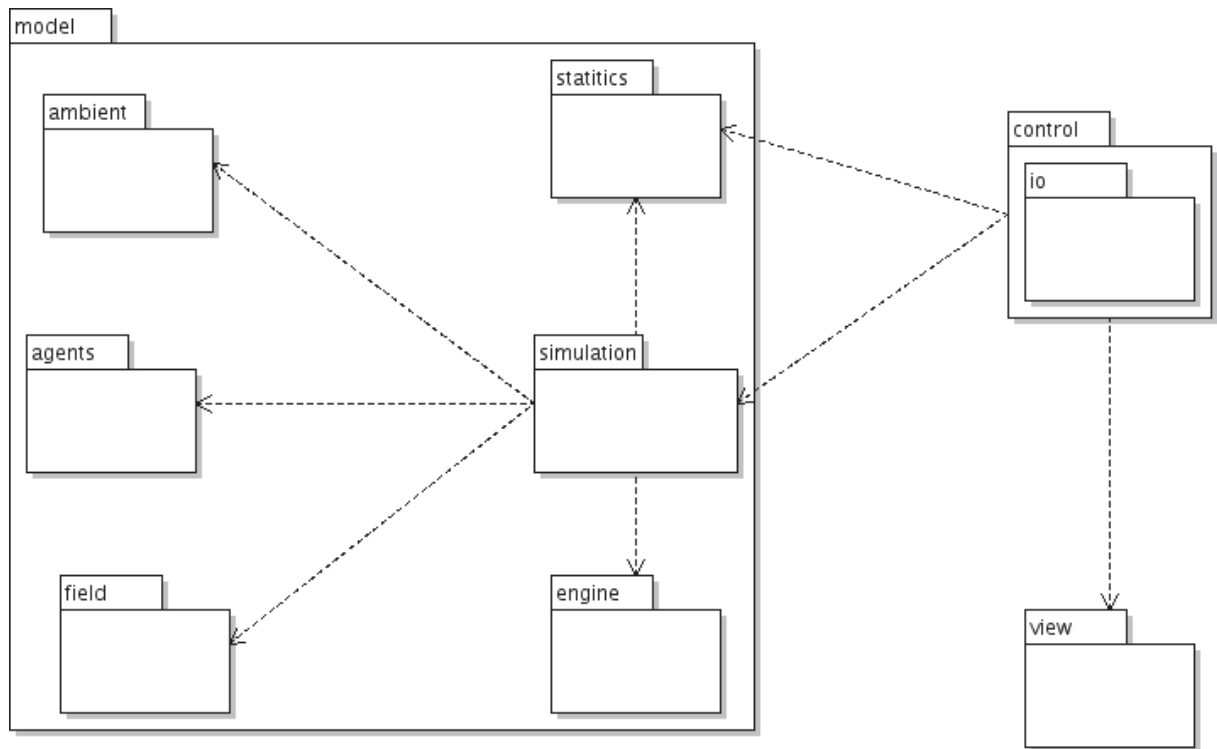


Figura 4 : Diagrama apresentando a relação e a dependência entre diferentes pacotes que foram construídos para a modelagem do problema.

O pacote “*control*” tem ainda a responsabilidade do tratamento dos dados de entrada (pacote “*io*”). Os detalhes das classes que compõem cada um dos pacotes serão apresentados na próxima sessão do trabalho, junto aos diagramas de classes de cada um.

4.4 MODELAGEM CONCEITUAL DO PROBLEMA

A modelagem estática se destina a descrever como os elementos computacionais que constituem as classes no paradigma orientado a objetos se relacionam, a descrição do simulador a partir deste enfoque se dá a partir do diagrama de classes UML (Figura 4).

O processo de modelagem do simulador deve levar em contas os elementos que compõem o problema a ser representado, dessa forma, deve-se atentar-se em representar de maneira coerente os processos e condição sob os quais os indivíduos da simulação reagirão ou

se modificarão. Para este fim, as características que diferencia os indivíduos bem como seu comportamento definirão os atributos e os métodos das classes que os representarão.

Para o problema em específico, as classes do sistema corresponderão aos agentes da simulação multiagente e o seu ambiente de interação. A descrição das classes a partir do digrama de classes é dada a seguir na Figura 5.

Descrição das classes e dos agentes.

A seguir apresenta-se uma descrição das classes que representarão os agentes ao longo das simulações. Estas classes são encontradas no pacote “*model.agents*”.

- **Classe *BrownianAgent***

É a classe que implementa a metodologia de agentes Brownianos, tem como atributos o conjunto de variáveis de estado (informação interna e energia) e variáveis externas observáveis (posição e velocidade), determinantes para a representação do problema.

- **Interface *VectorAgent***

Apresenta o método que constrói novos objetos das classes *Cercarie* ou *Miracidea*. Este método é executado se o estado interno do agente da classe *Human* ou *Biomphalaria* é igual a zero. É um método implementado pelas classes *Human* ou *Biomphalaria* a partir de Polimorfismo. Em particular, segundo o ciclo de vida do *Shistosoma mansoni*, agentes humanos devem gerar agentes *Miracideas*, se na presença de recurso hídrico, e agentes *Biomphalaria* devem produzir agentes *Cercarie*.

- **Interface *ParasiteAgent***

Apresenta dois métodos polimórficos que serão implementados pelas classes que representam os parasitas do modelo (*Cercarie* e *Miracidea*). O método “*hasAgentNear*” tem a função de computar durante um dado passo de simulação se existe, ou não, algum agente na proximidade. Este método, por ser um método polimórfico, apresenta ações diferentes para as duas classes. Em particular, é necessário que os agentes da classe “*Miracidea*”, possam visualizar somente os agentes da classe “*Biomphalaria*”, e que agentes da classe “*Cercarie*”,

possam visualizar somente agentes da classe “Human”. O outro método denominado “contaminate” faz com que o agente previamente selecionado pelo método “hasAgentNear” torne-se contaminado, modificando sua informação interna (o atributo “state”) para o valor de uma unidade.

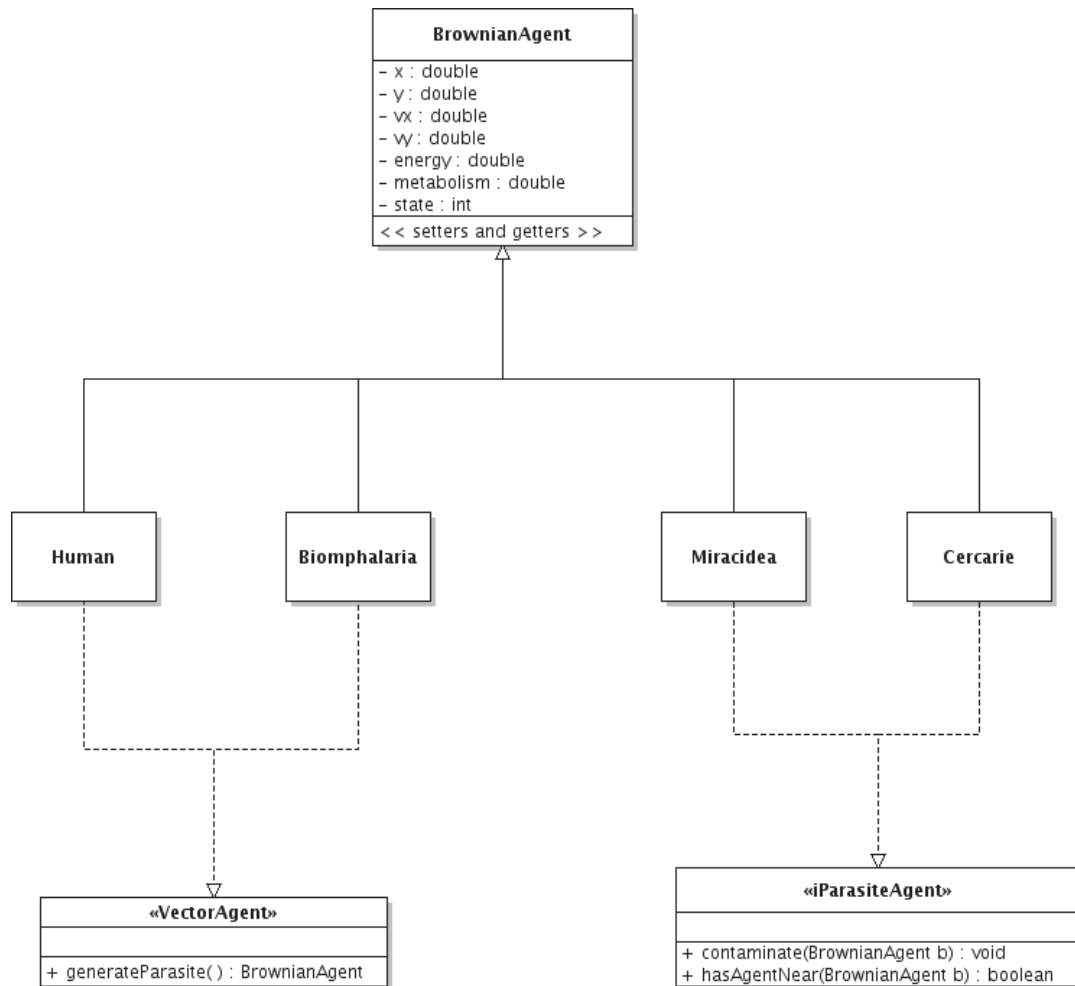


Figura 5 : Diagrama apresentando a relação entre as classes que definem os agentes modelados presentes no pacote `model.agents` do simulador.

- **Classe Human e Classe Biomphalaria**

Estas classes representam os agentes que modelam os vetores necessários para o ciclo de vida do *Schistosoma*.

- **Classe Cercarie e Classe Miracidea**

Representam uma população de agentes parasitas (Cercárias e Miracídeos).

Para o ambiente de interação dos agentes são utilizadas as classes “*GeographicalRegion*” e “*WaterResource*”, encontradas no pacote “*model.ambient*”. A modelagem computacional do ambiente requer que o ambiente geográfico apresente recursos hídricos onde as espécies de caramujos *Biomphalaria*, e os parasitas Cercárias e Miracídeos devem ser encontrados. Dessa forma, esta classe é composta por outra classe denominada “*WaterResource*” que representa um recurso hídrico presente, como por exemplo, açudes, lagos, água empoçada, etc [6, 7].

Por hipóteses simplificadoras, admite-se que estes recursos apresentam geometria circular ou elíptica, dessa forma, seus atributos “*length*” e “*width*” representariam os semi-eixos de uma forma elíptica.

No modelo, os agentes das classes “*Biomphalaria*”, “*Miracidea*” e “*Cercarie*”, sentem a influência do recurso hídrico na forma de um potencial atrativo, que direciona um dado agente das referidas classes para o interior do recurso hídrico. Este potencial, já discutido no trabalho, tem sua intensidade associada ao atributo “*depth*” da classe “*WaterResource*”, dessa forma, assume-se que quanto mais profundo o recurso, maior é a força determinística que influencia o comportamento do agente.

É pelo fato acima descrito, que a classe “*GeographicalRegion*”, apresenta o método chamado “*fx*”, que calcula a partir da equação (verificar na sessão Modelagem matemática) o valor total resultante de todos os recursos hídricos da região geográfica, num dado ponto de coordenadas *x* e *y*. Este valor, representado na forma de um potencial, será usado pelos métodos “*gradientX*” e “*gradientY*” para o cálculo das componentes da força determinística.

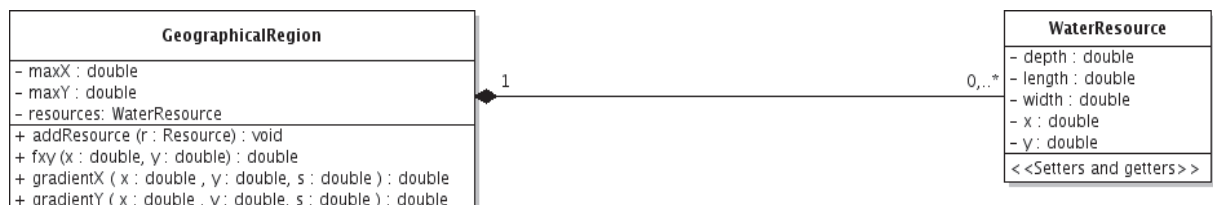


Figura 6 : Diagrama apresentando a a relação entre as classes que definem o ambiente geográfico do simulador.

Em futuras modelagens será assumido a existências de outros recursos presentes no ambiente, motivadores para populações humanas, tais como habitações, estabelecimentos variados, hospitais, etc.

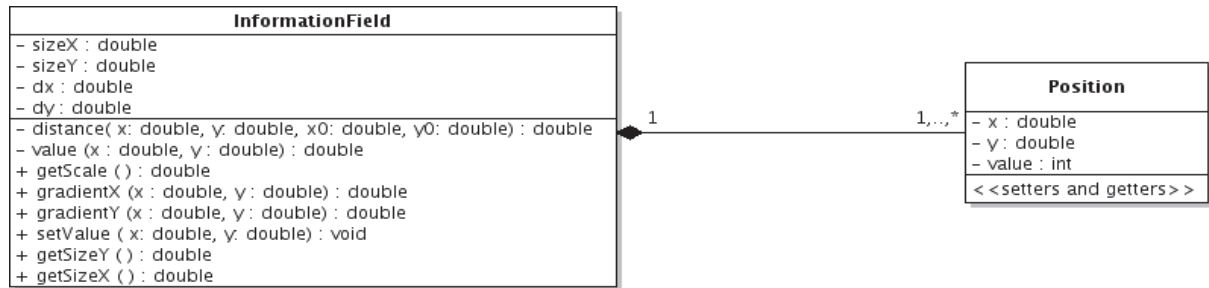


Figura 7 : Diagrama apresentando a a relação entre as classes que definem o campo de informações, aspecto chave para a modelagem do problema.

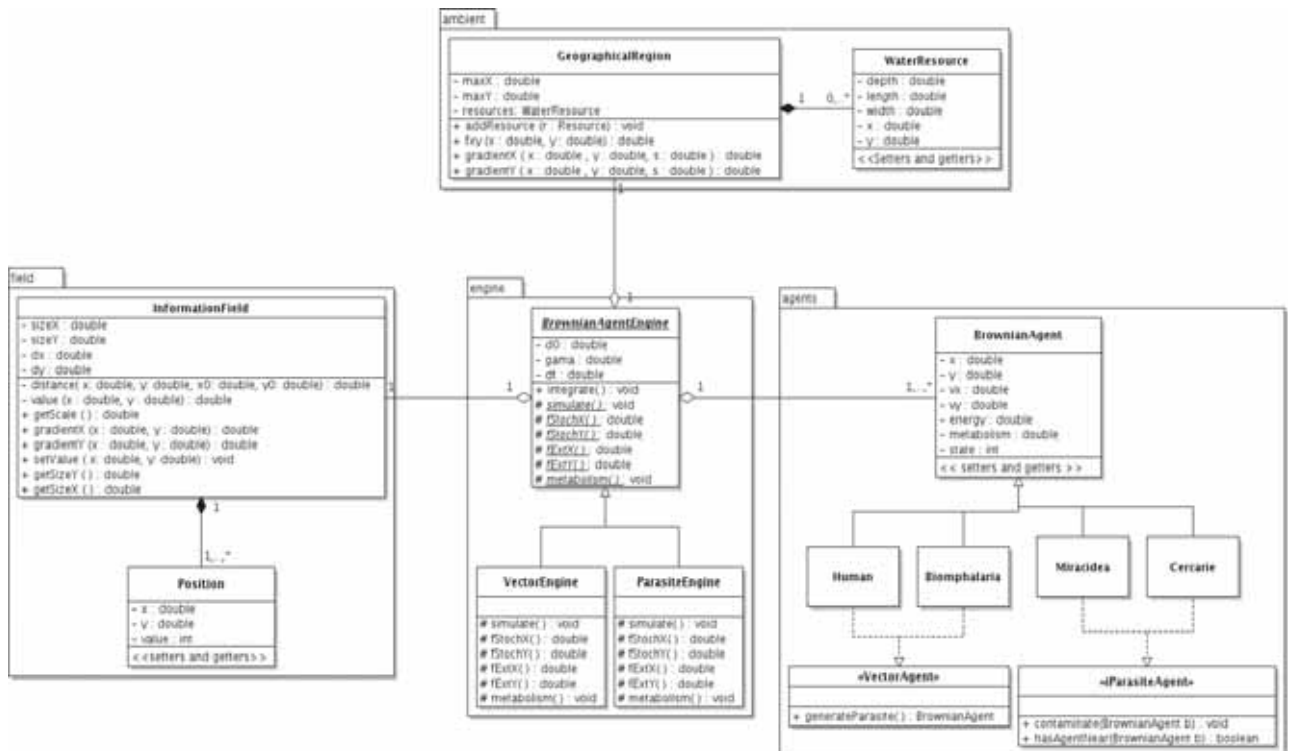


Figura 8 : Diagrama apresentando a a relação entre as classes presentes nos diferentes pacotes apresentados anteriormente.

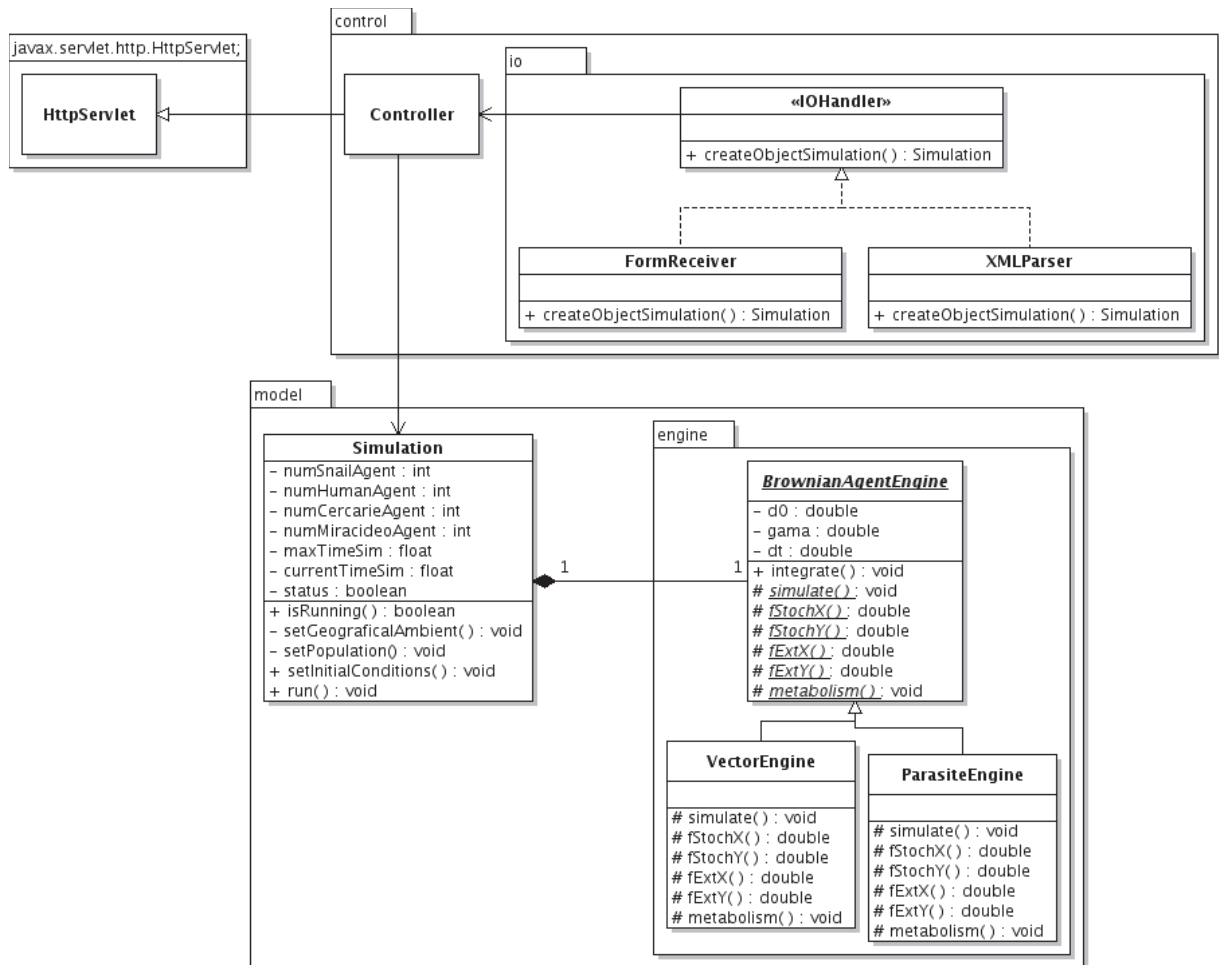


Figura 9 : Diagrama apresentando a relação entre as camadas implementadas a partir do modelo MVC.

Por fim, apresenta-se as classes do pacote “*model.field*” onde o campo de informações, elemento chave para a implementação da técnica de agentes brownianos é apresentado.

O campo de informações é um campo dinâmico que se modifica de acordo com a posição e a informação interna dos agentes. Basicamente, é composto pela classe de objetos “*Position*”, que representa os pontos de função de duas variáveis, por isso, a existência dos atributos “*x*”, “*y*” e “*value*”, e pelos atributos “*sizeX*”, “*sizeY*”, “*dx*” e “*dy*”. Os dois primeiros atributos fazem referência do tamanho da região geográfica, pois assume-se que este é o tamanho máximo do campo de informações. Os demais atributos, “*dx*” e “*dy*”, referem-se à escala usada na simulação, ou seja, definem qual é a menor porção possível de medida tanto

para a região geográfica, quanto para o campo de informações. Em particular, estes parâmetros são necessários para o cálculo das componentes do gradiente.

O cálculo das componentes do gradiente ocorre de forma análoga ao cálculo computado na classe “*GeographicalRegion*”, contudo o valor do potencial calculado na posição x e y é encontrado a partir da somatória das contribuições do valor da informação interna de todos os agentes das classes que representam os vetores da doença (*Biomphalaria* e *Human*), este valor é obtido a partir do método privado “*value*”

4.5. FUNCIONAMENTO DO SIMULADOR

O usuário entra com os dados, a partir de um arquivo XML, no simulador. Estes dados são enviados para uma classe Java que ajusta os parâmetros de entrada da simulação. Em geral, os parâmetros ajustados pelo usuário serão agrupados em três categorias de acordo com sua necessidade:

- Dados da região geográfica.
- Dados demográficos das populações.
- Dados do tempo.

Os dados demográficos se baseiam na densidade populacional da região considerada levando-se em conta o tamanho da população de humanos e de caramujos do gênero *Biomphalaria* que habitam a região bem como sua distribuição no espaço geográfico, com isso, defini-se uma densidade populacional em torno de alguns pontos espaciais que representam, por exemplo, a ocupação habitacional da região.

O outro grupo de parâmetros de entrada se baseia nas dimensões da região geográfica e na escala usada durante a simulação. Ainda, o usuário ajusta os parâmetros associados aos recursos hídricos, entre estes parâmetros, estão a extensão do recurso, a profundidade média e localidade no espaço considerado. Por fim, o último grupo de parâmetros será associado ao tempo da simulação, por exemplo, duração da simulação.

A seguir é apresentado um exemplo do arquivo de configuração que o usuário envia

para o sistema, destacando-se os principais parâmetros que o usuário deverá ajustar:

```
<?xml version="1.0" encoding="UTF-8"?>
<simulation>
  <parameters>
    <maxTimeSim>3000</maxTimeSim>
    <maxXSize>2000</maxXSize>
    <maxYSize>2000</maxYSize>
  </parameters>
  <population type="vector">
    <numSnailAgent>3000</numSnailAgent>
    <numHumanAgent>300</numHumanAgent>
  </population>
  <population type="parasite">
    <numCercarieAgent>100</numCercarieAgent>
    <numMiracideaAgent>300</numMiracideaAgent>
  </population>
  <ambient>
    <water-resource>
      <length>20</length>
      <depth>100</depth>
      <width>200</width>
      <xposition>100</xposition>
      <yposition>200</yposition>
    </water-resource>
    <water-resource>
      <length>20</length>
      <depth>300</depth>
      <width>200</width>
      <xposition>200</xposition>
      <yposition>400</yposition>
    </water-resource>
  </ambient>
</simulation>
```

```

        </water-resource>
    </ambient>
    <information-field>
        <cercarie-field dx="1" dy="1"> </cercarie-field>
        <miracidea-field dx="1" dy="1"> </miracidea-field>
    </information-field>
</simulation>

```

A execução dos processos de configuração, a partir do usuário, e a inicialização da simulação se dá a seguir pelo diagrama de sequência descrito na figura abaixo:

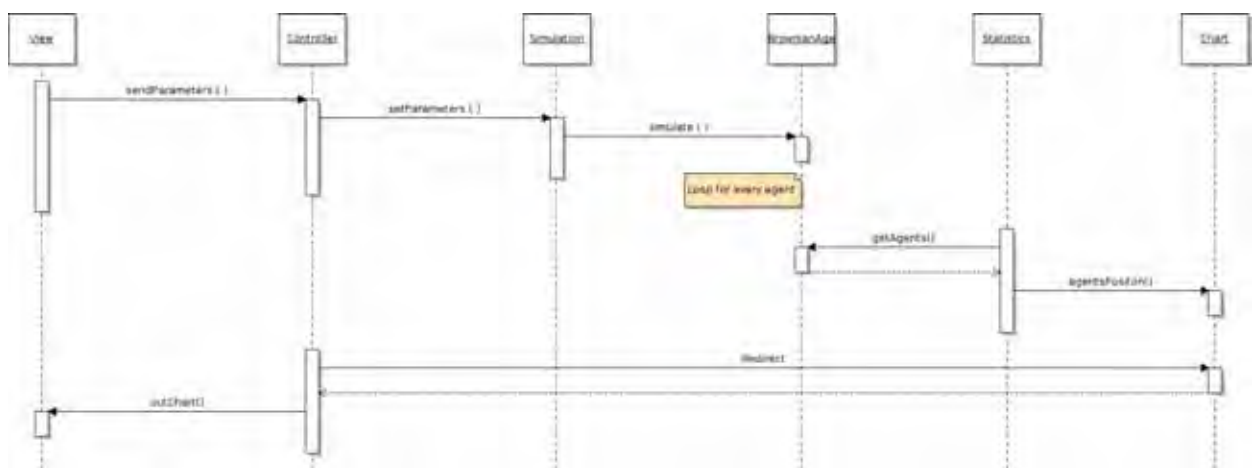


Figura 10 : Diagrama de sequência apresentando o fluxo principal do simulador

Durante a simulação uma lista de todos os agentes é mantida em memória e deve ser percorrida até que todos os agentes que estão ativos na simulação sejam modificados pelas classes “VectorAgentEngine” ou “ParasiteAgentEngine”.

Cada passo de simulação é dividido em três fases principais:

1. Atualização dos campos de informação: a lista de agentes é percorrida para cada agente das classes de vetores (Human ou Biomphalaria), o campo associado é atualizado, ou seja, o valor associado às novas posições dos agentes vetores é alterado de acordo com os eventos que ocorreram;

2. Proliferação e contaminação: nessa fase a lista de agentes das classes de parasitas

(Cercarie ou Miracidea) é novamente percorrido. A contaminação ocorre da seguinte forma: se o agente for do tipo miracídeo ou cercária, e houver um humano ou caramujo respectivamente, em um determinado raio, esses agentes são contaminados (Figura 5). Uma vez contaminados, tais agentes não são curados (o modelo ainda não inclui esse comportamento). Já a proliferação se dá a seguir: todo agente do tipo humano, que se encontra contaminado, gera um novo agente miracídio, que será adicionado ao final da lista de agentes em memória, o mesmo ocorre com os agentes caramujos que estão contaminados, contudo o novo agente criado é um agente cercária.

3. Atualização das posições dos agentes: a posição, velocidade e energia de todos os agentes no sistema são atualizados. Caso a energia de um determinado agente se esgote, o respectivo agente é removido da simulação.

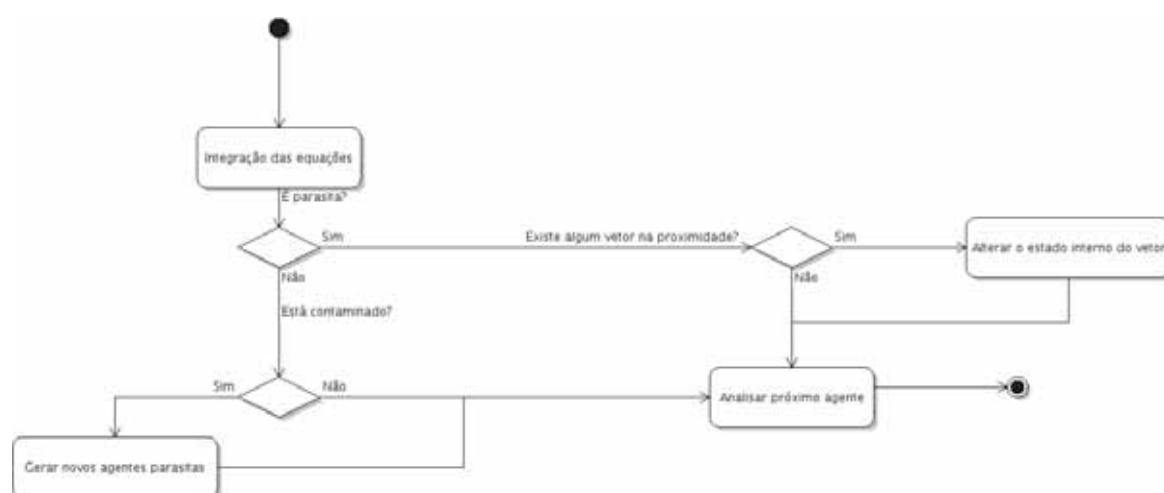


Figura 11 : Diagrama de estados apresentando o fluxo proliferação e contágio.

A simulação consiste na execução das fases 1, 2 e 3, nessa ordem, para um número pré-determinado de passos de simulação (Figura 12).

A fase de execução do simulador ocorre de maneira mais complexa que a fase descrita anteriormente. A figura (Diagrama de atividade) apresenta como ocorre os principais cenários da simulação.

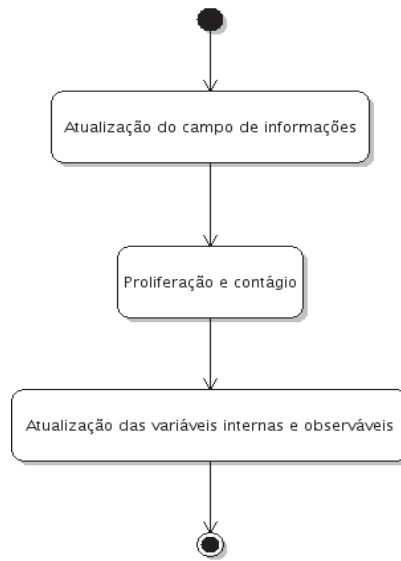


Figura 12 : Diagrama de estados apresentando o fluxo principal de simulação.

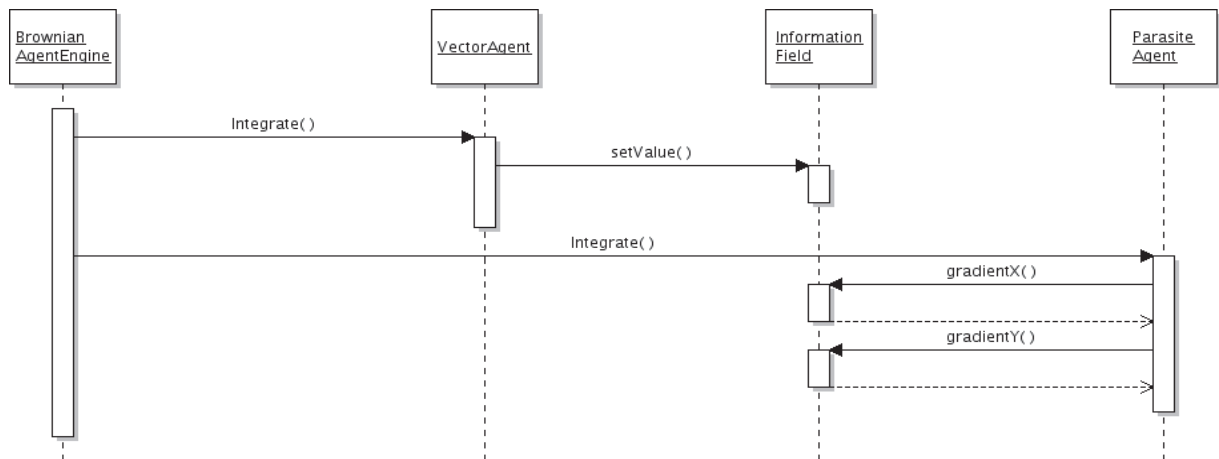


Figura 13 : Diagrama de sequência apresentando o fluxo principal de simulação.

Inicialmente pressupõe-se a existência de pelo menos um agente humano infectado pelo parasita *Schistosoma* libera suas fezes próximas a um recurso hídrico. Do ponto de vista computacional, quando a posição de um agente humano é domínio da função que descreve um recurso hídrico, novos objetos da classe Parasita são criados (Figuras 11 e 13).

Por serem agentes Brownianos, os objetos da classe Parasitas modificam sua dinâmica

ao serem influenciados pelo campo de informações gerado pela população de agentes da classe Vetores, isso ocorre pois o campo de informações gera uma força determinística atrativa que direciona os agentes contaminantes (Parasitas) para os agentes da classe vetores (Figura 13).

Se os agentes Parasitas conseguirem atingir agentes da classe Vetores, o contágio ocorrerá o que desencadeará a criação de novos objetos da classe Parasitas e mudará o atributo “estado” do objeto Vetor. Os objetos do tipo Vetores somente contribuirão para o campo de informações se o valor do atributo “estado” for igual a um. Dessa forma, este objeto já contaminado não influencia para mais o campo de informações acessível aos agentes da classe Parasita, o que garante que os agentes Parasitas sempre visem a busca de diferentes indivíduos da população Vetores. O diagrama de estados da figura 11 ilustra o processo de contágio simulado.

Depois de executada a simulação, os dados de cada passo da simulação são enviados para o Framework JFreeChart para posterior renderização e por fim os resultados são impressos numa página HTML a partir da tecnologia JSP.

5. RESULTADOS E DISCUSSÃO.

Os resultados apresentados nesta seção foram obtidos a partir de diferentes parâmetros de entrada. Uma série de 3 cenários de simulações distintos foram executadas e os resultados obtidos serão discutidos mais adiante neste capítulo. Estas simulações têm por objetivo, a validação do modelo proposto, demonstrando que este modelo representa de maneira adequada o problema do contágio.

Os cenários simulados serão divididos em duas categorias distintas: cenários de ambiente, onde o ambiente geográfico é alterado em cada um dos cenários, mas as populações de vetores permanecem constantes, e cenários populacionais, onde as populações, em particular do tipo humana é variada, mas o ambiente se mantém.

A tabela a seguir demonstra a nomenclatura que será utilizada para identificar cada um dos resultados obtidos dos cenários:

Cenários analisando o o ambiente.	Simulação 1.1
	Simulação 1.2
	Simulação 1.3
Cenário analisando fatores populacionais (vetores humanos)	Simulação 2.1
	Simulação 2.2
	Simulação 2.3
	Simulação 2.4
Cenário fatores populacionais (vetores caramujos)	Simulação 3.1
	Simulação 3.2
	Simulação 3.3
	Simulação 3.4

Tabela 1: Tabela de simulações e cenários

Alguns parâmetros serão mantidos em todos os cenários apresentados a seguir, estes parâmetros dizem respeito às variáveis utilizadas pela classe “*BrownianAgentEngine*”, responsável pela integração das equações já discutidas. O valor do parâmetro do coeficiente de difusão do meio (D_0), associado ao comportamento estocástico, é de 0.75 e o parâmetro viscosidade do meio, associado à resistência sobre o movimento estabelecido pelo meio (γ) é de 0.95.

Devido à aplicação do método numérico, o tempo foi discretizado em unidades de tamanho unitário. Outros parâmetros envolvendo a região geográfica e conseqüentemente o campo de informações definem as unidades em que o espaço é discretizado (dx e dy), ambos adotados como uma unidade.

Ambos conjuntos de parâmetros foram utilizados nos trabalhos da referência [34].

5.1 ANÁLISE DA INFLUÊNCIA DO AMBIENTE SOBRE O CONTÁGIO.

Os estudos descritos nesta sessão, ocorreram a partir da modificação do ambiente, alterando o número de recursos hídricos e as propriedades de cada um desses recursos. As simulações foram executadas utilizando-se inicialmente 3300 agentes, sendo 3000 agentes da classe *Biomphalaria* e 300 agentes da classe *Human*. Todas as simulações para este cenário foram executadas no tempo máximo de 3000 passos de simulação.

Estes parâmetros foram escolhidos pois se demonstraram resultados viáveis em ensaios anteriores.

De início, para a primeira simulação (Simulação 1.1), serão utilizados os parâmetros passados para a execução do problema dados no arquivo de configuração descrito abaixo:

```
<?xml version="1.0" encoding="UTF-8"?>
<simulation>
  <parameters>
```

```

    <maxTimeSim>3000</maxTimeSim>
    <maxXSize>1000</maxXSize>
    <maxYSize>1000</maxYSize>
</parameters>
<population type="vector">
    <numSnailAgent>3000</numSnailAgent>
    <numHumanAgent>300</numHumanAgent>
</population>
<population type="parasite">
    <numCercarieAgent>1000</numCercarieAgent>
    <numMiracideaAgent>1000</numMiracideaAgent>
</population>
<ambient>
    <water-resource>
        <length>100</length>
        <depth>2.0</depth>
        <width>100</width>
        <xposition>200</xposition>
        <yposition>200</yposition>
    </water-resource>
</ambient>
<information-field>
    <cercarie-field dx="1" dy="1"> </cercarie-field>
    <miracidea-field dx="1" dy="1"> </miracidea-field>
</information-field>
</simulation>

```

Os resultados apresentados nessa seção demonstraram a dispersão da informação ao longo das populações de agentes humanos e caramujos.

A simulação inicia-se com as diversas populações de agentes dispostas ao longo do espaço geográfico (Figura 14), os agentes do tipo humanos (em vermelho) estarão dispostos

aleatoriamente, simulando uma ocupação desordenada, geralmente característica em regiões de baixo saneamento. As demais populações que dependem do recurso hídrico, são alocadas ao longo das dimensões do recurso hídrico, por isso inicialmente existe uma alta concentração de agentes caramujos (em azul), numa dada região do espaço. Um parasita torna os agentes sadios contaminados, se caso as condições de contágios, já discutidas, forem satisfeitas (verde para humanos infectados e amarelo para caramujos infectados).

Ainda nos primeiros estágios de simulação (Fig. 3), foi possível observar um aumento substancial no número de agentes caramujos (linha azul no gráfico ao lado). Isso se deve ao aparecimento de um padrão difusivo da informação, ou seja, os agentes parasitas que encontrarem vetores não contaminados, modificarão o estado dos vetores. Por isso, existe um elevado número de agentes contaminados já nos passos iniciais. É certo que por se tratar de um sistema dinâmico, existirá agentes que se não se encontrarão nas condições propícias para contágio, por isso observa-se um decrescimento da taxa com que os agentes se tornam contaminados.

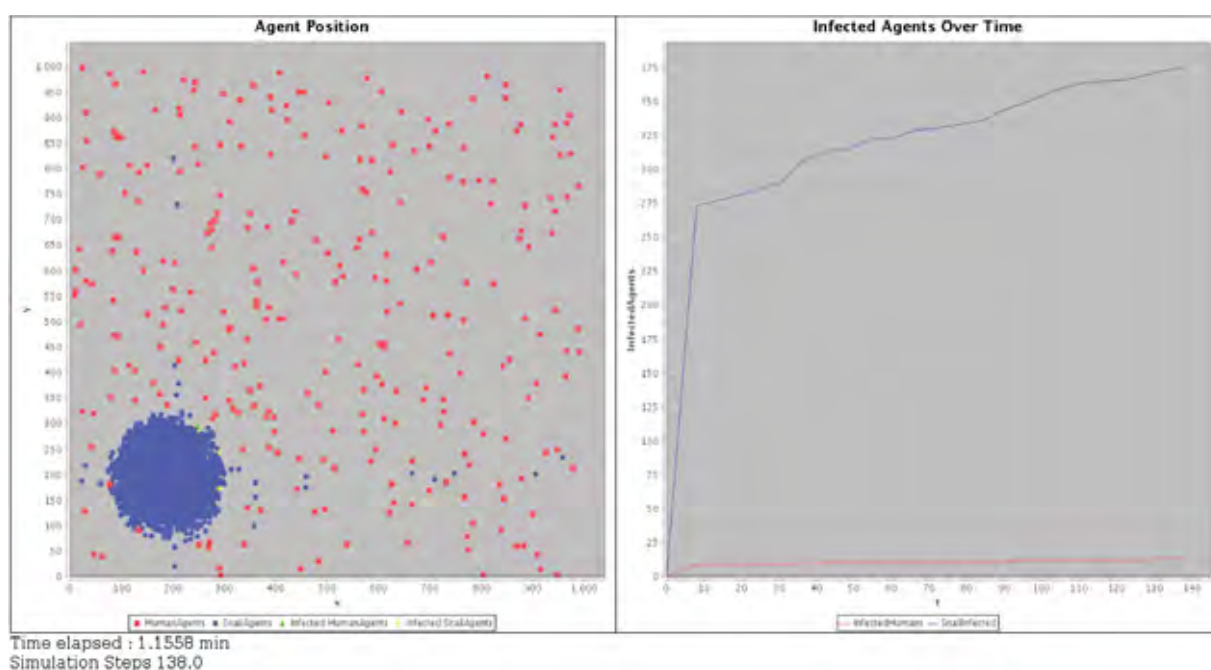


Figura 14 : Dispersão da informação (contágio) nos instantes iniciais da simulação 1.1.

Ao longo do tempo de simulação espera-se que o número de agentes contaminados de ambas populações (humanos e caramujos) tenda a crescer pois a probabilidade de existir um

vetor próximo ao agentes já contaminados agora torna-se ainda mais alta. Contudo, nota-se que o número de caramujos infectados cresce de maneira mais acentuada, pois existe um fator de localidade associado a este fato (Figura 15). Isso indica que ao longo do tempo, mais e mais agentes caramujos são agora portadores dos vermes *S. mansoni*, e portanto a informação se propagará de maneira ainda mais rápida. No caso dos agentes humanos, a taxa de crescimento é menor, pois os humanos não se restringem às regiões de recursos hídricos. Dessa forma, esse fator de localidade se torna menos acentuado.

É de ser observado que nesta simulação o número de agentes humanos e caramujos permanece inalterado, ou seja, assume-se que a população de indivíduos permaneça num valor constante. Numa abordagem futura, é possível definir uma taxa de variação aplicada sobre o número de agentes, agora levando-se em conta a mortalidade e a taxa de nascimento das populações de vetores.

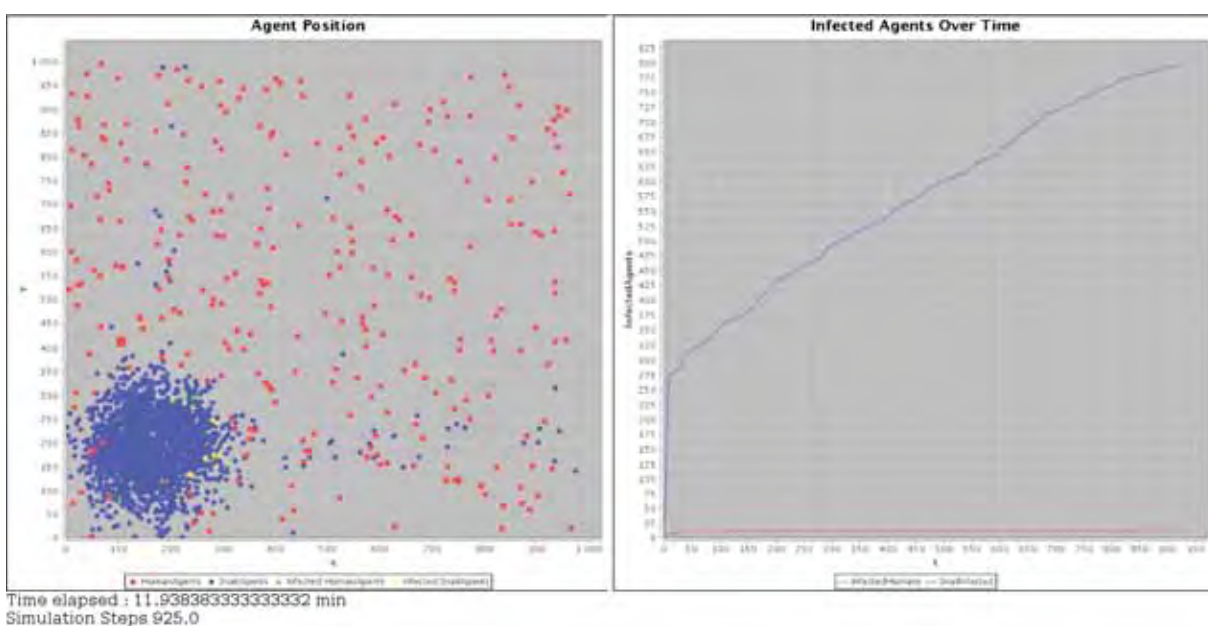


Figura 15 : Simulação 1.1 em tempo intermediário. Observa-se uma alta fração de agentes do tipo caramujo contaminados.

Ao término da simulação observou-se que uma pequena porção de agentes humanos que iniciaram na simulação se tornaram contaminados, em torno de 8%, carregando consigo o verme *S. mansoni*. Isso se deve ao fato de que a informação se difunde ao longo da região geográfica, onde reside o recurso hídrico. Em geral, segundo o problema descrito no capítulo 2 deste trabalho, onde observa-se endemias da doença em córregos, açudes e regiões de baixo

saneamento, o simulador descreveu de maneira adequada o contágio somente nessas áreas de risco (Figura 16).

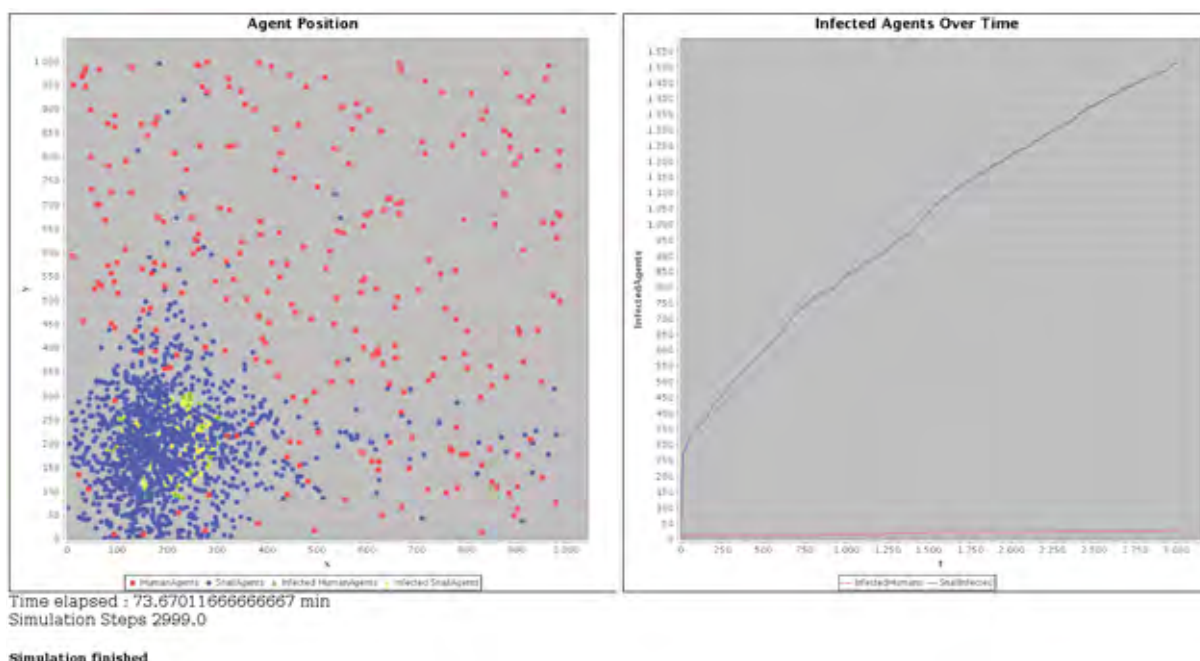


Figura 16 : Término da simulação 1.1. Observa-se uma alta fração de agentes do tipo caramujo contaminados.

Na simulação 1.2, adicionou-se mais um recurso hídrico de maior dimensão e manteve-se o número de agentes da simulação anterior. O parâmetros iniciais dessa simulação descrevem o cenário demonstrado na Figura 17. Os agentes humanos ocupam a região geográfica de maneira aleatória e os agentes caramujos agora se espalham ao longo dos dois recursos hídricos.

Observa-se maior dispersão de agentes caramujos e maior número de agentes humanos próximo aos recursos hídricos. Observa-se agora existe um maior número de agentes humanos nas localidades do recurso hídrico, dessa forma, espera-se ao término da simulação maiores agentes humanos contagiados.

Durante os estágios intermediário e final da simulação (Figuras 18 e 19), observa-se um aumento na taxa com que os agentes caramujos são contaminados. Isso pressupõe-se, tal como esperado, que a dispersão geográfica e o número de humanos próximos aos recursos hídricos são mecanismos determinantes para transmissão da esquistossomose. Em particular, 25% dos humanos se tornaram infectados até o fim da simulação.

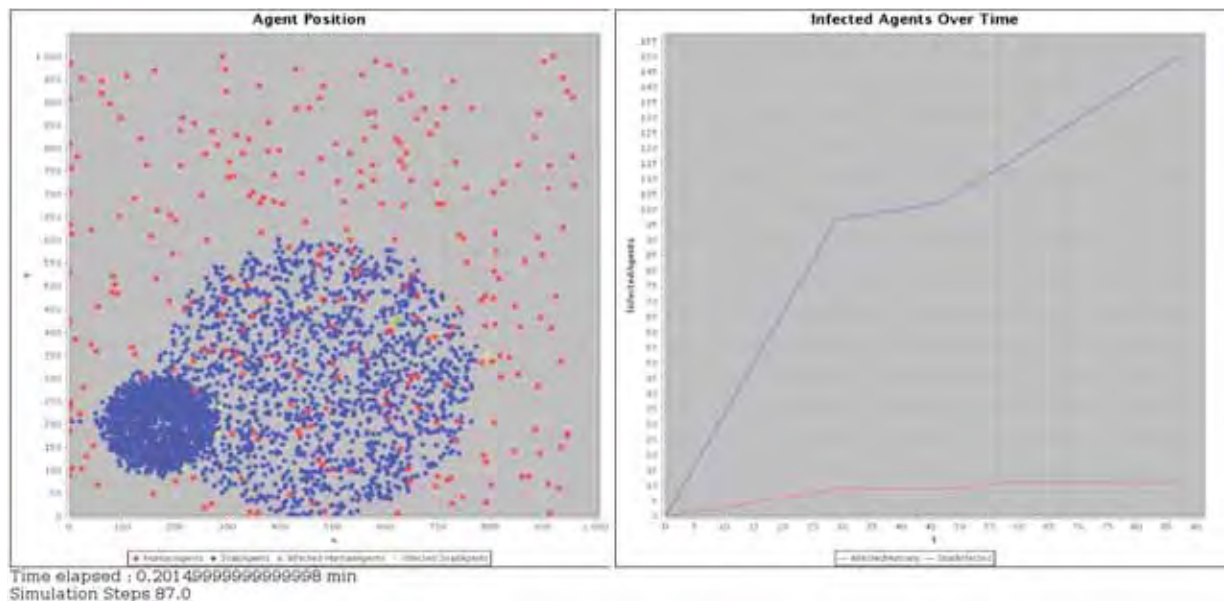


Figura 17 : Estado inicial da simulação 1.2.

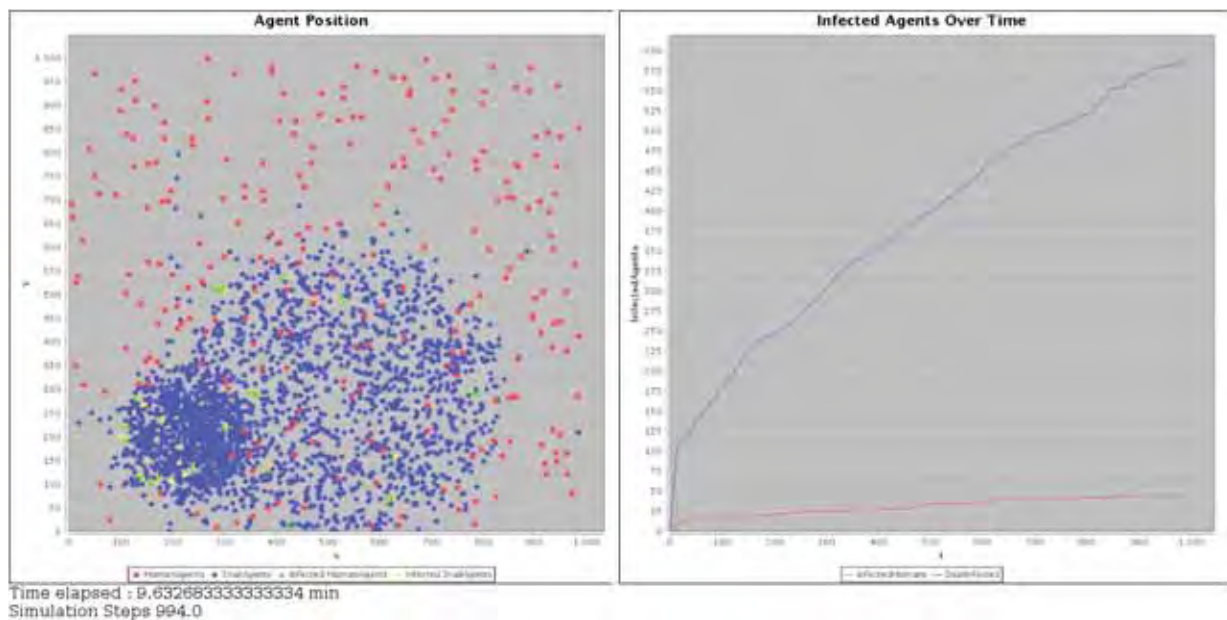
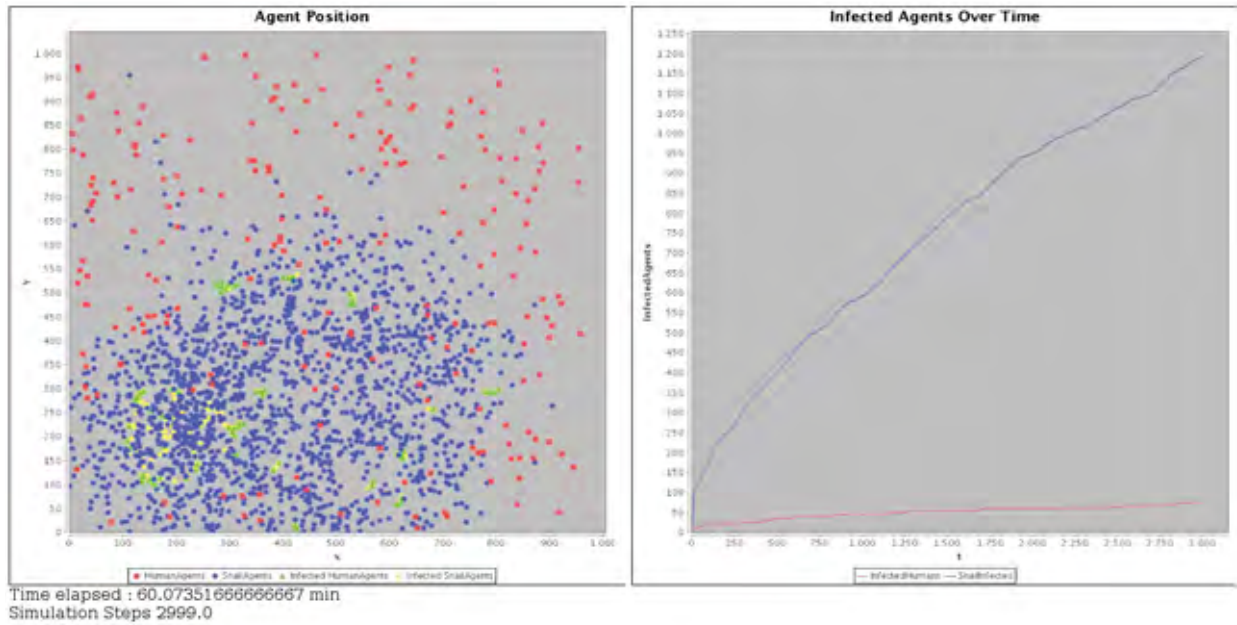


Figura 18 : Simulação 1.2 em tempo intermediário. Observa-se uma alta fração de agentes do tipo caramujo contaminados.



Simulation finished

Figura 19 : Término da simulação 1.2. Observa-se uma alta fração de agentes do tipo caramujo contaminados e um aumento significativo no número de humanos contaminados em comparação com a simulação anterior.

Por fim, a última simulação apresentando três recursos hídricos é analisada da mesma forma das anteriores. Nesta última simulação, os agentes caramujos serão dispostos ocupando uma região ainda maior do espaço geográfico (Figura 8).

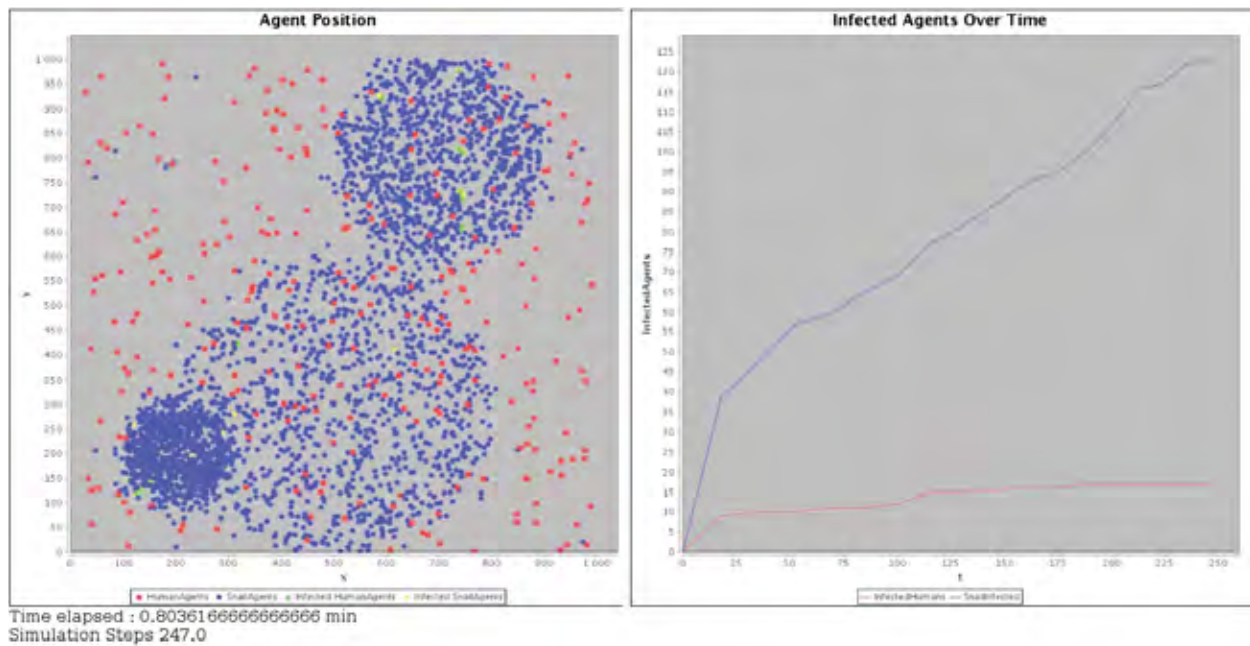


Figura 20 : Início da simulação 1.3. Observa-se um aumento significativo da ocupação do ambiente geográfico por partes dos agentes caramujos.

Dessa forma, espera-se um número ainda maior de agentes humanos sejam contaminados e que a taxa com que os agentes caramujos se contaminam aumente ainda mais.

A hipótese da efeito da dispersão dos agentes pode ser confirmada nas figuras 21 e 22, dada a seguir, demonstrando o resultado final dessa simulação.

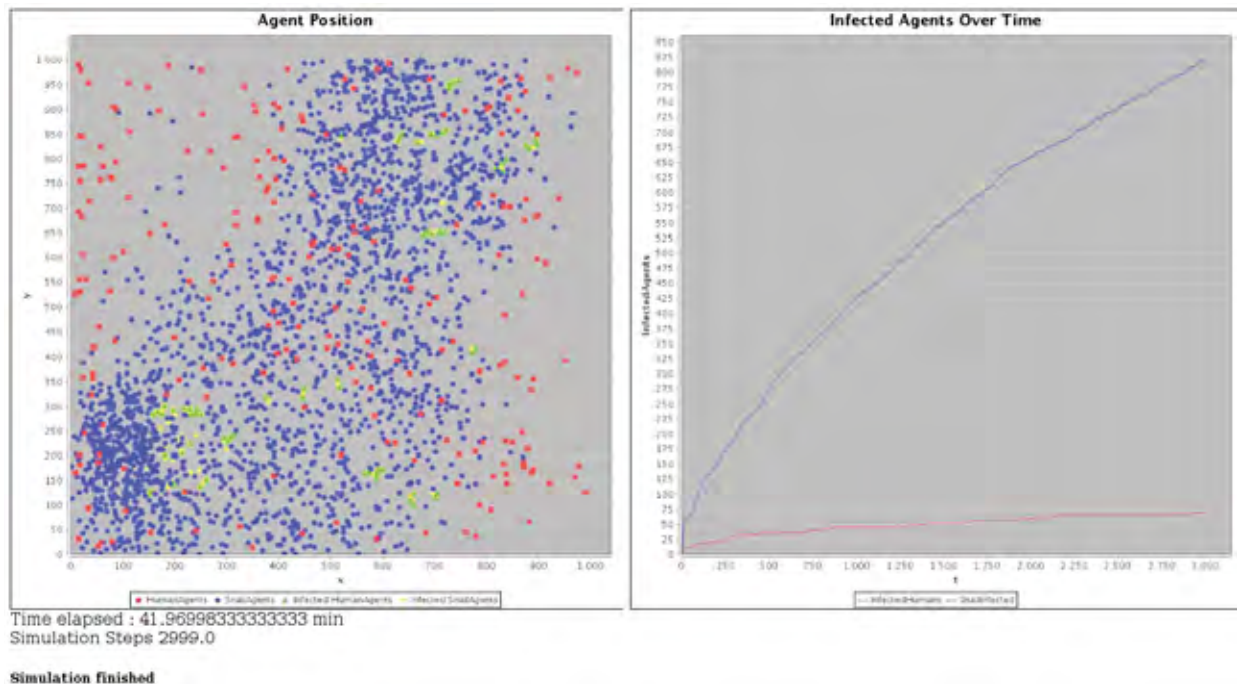


Figura 21 : Término da simulação 1.3.

Os resultados dados nas sessões a seguir, verifica o impacto que a ocupação de agentes humanos faz sobre o problema do contágio.

5.2 ANÁLISE DA INFLUÊNCIA DA POPULAÇÃO DE HUMANOS SOBRE O CONTÁGIO.

O próximo cenário de simulação será realizado a partir da modificação do número de agentes humanos, buscando analisar a influência da ocupação da população local segundo a proximidade dos recursos hídricos. O ambiente considerado permanece com a mesma

quantidade e recursos hídricos da simulação anterior considerada (3 recursos). Os demais parâmetros para as outras populações da simulação permanecerão inalterados.

A simulação inicial, denominada de simulação 2.1, utilizará o número de agentes de 50 agentes humanos. O estado final dessa simulação é dado a seguir na Figura 22.

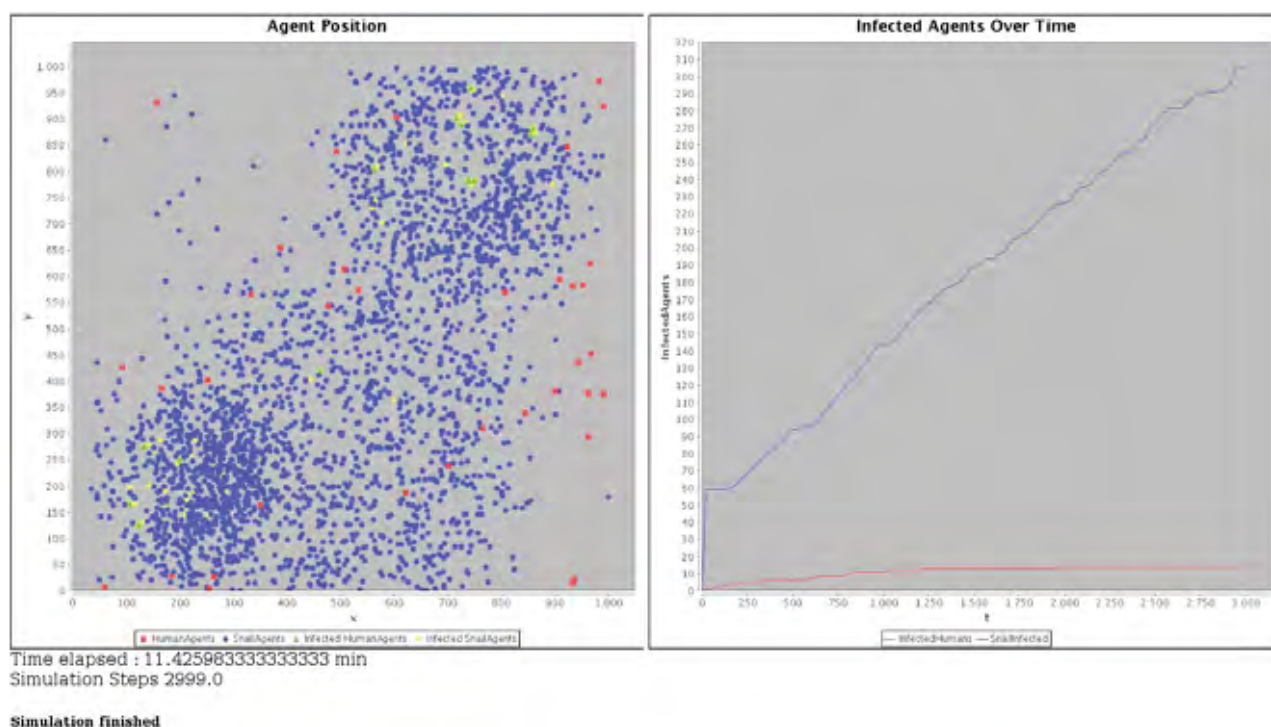


Figura 22 : Estado final da simulação 2.1.

Apesar do ambiente geográfico ser parecido, o número de agentes considerados nesta primeira simulação deste novo cenário (simulação 2.1) é muito menor que a simulação anterior (simulação 1.3), por isso observa-se um comportamento similar ao observado anteriormente (Figura 9), contudo pela baixa densidade populacional de humanos, a taxa de contágio de humanos ainda é reduzida (3% de infectados).

Um comportamento interessante evidente nesta simulação devido ao baixo número de agentes humanos, se deve ao fato de existir regiões no espaço geográfico onde existe uma densidade de agentes caramujos contaminados maiores que os demais. Este comportamento ocorre principalmente próximos às regiões onde existem agentes humanos contaminados, pois os agentes humanos contaminados são condição necessária para a propagação da doença entre a população de caramujos.

Outra evidência deste fato é a existência de agentes caramujos contaminados isolados

em meio ao recurso hídrico. Estes agentes somente se mantêm durante toda a simulação por não existir nenhum mecanismo que os removam da simulação, uma vez que assume-se que nenhum agente caramujo morre durante a execução.

As próximas simulações, simulação 2.2, 2.3 e 2.4 serão executadas com um número de 1000, 3000 e 6000 agentes humanos respectivamente.

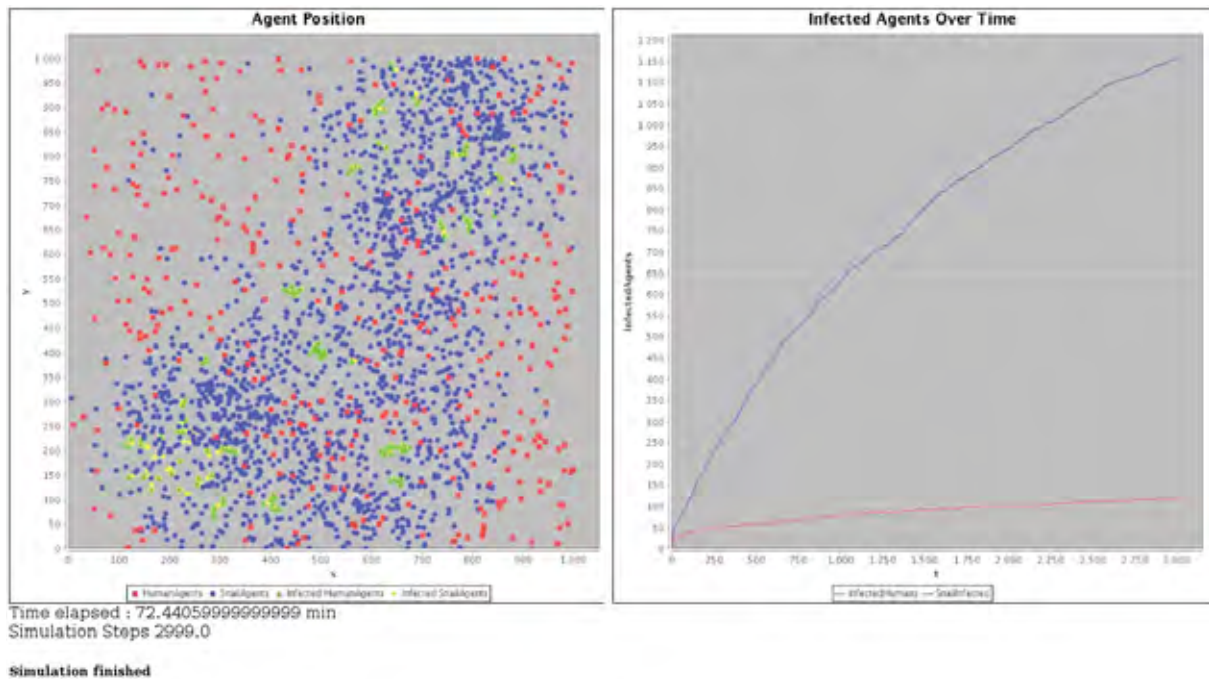


Figura 23 : Estado final da simulação 2.2.

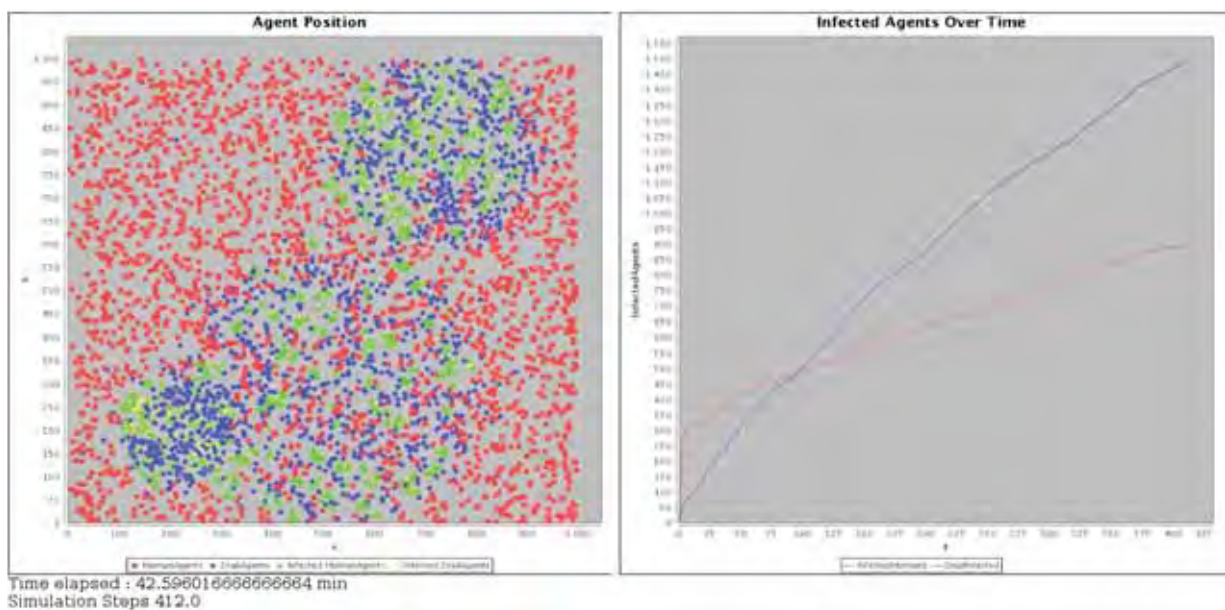


Figura 24 : Estado inicial da simulação 2.3, após 400 passos.

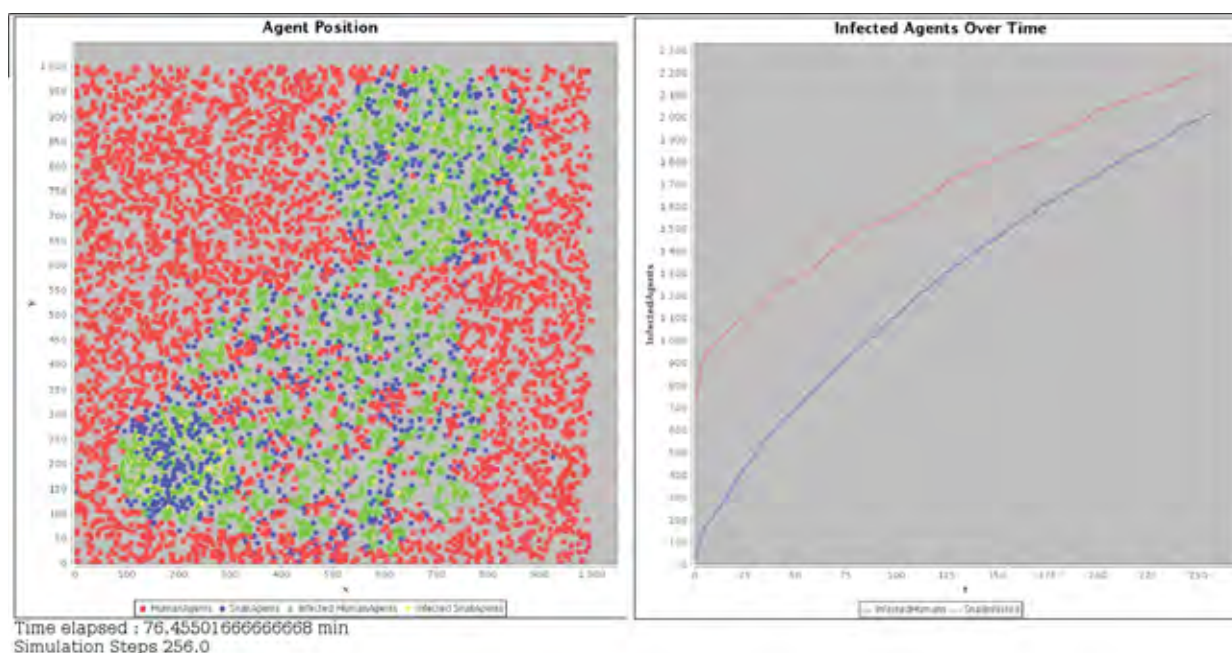


Figura 25 : Estado inicial da simulação 2.4, após 200 passos.

Os resultados das quatro simulações consideradas para este cenário (Figuras 23, 24 e 25), sugerem que a densidade populacional de agentes humanos interfere na taxa com que o contágio ocorre, pois observa-se o elevado número de agentes contaminados num curto intervalo de tempo para as simulações 2.3 e 2.4, já nos estágios iniciais da simulação (entre 200 e 400 passos). Este comportamento era esperado pois o fator localidade é um fator crítico para a propagação da doença, ou da mesma forma, segundo o modelo de agentes brownianos, quanto maior o número de agentes numa dada região do espaço, mais rápido ocorre a difusão da informação para outras regiões.

O trabalho segue com a análise do impacto que as populações de caramujos apresentam sobre o problema do contágio.

5.3. ANÁLISE DA INFLUÊNCIA DA POPULAÇÃO DE CARAMUJOS SOBRE O CONTÁGIO.

Por fim, para o último cenário de simulação foi realizado mais uma série de

simulações variando o número de agentes caramujos (da classe *Biomphalaria*). O número de recursos hídricos e os demais parâmetros se mantiveram inalterados. Para todas as simulações apresentadas nesta sessão, o número de agentes humanos é de 300 indivíduos.

A primeira simulação foi executada com um número de 1000 agentes caramujos, o resultado obtido ao término da simulação é demonstrado na Figura 26 a seguir.

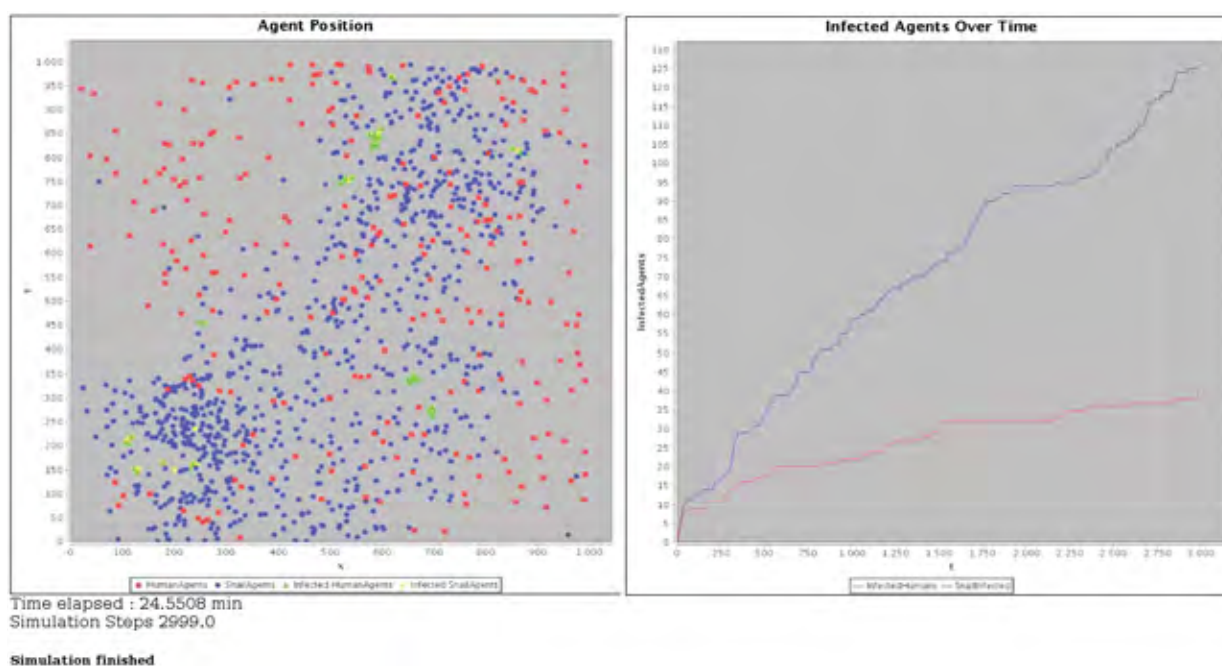


Figura 26 : Estado final da simulação 3.1.

O resultado obtido pela simulação 3.1, demonstra que a baixa densidade populacional de caramujos nos recursos hídricos, foi o fator que não permitiu a dispersão da doença. Isso pode ser verificado a partir das regiões de contágio bem características e isoladas das regiões onde existem indivíduos sadios. Essa análise vale tanto para a população de vetores caramujos como também para a população de humanos. Apesar disso, existiram ainda uma parcela significativa de humanos contaminados (13%)

As próximas simulações (simulação 3.2, 3.3 e 3.4) apresentarão 2000, 4000 e 8000 agentes caramujos respectivamente. O resultado final de cada uma dessas simulações serão apresentados nas figuras 27, 28 e 29.

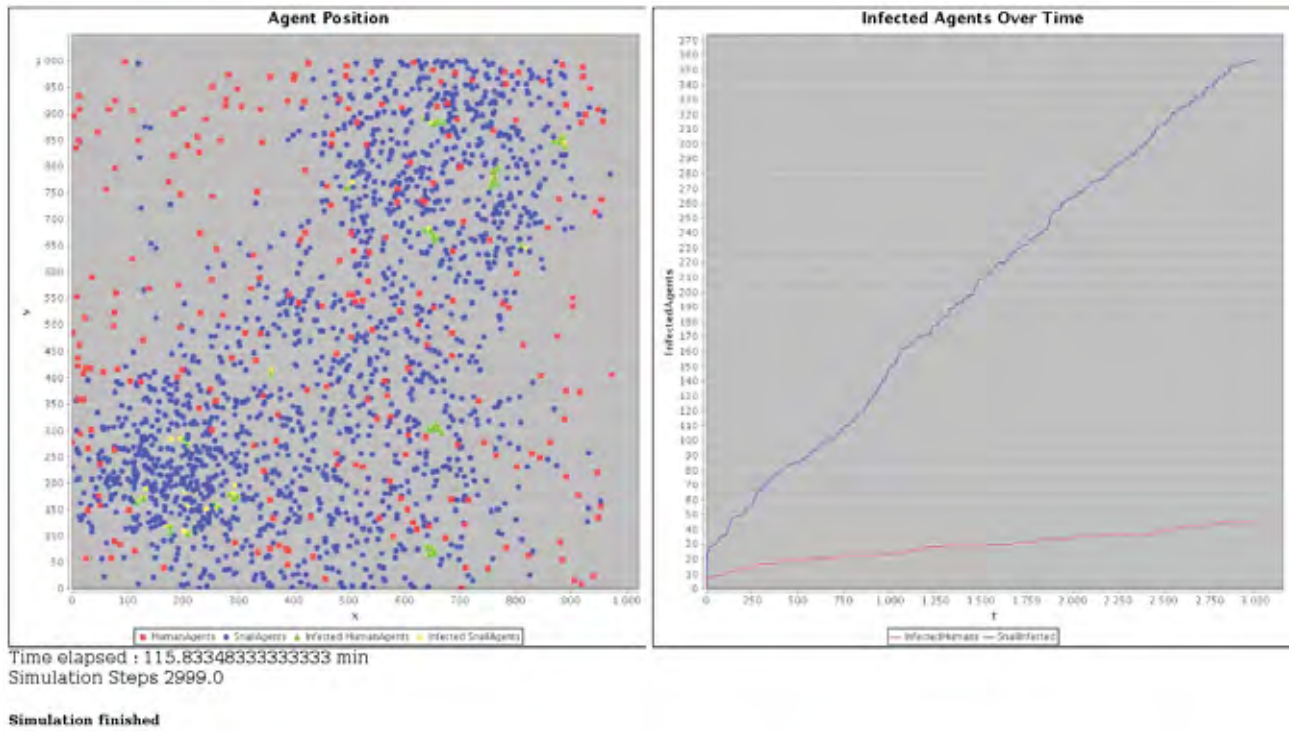


Figura 27 : Estado final da simulação 3.2.

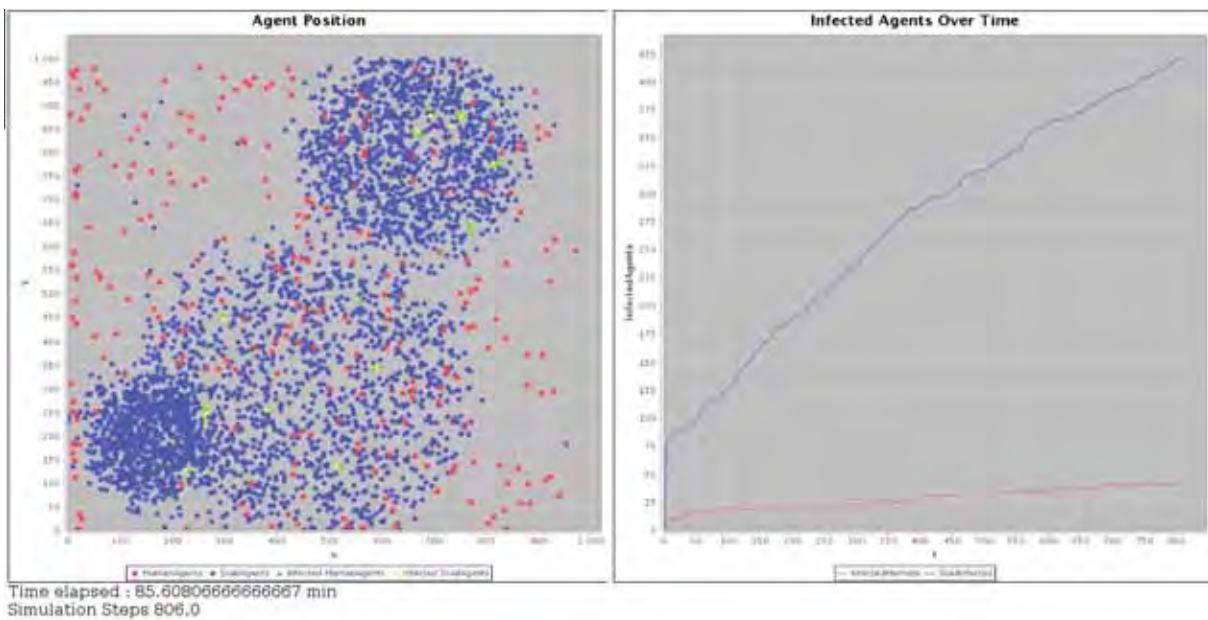


Figura 28 : Estado intermediário da simulação 3.3.

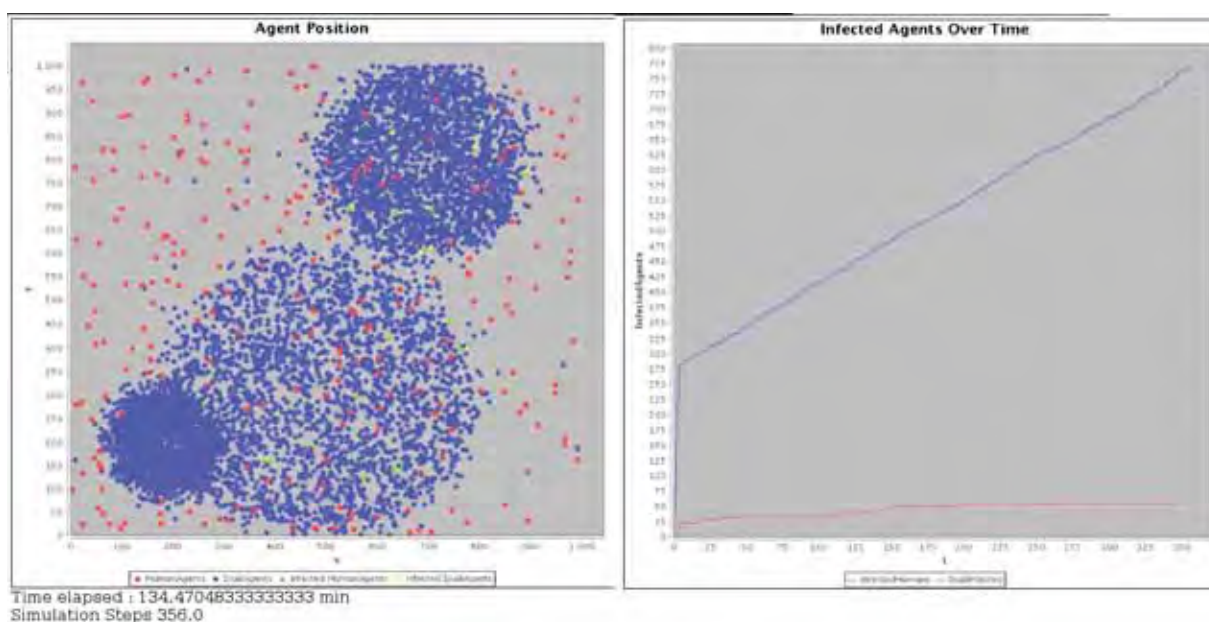


Figura 29 : Estado intermediário da simulação 3.4.

Por fim, observa-se que o aumento na população de caramujos, facilita a disseminação de indivíduos contaminados na população humana. Isso se deve ao aumento significativo de indivíduos da população de cercárias, aumentando a possibilidade de contágio a partir dos indivíduos humanos. Este comportamento era esperado, pois o elevado número de agentes facilita a propagação da informação, como discutido já anteriormente.

Em geral, os resultados obtidos a partir do simulador nesta sessão, demonstraram que o fator localidade é determinante para o contágio, e este fator é influenciado pela densidade populacional de vetores encontrados ao longo do espaço geográfico.

6. CONCLUSÃO

Este trabalho teve por objetivo o desenvolvimento de um simulador que representasse de maneira eficaz o problema do contágio da doença da esquistossomose, considerando-se a interação entre as populações de vetores e parasitas.

Um modelo matemático e computacional foi desenvolvido de modo a tornar possível a simulação deste problema. Além disso, a arquitetura do simulador foi implementada de modo a reduzir, dentro do possível, o acoplamento entre as classes. Isso pôde ser possível a partir da adoção do modelo MVC. A arquitetura do simulador foi implementada a partir de uma estrutura WEB, considerando-se o uso da tecnologia JSP e de Servlets.

Outro aspecto considerado na concepção do simulador se deve à forma com que o usuário configura os parâmetros chave para o simulador.

Com relação à precisão do modelo, o simulador foi submetido a uma série de testes verificando diversos parâmetros, e de forma geral, o simulador foi validado demonstrando que o mesmo representa adequadamente a realidade do contágio. Contudo, ainda nenhum dado real foi submetido à teste no simulador.

Um fato a ser considerado se deve ao elevado tempo de execução para cada cenário de simulação que apresentam um elevado número de agentes. Observou-se que mesmo os cenários onde um menor número de agentes foi utilizado, apresentaram tempos de execução da ordem de 30 a 60 min.

SUGESTÕES PARA FUTUROS TRABALHOS.

- Desenvolver um estudo para aplicar técnicas de paralelização ao sistema.

- Modelar outros recursos do ambiente para agentes humanos e caramujos.
- Modificar o modelo de modo que o tempo de vida das espécies de caramujos varie ao longo do tempo.
- Aplicar outras técnicas de cálculo numérico às equações diferenciais.
- Modificar o modelo levando em consideração medidas profiláticas às populações de humanos.
- Modificar o modelo de simulação da levando em consideração aspectos sociais e educacionais.
- Desenvolver um experimento para validar o simulador utilizando dados reais.
- Aplicar o modelo de multiagentes a outros sistemas complexos, ou dinâmicas populacionais.
- Estabelecer mecanismos para integrar dados geográficos ao sistema multiagente.

REFERÊNCIAS

1. ALI W., Developing 2d and 3d multiagent geosimulation, a method and its application: the case of shopping behavior geosimulation in square one mall (Toronto) , 2006.
2. FURTADO V., Modelagem e Simulação Multiagente da Criminalidade, *Comp.Sci.* 0002. Dec. 7, 2008.
3. BOUSQUET F., LE PAGE C. Multi-agent simulations and ecosystem management: a review. *Ecol. Model.* 176 (2004), 313–332.
4. BOUSQUET F., BARRETEAU O., LE PAGE C., MULLON C., WEBER J. An environmental modelling approach: the use of multi-agents simulations. *Advances in Environmental and Ecological Modelling*. Elsevier, 1999, pp. 113–122.
5. ALMEIDA, S.J.D., FERREIRA, R.P.M., EIRAS, Á.E., OBERMAYR, R.P., GEIER, M. Multi-agent modeling and simulation of an *Aedes aegypti* mosquito population. *Environmental Modelling and Software*(2010) 1490-1507.
6. VIGILÂNCIA e controle e moluscos de importância epidemiológica: diretrizes técnicas: Programa de vigilância e controle da Esquistossomose (PCE). 2 ed. Brasília, D.F.: ed. MS, 2008. 177p.
7. BERENGUER J. G. , Atlas de Parasitologia, Livro Ibero-Americano LTDA, Rio de Janeiro.
8. SCHALL V., MASSARA C. L., ENK M.J., BARROS H.S. Os Caminhos da Esquistossomose. Parte I Dentro do nosso corpo. Centro de Pesquisas René Rachou/Fiocruz. 2007. (Série Esquistossomose, 8).
9. SCHALL V., MASSARA C. L., ENK M.J., BARROS H.S. Os Caminhos da Esquistossomose. Parte II No meio ambiente. Centro de Pesquisas René Rachou/Fiocruz. 2007. (Série Esquistossomose, 8).
10. CHWIF L., MEDINA A. C., Modelagem e Simulação de eventos discretos: teoria e aplicações 3 ed., Bravarte, 2010.
11. MACAL C. M., NORTH M. J., Tutorial on agent-based modeling and simulation, *Proceedings of the 2005 Winter Simulation Conference*, 2005; 2 – 15.
12. MACAL C. M., NORTH M. J., Tutorial on agent-based modeling and simulation part 2: how to model with agents, *Proceedings of the 2006 Winter Simulation Conference*, 2006; 73 – 83.
13. RATZE C., GILLET F., MULLER J. P., STOFFEL K., Simulation modelling of ecological hierarchies in constructive dynamical systems , *Ecological Complexity* 4 (2007) 13–25.
14. SCHWEITZER, F.; LAO, K.; FAMILY, F. (1997). Active Random Walkers Simulate Trunk Trail

Formation by Ants, *BioSystems* 41, 153-166.

15. ISIDORO C., FACHADA N., BARATA F., ROSA A., Agent-Based Model of *Aedes aegypti* Population Dynamics , L. Seabra Lopes et al. (Eds.): EPIA 2009, LNAI 5816, pp. 53–64, 2009.

16. LAPERRIE V., BADARIOTTI D., BANOS A., MULLERRE J. P., Structural validation of an individual-based model for plague epidemics simulation, *Ecological Complexity* 6 (2009) 102–112.

17. LINARD C., PONCON N., FONTENILLE D., LAMBIN E. F., A multi-agent simulation to assess the risk of malaria re-emergence in southern France , *Ecological Modelling* 220 (2009) 160–174.

18. RUSSELL S., NORVIG P. Artificial Intelligence – a modern approach 2ed, Elsevier, 2003.

19. COPPIN B., Inteligência Artificial, Ed. 1, LTC, Rio de Janeiro, 2010.

20. FERNANDES, A. M. Inteligência artificial : noções gerais. Florianópolis: VisualBooks, 2005. 160p.

21. BORSHCHEV, A., FILIPPOV, A., From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools. In: Proceedings of the 22nd International Conference of the System Dynamics Society, July 25–29 (2004).

22. BOUSQUET, F., LE PAGE, C., 2004. Multi-agent simulations and ecosystem management: a review. *Ecol. Model.* 176, 313–332.

23. LINARD C., Spatial and integrated modelling of the transmission of vector-borne and zoonotic infections, 2009.

24. DROGOUL A., & FERBER J. Multi-agent simulation as a tool for modeling societies: Application to social differentiation in ant colonies. In Proceedings of 4th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Artificial Social Systems (MAAMAW), Lecture Notes in Computer Science, vol. 830 (1992) , pp. 3-23.

25. CEDERMAN, L.-E., Endogenizing geopolitical boundaries with agent-based modeling, *Proc. National Academy of Sciences*, 2002, 99 (suppl. 3):7796-7303.

26. BONABEAU, E., Agent-based modelling: methods and techniques for simulating human systems. *Proc. National Academy of Science*, 2001, 99, 7280 – 7287.

27. MARTINS F. T. , Mapeamento do risco da esquistossomose no estado de minas gerais, usando dados ambientais e sociais, 2009.

28. PAZ R. J. da, Biologia e Ecologia de *Biomphalaria glabrata* (Say, 1818) (Mollusca: Pulmonata: Planorbidae), na Fazenda Árvore Alta, Alhandra (Paraiíba: Brasil) , 1997.

29. GIOVANELLI A., SOARES M. S., D'ANDRÉA P. S., GONÇALVES M. M. L., REY L., Abundância e infecção do molusco *Biomphalaria glabrata* pelo *Schistosoma mansoni* no Estado do Rio de Janeiro, Brasil, *Rev Saúde Pública* 2001; 35(6): 523-30.

30. Souza et al..*Revista do Instituto de Medicina Tropical de São Paulo*, 1994 36(6): 485-489.

31. Muñoz-Antoli et al.. *Parasitology Research*, 2007, 101(6): 1663-1670.

32. BOAVENTURA M. F., FERNANDEZ M. A., THIENGO S. C., SILVA R. E., MELO. A. L., Formas larvais de Trematoda provenientes de gastrópodes límnicos da microrregião Rio de Janeiro, sudeste do Brasil, *Lundiana* 3 : 45 – 49.

33. VINAUD M. C., SANTOS S. C., FERRI P. H., LINO JUNIOR R. de S., BEZERRA J. C. B., Avaliação da atividade larvívora de plantas fitoterápicas do cerrado do gênero *Stryphnodendron spp.* sobre miracídeos e cercárias de *Schistosoma mansoni*, Revista De Patologia Tropical ,Vol. 34 (2): 137-143. maio-ago. 2005 .
34. SHIN J.,PARK Y.. Brownian agent-based technology forecasting. Technological Forecasting & Social Change 76 (2009) 1078–1091.
35. SCHWEITZER F. Brownian Agents and Active Particles : Collective Dynamics in the Natural and Social Sciences, Ed. Springer-Verlag, 2003 Berlin.
36. EBELING, W.; SCHWEITZER, F.; TILCH, B. Active Brownian particles with energy depots modelling animal mobility. BioSystems 49 (1999), 17–29.
37. DEITEL P., DEITEL H. Java como programar. ed. 8, São Paulo: Pearson, 2010, p 1145.
38. BURDEN R. L., FAIRES J. D., Análise Numérica, Editora Thomson, (2003) 605-614.
39. TROVÓ D. G. E., BALTHAZAR J . M., I Minicurso: Métodos Seqüencial e Paralelo das Diferenças Finitas Aplicados à Ciência da Engenharia. 4o Congresso Temático de Dinâmica, Controle e Aplicações (2005).
40. LARMAN C. Utilizando UML e padrões: uma introdução à análise e ao projeto orientado a objetos. ed. 3, Porto Alegre: Bookman, 2007. 694 p.
41. SOMMERVILLE I. Engenharia de software. 8.ed. São Paulo: Pearson Addison-Wesley, 2007. xiv, 552p.
42. HORSTMANN C. S.; CORNELL G.. Core Java 2 : fundamentos. 7.ed. Rio de Janeiro: Alta Books, 2005. viii, 568p.
43. MOREIRA N.,OZIEL . Entendendo e dominando o Java para internet. 2.ed. São Paulo: Digerati Books, 2009. 318p.
44. The JFreeChart Class Library Version 1.0.4 Developer Guide Written by David Gilbert February 9, 2007 c 2000-2007, Object Refinery Limited. All rights reserved.
45. Java EE and.NET interoperability : integration strategies, patterns and best practices. Upper Saddle River,NJ: Sun Microsystems Press : Prentice Hall, 2006. xxxvii, 608p.
46. The Apache Software Foundation, disponível em <http://www.apache.org/>. Acessado em março, 2012.