

Sandro Jerônimo de Almeida

Modelo e simulação multiagente de uma população de mosquitos
Aedes aegypti

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica de Minas Gerais, como requisito parcial para a obtenção do grau de Mestre em Informática.

Belo Horizonte

Setembro de 2007

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.



PUC Minas
Programa de Pós-graduação em Informática

FOLHA DE APROVAÇÃO

Modelo e simulação multiagente de uma população de mosquitos *Aedes aegypti*

Sandro Jerônimo de Almeida

Dissertação defendida e aprovada pela seguinte banca examinadora:

Prof. Ricardo Poley Martins Ferreira - Orientador (PUC Minas)
Doutor em Ciência da Computação (UFMG)

Prof. Henrique Pacca Loureiro Luna (UFAL)
Doutor em Automática e Otimização (Universidade de Toulouse III, França)

Prof. Álvaro Eduardo Eiras (UFMG)
Doutor em Entomologia (Universidade de Southampton, Inglaterra)

Prof. Clodoveu Augusto Davis Junior (PUC Minas)
Doutor em Ciência da Computação (UFMG)

Belo Horizonte, 12 de setembro de 2007.

FICHA CATALOGRÁFICA

Elaborada pela Biblioteca da Pontifícia Universidade Católica de Minas Gerais

A447m	<p>Almeida, Sandro Jerônimo de Modelo e simulação multiagente de uma população de mosquitos <i>Aedes aegypti</i> / Sandro Jerônimo de Almeida. – Belo Horizonte, 2007. 153 f. : il.</p> <p>Orientador: Prof. Dr. Ricardo Poley Martins Ferreira. Dissertação (mestrado) – Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-graduação em Informática. Bibliografia.</p> <p>1. Simulação (Computadores). 2. <i>Aedes aegypti</i> – Simulação por computador. 3. Framework (Programa de computador) I. Ferreira, Ricardo Poley Martins. II. Pontifícia Universidade Católica de Minas Gerais. III. Título</p> <p>CDU: 681.3.091:614.449</p>
-------	---

Bibliotecário: Fernando A. Dias – CRB6/1084

Resumo

Este trabalho apresenta um simulador de população de mosquitos *Aedes aegypti*. O simulador desenvolvido realiza simulações de micro-regiões como, por exemplo, uma casa com seus moradores, mosquitos voando no ambiente e objetos da casa. O simulador é composto por um modelo de simulação denominado *SimPopMosq* e um *framework* de simulação de sistemas ecológicos denominado *SIMES*. O modelo de simulação *SimPopMosq* adota abordagem baseada em agentes computacionais. Este modelo possui agentes que representam mosquitos *Aedes aegypti* e alguns mamíferos. Uma das principais contribuições deste trabalho foi a definição do modelo de tomada de decisão dos mosquitos *Aedes aegypti*. O modelo *SimPopMosq* é descrito conceitualmente através do protocolo *ODD* (*Overview, Design concepts and Details*). Este protocolo foi projetado para descrever modelos baseados em indivíduos. O *framework SIMES* é base para a implementação do modelo *SimPopMosq*, pois oferece um conjunto de funcionalidades para a simulação dos agentes. Este *framework* é uma extensão do *framework* de simulação multiagentes *Repast*. Vários experimentos computacionais foram realizados com o objetivo de validar e demonstrar a coerência e a eficiência do modelo proposto. Os resultados das simulações foram comparados com dados obtidos em experimentos biológicos reais. Os resultados obtidos são promissores e a correlação entre os dados reais e os dados simulados foram superiores a 90% nas principais variáveis analisadas.

Abstract

This work presents a mosquitoes *Aedes aegypti* population simulator. The simulator developed allows performing micro-regions simulations, for example, a house with its residents, mosquitoes flying and house's objects. The simulator is composed by one model of simulation called *SimPopMosq* and a *framework* for simulation of ecological systems called *SIMES*. The model of simulation *SimPopMosq* adopts approach based on computational agents. This model has agents who represent mosquitoes *Aedes aegypti* and some mammals. One main contribution of this study was the definition of model to the decision-making of the mosquitoes *Aedes aegypti*. The model *SimPopMosq* was conceptually described by the *ODD* protocol (Overview, Design concepts and Details). This protocol was designed to describe individual based models. The *framework SIMES* is the base for the implementation of the model *SimPopMosq*, because offers a set of functionality for agent based simulations. This *framework* is an extension of the *Repast framework*. Several computational experiments were conducted with the objective to validate and show the coherence and the efficiency of the proposed model. Results of simulations were compared with data collected from real biological experiments. The results obtained were promising and the correlation between real biological data and simulated data was above 90% in the main analyzed variables.

*Aos meus pais
Pelo amor
e incentivo.*

Agradecimentos

```
public class Agradecimentos {  
  
    public Agradecimentos(String nomeAgente) {  
  
        String mensagem = "";  
  
        if (nomeAgente.equals("DEUS") || nomeAgente.equals("JESUS")) {  
            mensagem = "Agradeço a DEUS pela vida e pela oportunidade " +  
                "de concretizar o sonho de realizar um mestrado.";  
        }  
        else if (nomeAgente.equals("Ricardo Poley")) {  
            mensagem = "Agradeço ao meu orientador pela paciência, " +  
                "por toda ajuda no desenvolvimento dos trabalhos " +  
                "realizados e em especial pela maravilhosa " +  
                "oportunidade de convivência.";  
        }  
        else if (nomeAgente.equals("Pai") || nomeAgente.equals("Mãe")) {  
            mensagem = "Agradeço por todo incentivo que meus PAIS me deram " +  
                "para enfrentar esta jornada, pelo apoio nas horas " +  
                "difíceis e pela infra-estrutura familiar que me " +  
                "permitiu desenvolver meus trabalhos em PAZ!";  
        }  
        else if (nomeAgente.equals("Célia")) {  
            mensagem = "Agradeço profundamente a Célia por me tolerar " +  
                "nos momentos de insegurança, pelo apoio dados nas " +  
                "horas difíceis e pelas sábias ponderações nos " +  
                "momentos de decisão.";  
        }  
        else if (nomeAgente.equals("Irmãos")) {  
            mensagem = "Agradeço aos meus irmãos e irmãs pela convivência " +  
                "tranquila e carinho.";  
        }  
        else if (nomeAgente.equals("Professores do Mestrado")) {  
            mensagem = "Agradeço aos mestres pela contribuição para minha " +  
                "formação pessoal e profissional.";  
        }  
        else if (nomeAgente.equals("Colegas de Mestrado")) {  
            mensagem = "Agradeço aos colegas com que pude compartilhar " +  
                "vários momentos no mestrado. Esta convivência " +  
                "contribuiu para ampliar nossos laços de amizade.";  
        }  
        else if (nomeAgente.equals("Funcionários do Mestrado")) {  
            mensagem = "Agradeço a todos os funcionários do mestrado pela " +  
                "amizade e suporte. Agradeço especialmente a Giovana " +  
                "por toda sua paciência.";  
        }  
        else if (nomeAgente.equals("CAPES") || nomeAgente.equals("PUC Minas")) {  
            mensagem = "Agradeço a CAPES e PUC Minas pelo apoio financeiro " +  
                "e oportunidade de estudo.";  
        }  
        System.out.println(mensagem);  
    }  
}
```


Conteúdo

Lista de Figuras	viii
Lista de Tabelas	xi
1 Introdução	1
2 Referencial Teórico	6
2.1 Modelos baseados em indivíduos	6
2.2 Sistemas multiagentes	8
2.2.1 Agentes	10
2.2.2 Engenharia de software orientada a agentes	12
2.2.3 <i>Frameworks</i> de simulação multiagentes	12
2.3 Trabalhos relacionados	17
2.3.1 Modelos matemáticos	17
2.3.2 Modelos baseados em indivíduos	20
2.4 O mosquito <i>Aedes aegypti</i> e a <i>dengue</i>	22
2.4.1 A dengue	22
2.4.2 A biologia do mosquito <i>Aedes aegypti</i>	23
3 Modelo Conceitual	25
3.1 Visão geral	26
3.1.1 Propósito	26
3.1.2 Variáveis de estado e escalas	26
3.1.3 Visão de processo e escalonamento	36
3.2 Conceitos de projeto	40
3.2.1 Fenômenos emergentes	40

3.2.2	Adaptabilidade	40
3.2.3	<i>Fitness</i>	41
3.2.4	Predição	42
3.2.5	Sensibilidade	42
3.2.6	Interação	42
3.2.7	Estocasticidade	43
3.2.8	Coletividade	44
3.2.9	Observação	44
3.3	Detalhes	45
3.3.1	Inicialização	45
3.3.2	Entrada	46
3.3.3	Sub-modelos	48
4	Modelo de Implementação	56
4.1	Tecnologia base	57
4.1.1	Desenvolvimento de sistemas	57
4.1.2	Recursos computacionais	58
4.2	<i>Framework SIMES</i>	58
4.2.1	Itens estendidos do <i>framework Repast</i>	59
4.2.2	Classes	61
4.3	Modelo de simulação <i>SimPopMosq</i>	70
4.3.1	Classes	70
4.4	Simulação	75
4.4.1	Classes	76
5	Experimentos Computacionais	77
5.1	Experimentos qualitativos - Ilustração do modelo	77
5.1.1	Questionário de decisão do mosquito <i>Aedes aegypti</i> macho	77
5.1.2	Questionário de decisão do mosquito <i>Aedes aegypti</i> fêmea	80
5.2	Experimento <i>Green-House</i> - Validação do modelo	84
5.2.1	Dados experimentais <i>reais</i>	84
5.2.2	Simulações computacionais	87
5.2.3	Calibração de parâmetros	88

5.2.4	Validação do modelo	89
5.3	Estudo de caso - Belo Horizonte	100
5.3.1	Cenário real	101
5.3.2	Simulações computacionais	101
5.3.3	Resultados	104
6	Comentários e Conclusões	107
6.1	Limitações do trabalho	109
6.2	Trabalhos futuros	110
	Bibliografia	111
	Apêndice	119
A	Dados Meteorológicos	120
B	Distribuição de Ovos do Mosquito <i>Aedes aegypti</i>	123
C	<i>Green-House</i> - Resultados Complementares	126
C.1	Evolução da produção de ovos	126
C.2	Evolução da produção de adultos	128
C.3	Comportamento da taxa de picadas	129
D	Parâmetros	131
D.1	Parâmetros da simulação	131
D.2	Parâmetros de infra-estrutura	139

Lista de Figuras

1.1	Composição do simulador	4
2.1	Estrutura de um sistema <i>RDP</i> [Sichman, 2003]	9
2.2	Agentes interagindo com o ambiente [Russell and Norvig, 2003]	10
2.3	Visão geral do <i>framework Repast</i> [North and Macal, 2005]	15
2.4	Modelo de simulação <i>HeatBugs</i>	17
3.1	Estrutura do protocolo <i>ODD</i> [Grimm et al., 2006]	25
3.2	Elementos que compõem o modelo de simulação <i>SimPopMosq</i>	27
3.3	Exemplo de espaço discreto formado pelos eixos <i>x</i> e <i>y</i>	28
3.4	Movimento de um ser humano e seu rastro de odor	29
3.5	Níveis de proximidade de vegetação para o mosquito <i>Aedes aegypti</i>	32
3.6	Protocolo <i>ODD</i> - Visão de processo e escalonamento	37
3.7	Ciclo de vida do agente mosquito <i>Aedes aegypti</i>	38
3.8	Protocolo <i>ODD</i> - Conceitos de projeto	40
3.9	Protocolo <i>ODD</i> - Inicialização	45
3.10	Inicialização da simulação	45
3.11	Protocolo <i>ODD</i> - Sub-modelos	48
3.12	Rastro de difusão: Autômato celular	48
3.13	Questionário do modelo de comportamento do mosquito <i>Aedes aegypti</i> macho	50
3.14	Questionário do modelo de comportamento do mosquito <i>Aedes aegypti</i> fêmea	53
4.1	Composição tecnológica do simulador	56
4.2	Diagrama de pacotes do <i>framework SIMES</i>	62
4.3	Diagrama de pacotes detalhado do <i>framework SIMES</i>	63
4.4	Diagrama de classes do <i>framework SIMES</i>	67

4.5	Diagrama de classes: Relacionamento entre as classes dos <i>frameworks Re-</i> <i>past</i> e <i>SIMES</i>	69
4.6	Diagrama de pacotes do modelo <i>SimPopMosq</i>	71
4.7	Diagrama de pacotes detalhado do modelo <i>SimPopMosq</i>	72
4.8	Diagrama de classes do modelo <i>SimPopMosq</i>	73
4.9	Diagrama de classes: Relacionamento entre as classes dos <i>SIMES</i> e o mo- delo de simulação <i>SimPopMosq</i>	74
4.10	Visão geral do simulador de população de mosquitos	75
4.11	Diagrama de classes: Instância de simulação	76
5.1	Simulação do mosquito <i>Aedes aegypti</i> macho: execução detalhada	78
5.2	Simulação do mosquito <i>Aedes aegypti</i> macho: primeiro momento	79
5.3	Simulação do mosquito <i>Aedes aegypti</i> macho: segundo momento	80
5.4	Simulação do mosquito <i>Aedes aegypti</i> fêmea: execução detalhada	82
5.5	Simulação do mosquito <i>Aedes aegypti</i> fêmea: primeiro momento	83
5.6	Simulação do mosquito <i>Aedes aegypti</i> fêmea: segundo momento	83
5.7	Green-House: O cenário de experimentos reais	86
5.8	Green-House: O cenário de experimentos virtuais	87
5.9	Produção acumulada de ovos no período pré-armadilha	93
5.10	Produção quinzenal de ovos no período <i>pré-armadilha</i> - Janela deslizante	93
5.11	Produção acumulada de adultos no período pré-armadilha	94
5.12	Produção quinzenal de adultos no período <i>pré-armadilha</i> - Janela deslizante	95
5.13	Produção acumulada de ovos no período <i>com-armadilha</i>	97
5.14	Produção quinzenal de ovos no período <i>com-armadilha</i> - Janela deslizante	98
5.15	Produção acumulada de adultos no período <i>com-armadilha</i>	99
5.16	Produção quinzenal de adultos no período <i>com-armadilha</i> - Janela deslizante	100
5.17	Bairro Vista Alegre de Belo Horizonte - MG, Brasil	101
5.18	Quarteirão 3545 simulado	103
5.19	Número de ovos capturados	104
5.20	Dinâmica da distribuição de ovos na simulação	105
5.21	Número de ovos capturados na armadilha relacionado com o pluvio	106
5.22	Dinâmica da população de mosquito <i>Aedes aegypti</i> em diferentes fases	106

B.1	Distribuição diária de ovos de <i>Aedes aegypti</i>	123
B.2	Distribuição de ovos de <i>Aedes aegypti</i> durante o dia	124
C.1	Produção diária de ovos no período pré-armadilha	126
C.2	Produção diária de ovos no período pré-armadilha - Dados totalizados par- cialmente	127
C.3	Produção semanal de ovos no período pré-armadilha - Janela fixa	127
C.4	Produção quinzenal de ovos no período pré-armadilha - Janela fixa	128
C.5	Produção diária de adultos no período pré-armadilha	128
C.6	Produção semanal de adultos no período pré-armadilha - Janela fixa	129
C.7	Produção quinzenal de adultos no período pré-armadilha - Janela fixa	129
C.8	Taxa de Picadas	130

Lista de Tabelas

2.1	Trabalhos revisados: modelos matemáticos e baseados em indivíduos	21
5.1	Análise de correlação no período <i>pré-armadilha</i>	90
5.2	Análise de correlação no período <i>com-armadilha</i>	96
B.1	Percentuais de distribuição diária de ovos de <i>Aedes aegypti</i>	124
B.2	Percentuais de distribuição de ovos de <i>Aedes aegypti</i> durante o dia	125

Capítulo 1

Introdução

As doenças infecto-contagiosas como a dengue são uma preocupação para a humanidade. A dengue é transmitida para o homem através do mosquito do gênero *Aedes*. Atualmente, a dengue é endêmica em diversos países e os gastos públicos com o tratamento da doença estão aumentando. O fenômeno do aquecimento global é um fator que afetará a dispersão das populações de mosquitos [Houghton et al., 1997, Peterson, 1998]. Os mosquitos estão aparecendo em altitudes cada vez maiores.

Ainda não existe vacina contra a dengue. O combate à doença é feito principalmente através do controle da população de mosquitos. A principal política de controle da população de mosquitos é o controle dos criadouros o que impede o desenvolvimento do inseto nas fases pré-adultas [World Health Organization, 2002].

Como o problema de controle de epidemias de doenças transmitidas por insetos está ligado ao controle da infestação, uma das principais questões para tomada de decisão é saber: qual é o tamanho da população de mosquitos em uma determinada área? Outra pergunta é: qual é a dinâmica espacial e temporal dessa população? As respostas a estas perguntas são fatores decisivos na escolha de políticas de controle mais eficientes e eficazes.

Atualmente, a estimativa do tamanho da população de mosquitos é feita através da quantidade de larvas encontradas no interior de residências em uma dada região monitorada. Além do controle larvário, outra maneira de estimar o tamanho da população é através da contabilização do número de ovos do mosquito depositados em armadilhas/receptáculos especiais conhecidos como *ovitampa*. Uma terceira maneira seria a contagem direta de indivíduos. Esta última é difícil e dispendiosa.

A meta deste trabalho é propor um modelo e uma ferramenta de simulação computacional da dinâmica populacional dos mosquitos *Aedes aegypti*. Pretende-se simular a população do mosquito presente em pequenas áreas geográficas, como uma casa, um quarteirão ou alguns quarteirões. A partir destas simulações, que recebem como parâmetros

de entrada os dados do controle larvário e das ovitrampas, espera-se melhorar a qualidade dos indicadores de infestação.

A simulação computacional tem sido empregada no auxílio à tomada de decisão e descoberta do conhecimento [Chung, 2003]. Simular um fenômeno através do computador pode ajudar na compreensão deste fenômeno. Essa compreensão se dá quando o ambiente de simulação é utilizado como um laboratório virtual, onde questionamentos do tipo “E se?” podem ser feitos, e respondidos. Além de auxiliar na compreensão do fenômeno estudado, a simulação pode obter respostas para perguntas difíceis de serem respondidas devido a limitações econômicas, ambientais, éticas, tecnológicas e científicas [Peck, 2004].

Diversos trabalhos já foram desenvolvidos com o intuito de avaliar a dinâmica de populações de mosquitos do gênero *Aedes* [Focks et al., 1993b, Focks et al., 1993a] [Yang et al., 2002, Ferreira and Yang, 2003, Yang, 2003] e a dinâmica de transmissão da dengue [Derouich et al., 2003, Leonel and Yoneyama, 2000, Focks et al., 1995]. Alguns desses trabalhos propõem modelos baseados em sistemas de equações diferenciais e outros modelos de simulação computacional. Esses trabalhos utilizam abordagens “macroscópicas”, onde o ambiente é tratado como sendo homogêneo e a população de mosquitos é tratada como uma variável contínua.

Alguns dos modelos de simulação encontrados na literatura modelam a evolução da população, mas não possuem como objetivo simular o comportamento de pequenos grupos de mosquitos dispersos em pequenas áreas. Entretanto, na vida real o controle populacional é feito casa a casa, rua a rua, ou seja, o controle é feito em pequenas regiões. O controle de infestação é sensível às características geográficas e sociais de cada pequena região. Um exemplo de simulador de população de mosquitos do gênero *Aedes* foi proposto por Focks et al. Nesse trabalho é proposto um modelo determinístico de simulação, denominado *CIMSiM*, da dinâmica da vida do mosquito *Aedes aegypti*, o qual produz valores de parâmetros de uma população de mosquitos em uma região de um hectare. *CIMSiM* é basicamente um conjunto de aplicativos estatísticos que mantém informações sobre os grupos de mosquitos *Aedes* e o ambiente em que estão, como temperatura e umidade [Focks et al., 1993b].

Os modelos matemáticos computacionais encontram uma série de dificuldades em capturar detalhes do ambiente. A representação através de equações matemáticas de características qualitativas como preferências e comportamentos, que normalmente são diferentes em indivíduos da mesma espécie, é muito complexa [Parunak et al., 1998].

Procurando superar as limitações dos modelos matemáticos clássicos e outras considerações, diversos trabalhos utilizando modelagem baseada em indivíduos (MBI)¹ têm

¹Do inglês - Individual Based Model - IBM.

sido propostos. Grimm define um *MBI* como um “modelo de simulação que trata cada indivíduo como uma entidade única e discreta que possui idade e pelo menos mais uma propriedade, como por exemplo: peso, posição social, etc.” [Grimm, 1999]. Como consequência, estes modelos permitem a observação de fenômenos e/ou comportamentos emergentes, ou seja, comportamentos globais que não foram pré-programados podem emergir a partir de interações locais [Grimm and Railsback, 2005].

Utilizando *MBIs*, Barrett et al. propuseram um ambiente para simulação de uma epidemia de varíola em uma cidade, chamado *EpiSims*. Esse sistema simula a cidade de Portland, localizada no Estado do Oregon (EUA), e tem por objetivo auxiliar entidades oficiais de saúde pública a identificar uma melhor resposta sobre uma variedade de condições frente a uma situação de epidemia [Barrett et al., 2005].

Considerando a relevância do problema de estimar uma população de mosquitos e as limitações dos atuais modelos e ferramentas desenvolvidas, o presente trabalho apresenta um simulador computacional, composto por um modelo de simulação de população de mosquitos *Aedes aegypti*, denominado *SimPopMosq*. A ferramenta desenvolvida permite realizar simulações de micro-regiões. O modelo de simulação - baseado em indivíduos - *SimPopMosq* é formado por agentes que representam mosquitos *Aedes aegypti*, seres humanos, alguns mamíferos, e objetos, como casas, paredes, vegetação e outros objetos encontrados em ambientes urbanos.

Esta ferramenta utiliza a abordagem de sistemas multiagentes (SMA)². Nesse tipo de abordagem cada indivíduo é modelado e implementado através de um agente computacional capaz de interagir com o ambiente e com outros agentes através de sensores e atuadores [Sandholm, 1999, Wooldridge, 2002, Vlassis, 2003].

O simulador desenvolvido é capaz, por exemplo, de simular uma casa com seus moradores, mosquitos voando nos limites da residência e objetos desta casa. É possível realizar simulações de experimentos científicos, como a simulação de mosquitos *Aedes aegypti* em um ambiente controlado (ex. quarto) para se observar a dinâmica de grupo de mosquitos diante de restrições estabelecidas, como falta de água ou sangue para repasto.

O grau de detalhamento do cenário simulado pode ser aumentado. É possível, por exemplo, modelar e simular um quarteirão, um bairro ou uma cidade. Entretanto, à medida em que o número de agentes e objetos aumenta, mais recursos computacionais são exigidos para que a simulação seja executada.

O simulador de população de mosquitos foi desenvolvido utilizando a linguagem de programação orientada a objetos *Java* e com o auxílio do *framework* de simulações multiagentes *Repast* - versão 3.0.

²Do inglês - Multiagent Systems - MAS.

Com o objetivo de calibrar parâmetros de simulação e validar o modelo proposto um conjunto de experimentos foi realizado. Esses experimentos foram divididos em três grupos.

O primeiro grupo simula alguns mosquitos *Aedes aegypti* em um ambiente fechado, com disponibilidade de alimento, criadouro (água) e parceiro para acasalamento. A partir desse experimento foi verificado, de forma qualitativa, a coerência entre o modelo computacional proposto (machos e fêmeas *Aedes aegypti*) e os dados informados sobre o comportamento e características do mosquito na literatura.

O segundo grupo de simulações é utilizado para calibrar os parâmetros do modelo computacional do mosquito *Aedes aegypti* e validar o modelo. Nesta simulação são reproduzidas as condições de um experimento real. Os dados referentes a este experimento foram fornecidos pelo professor doutor Álvaro Eduardo Eiras, pesquisador do departamento de Parasitologia do Instituto de Ciências Biológicas (ICB) da Universidade Federal de Minas Gerais (UFMG). Os resultados obtidos pela simulação são comparados com os dados do experimento real.

O terceiro grupo consiste de simulações de uma área correspondente a um quarteirão urbano. Este quarteirão foi modelado a partir de dados disponibilizados pelo centro de controle de zoonoses da Prefeitura de Belo Horizonte (MG). Os dados incluem informações sobre a população da região, os imóveis e o posicionamento de armadilhas para controle larvário. Resultados de simulação são comparados com dados obtidos pelo centro de controle de zoonoses.

Os passos intermediários seguidos na construção do simulador de população de mosquitos *Aedes aegypti* originaram algumas contribuições.

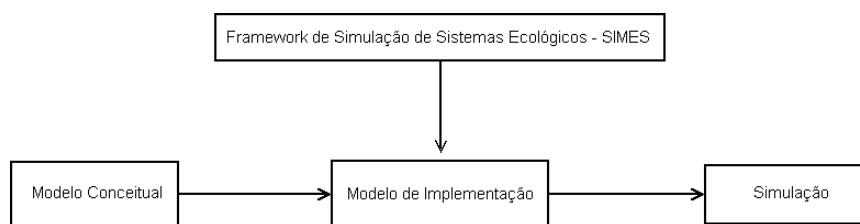


Figura 1.1: Composição do simulador

Na figura 1.1 pode-se observar as seguintes contribuições:

i) Na modelagem conceitual do simulador de população de mosquitos foram definidos e adotados modelos de agentes e objetos e ambiente que fazem parte do simulador. Dentre estes modelos, a principal contribuição foi a definição estruturada de um modelo de tomada de decisão para o mosquito *Aedes aegypti*. Até a presente data, nenhum modelo de tomada de decisão similar foi encontrado na literatura. Um modelo de tomada de

decisão é útil para compreender o comportamento do mosquito em diferentes situações.

ii) Uma extensão do *framework* de simulações multiagentes *Repast* foi desenvolvida. Essa extensão, denominada *framework de Simulação de Sistemas Ecológicos - SIMES*, possui algumas classes - implementadas na linguagem de programação *Java* - que tendem a maximizar a reusabilidade de código e padronização do desenvolvimento de aplicações. De forma geral, *MBIs* relacionados à ecologia podem ser implementados adotando o *framework SIMES*.

iii) A implementação do modelo conceitual proposto em linguagem de programação orientada a objetos, obedecendo à padronização do *framework SIMES*, resultou em um modelo de implementação que permite a simulação de população de mosquitos *Aedes aegypti*.

iv) A partir do modelo de implementação definido e de um cenário de simulação previamente especificado, foi possível realizar simulações computacionais e obter resultados que podem contribuir na tomada de decisão dos usuários do simulador. Atualmente, a ferramenta de simulação desenvolvida possui um repositório com alguns cenários de simulação que foram utilizados na validação e ilustração de seu funcionamento.

Os próximos capítulos estão estruturados da seguinte maneira: no capítulo 2 é realizada uma revisão bibliográfica dos assuntos relacionados ao trabalho desenvolvido. No capítulo 3, o modelo conceitual do simulador desenvolvido é descrito através do protocolo *ODD* (*Overview, Design concepts and Details*) [Grimm et al., 2006]. No capítulo 4 é descrito em termos técnicos e de implementação o simulador proposto e seus componentes. No capítulo 5 são apresentados os resultados dos experimentos computacionais realizados, relacionando-os ao processo de construção e validação dos modelos propostos. Por fim, no capítulo 6 são apresentados comentários e conclusões obtidas.

Capítulo 2

Referencial Teórico

Neste capítulo são apresentados os trabalhos relacionados ao trabalho desenvolvido e aspectos teóricos relacionados aos temas *Modelos baseados em indivíduos*, *Sistemas multiagentes*, *Dengue* e *Mosquito Aedes aegypti*.

2.1 Modelos baseados em indivíduos

Existem diversas definições para o termo “Modelo Baseado em Indivíduo” (MBI). Grimm define *MBI* como um “modelo de simulação que trata cada indivíduo como uma entidade única e discreta que possui idade e pelo menos mais uma propriedade, como por exemplo peso, posição social, etc.” [Grimm, 1999].

É comum o uso de outros termos que se relacionam parcialmente ou totalmente aos *MBI*. Pode-se citar os termos *Modelo Baseado em Agentes*, *Sistemas Multiagentes* e *Autômatos Celulares*. Na próxima seção o tema *Sistemas Multiagentes* é abordado.

Uma das principais aplicações dos *MBIs* é simular fenômenos. Estas simulações podem ter fins diferentes, como o auxílio na compreensão de fenômenos de difícil observação na prática [Grimm and Railsback, 2005]. Estes modelos, se validados, também podem ser utilizados para predição. Os *MBIs* também são utilizados na indústria de jogos digitais [Reynolds, 2007b] e do cinema [Massive software, 2007]. Reynolds cita como exemplo algumas cenas do filme “O Rei Leão”, criado pelos estúdios *Disney* [Reynolds, 2007a].

De acordo com Parunak et al., em muitos domínios os *MBIs* competem com modelos baseados em equações (MBE), que agrupam um conjunto de variáveis inter-relacionadas através de equações diferenciais. As principais diferenças entre estas abordagens estão na forma dos modelos e como os modelos são simulados. No *MBI* existem um ou mais agentes que encapsulam o comportamento de vários indivíduos, enquanto no *MBE* o modelo é um conjunto de equações e a execução consiste na resolução sistemática dessas equações

[Parunak et al., 1998].

Os modelos matemáticos computacionais encontram uma série de dificuldades em capturar detalhes dos fenômenos reais. Representar através de equações matemáticas características qualitativas dos modelos, como preferências e comportamentos, que normalmente são diferentes em indivíduos da mesma espécie, é muito complexo e se torna quase impossível de ser realizado [Parunak et al., 1998].

Os *MBIs* são apropriados para domínios caracterizados com detalhes, que inclui localização, distribuição discreta de indivíduos e decisões discretas [Parunak et al., 1998]. Bonabeau indica os seguintes benefícios que os *MBI* apresentam em relação a outros tipos de modelos [Bonabeau, 2002]:

1. *Captura de fenômenos emergentes* - “Fenômenos emergentes resultam da interação de entidades individuais. A totalidade do sistema é maior que a soma das partes devido a interação entre essas partes. Assim, um fenômeno emergente pode ter propriedades que são desacopladas das propriedades das partes” [Bonabeau, 2002].
2. *Descrição natural do sistema* - Em muitos casos descrever e implementar um *MBI* pode ser mais natural do que lidar com equações diferenciais que possuem grande números de parâmetros e operadores relacionados.
3. *Flexibilidade* - Os *MBIs* são flexíveis. A adição de novos agentes ao modelo pode ser simples. Aumentar ou diminuir o grau de detalhe de determinado sub-modelo também pode ser uma tarefa simples.

Uma das principais dificuldades dos *MBIs* está na definição dos parâmetros do modelo. Alguns parâmetros não têm valores reais conhecidos. A definição incorreta desses parâmetros pode levar o modelo a se comportar diferentemente da realidade estudada. A fase de calibração de parâmetros é uma das mais complexas e importantes na construção de um *MBI* [Grimm and Railsback, 2005]. Outra dificuldade enfrentada por desenvolvedores de *MBIs* está na definição do grau de detalhamento suficiente para representar o fenômeno desejado. Muitos detalhes podem ser simplificados sem alterar de forma significativa a dinâmica do modelo desenvolvido [Peck, 2004, Chung, 2003, Grimm and Railsback, 2005].

Um problema comum em alguns dos modelos de simulação encontrados na literatura está relacionado à validade dos modelos. Diversos trabalhos apresentam modelos complexos sem validade experimental cuidadosa.

A *ecologia* é uma das principais áreas que adotam *MBIs*. Grimm, um dos principais pesquisadores deste assunto, afirma que “na ecologia os indivíduos são organismos que possuem propriedades, crescem, desenvolvem-se e adaptam-se durante o ciclo de vida.

Indivíduos reproduzem e morrem” [Grimm and Railsback, 2005]. Ainda segundo Grimm, um ponto comum entre os *MBI* e a ecologia é que as propriedades e comportamentos dos indivíduos determinam as propriedades do sistema. Esta é uma das principais justificativas para se explorar sistemas ecológicos através de *MBI*.

A publicação de *MBI* na comunidade científica normalmente é feita pelo seus desenvolvedores, sem adotar um padrão de descrição. Recentemente foi proposto por Grimm et al. o *protocolo padrão ODD* (*Overview, Design concepts and Details*) para descrição - em termos conceituais - de *MBIs* [Grimm et al., 2006]. A descrição conceitual do modelo de simulação proposto neste trabalho é realizada no capítulo 3, seguindo as especificações do protocolo *ODD*.

2.2 Sistemas multiagentes

Segundo Rich e Knight, a inteligência artificial distribuída (*IAD*) é uma sub-área da inteligência artificial (*IA*), onde a busca por soluções considera aspectos de individualidade que emergem no sentido coletivo. Enquanto a *IA* clássica define modelos de tomada de decisão que sejam capazes de resolver problemas de forma eficiente (resultados precisos e obtidos rapidamente), a *IAD* se preocupa em definir modelos de raciocínio artificial que sejam eficientes e adotem o paradigma tomada de decisão distribuído [Rich and Knight, 1991].

Sichman afirma que o foco da *IAD* está na interação e no comportamento social, isto é, aborda-se o problema considerando a possível construção de uma inteligência “grupal”. Adota-se agentes inteligentes autônomos para representar os indivíduos do modelo [Sichman, 2003]. Ainda segundo Sichman, a *IAD* pode ser dividida em duas sub-áreas: resolução distribuída de problemas (*RDP*) e sistemas multiagentes (*SMA*). Na área de *RDP* estuda-se formas de dividir um problema em partes que cooperam no desenvolvimento de uma solução. Nos *SMA* estuda-se mecanismos de interação entre agentes e os efeitos destas interações [Sichman, 2003].

Faraco afirma que na *RDP* as entidades que interagem, os agentes, são “programados” para realizar tarefas específicas e para se comunicarem com outros agentes da sociedade a fim de atingir o objetivo global proposto [Faraco, 1998]. Neste tipo de abordagem, a aplicação da computação paralela e distribuída tende a melhorar o tempo de processamento da solução, uma vez que as “partes do problema” podem ser divididas e distribuídas entre outros computadores [Rich and Knight, 1991]. A figura 2.1 ilustra a estrutura de um sistema de *RDP*.

De acordo com Sichman não existe uma definição com semântica precisa para *SMA*.

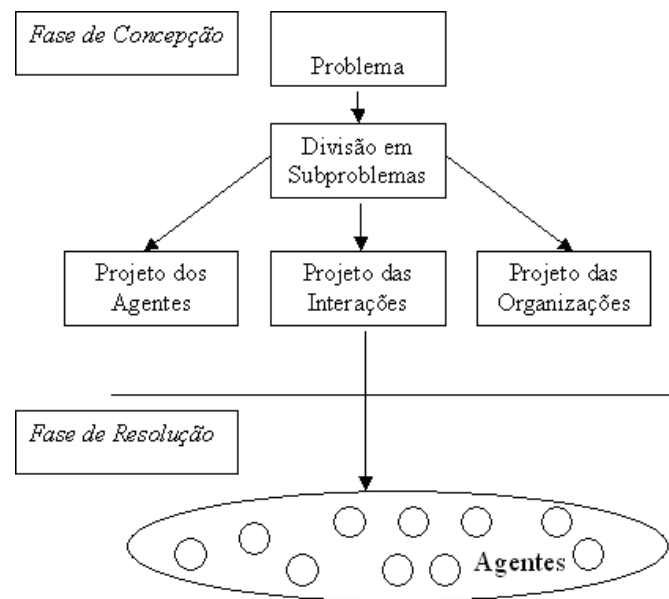


Figura 2.1: Estrutura de um sistema *RDP* [Sichman, 2003]

Isto se deve ao fato dos *SMA* serem utilizados por diversas áreas como *IA*, ecologia, sociologia, economia, administração e filosofia [Sichman, 2003].

Wooldridge define *SMA* como sistemas compostos pela múltipla interação de elementos computacionais, conhecidos por agentes [Wooldridge, 2002]. Considerando esta definição, os *SMA* podem ser considerados um tipo de modelo baseado em indivíduo, onde os indivíduos do modelo são implementados através de agentes computacionais.

Dentre as diversas aplicações dos *SMA*, destacam-se as seguintes:

1. *Simulação computacional* - Por se tratar de um modelo baseado em indivíduo, os *SMA* podem ser utilizados para simulação computacional. Diversos trabalhos têm utilizado este tipo de abordagem para simulações, como por exemplo simulações de pedestres, propagação de vírus, veículos, centros urbanos, etc.
2. *Entretenimento* - A indústria de jogos tem utilizado sistemas multiagentes para representar indivíduos em um cenário. Em um jogo eletrônico que incorpora este tipo de abordagem, o jogador real (pessoa) pode interagir com agentes inteligentes no ambiente virtual. Estes agentes podem ser implementados através de *SMA*.
3. *Processo de tomada de decisão distribuída* - A descentralização da tomada de decisão em determinados cenários tem se mostrado eficiente. Bertsekas propõe um modelo de tomada de decisão distribuído baseado no modelo econômico de um *leilão* [Bertsekas, 1992]. Neste modelo, cada agente envolvido no *leilão* efetua um

lance a um item que *mais lhe satisfaz*, e como consequência, ficam atribuídos a determinado item o agente participante do leilão que tiver maior interesse e condição econômica pelo referido item. O modelo proposto é aplicado na resolução de alguns problemas de otimização, como o problema do fluxo mínimo em redes [Bertsekas and Catañon, 1991], o problema atribuição e de transporte [Bertsekas and Catañon, 1989], o problema do menor caminho [Bertsekas, 1991], entre outros problemas [Bertsekas, 1988]. Este tipo de abordagem pode ser implementado através de um SMA.

4. *Segurança em redes* - Ultimamente alguns trabalhos têm adotado SMA para monitoramento de pacotes de dados em redes de comunicação de dados. Nessas aplicações, agentes computacionais inspecionam a rede. Eventuais tentativas de ataque por invasores são detectadas por esses agentes, que trabalhando de forma cooperativa tentam inibir o invasor [Zaki and Sobh, 2005].

2.2.1 Agentes

De acordo com Russell e Norvig 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 [Russell and Norvig, 2003]. Esta idéia é ilustrada na figura 2.2.

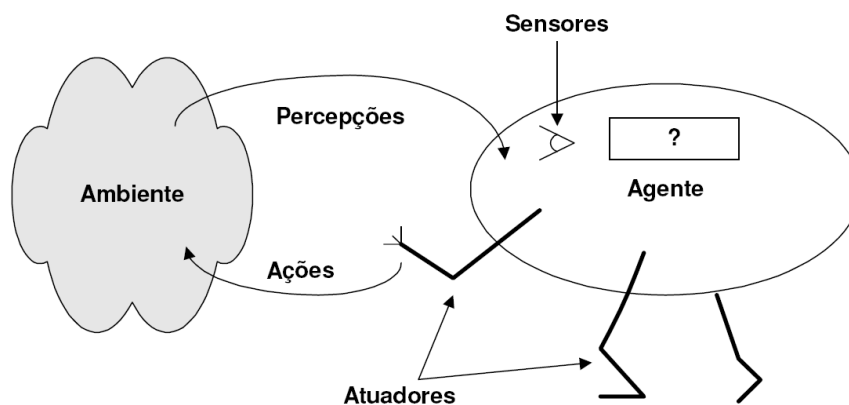


Figura 2.2: Agentes interagindo com o ambiente [Russell and Norvig, 2003]

Wooldridge e Jennings considera que agentes inteligentes podem ser caracterizados pela presença das seguintes capacidades [Wooldridge and Jennings, 1994]:

1. *Autonomia*: o agente deve poder funcionar sem a intervenção do homem, buscando suas ações em seu conhecimento armazenado sobre o ambiente.

2. *Reatividade*: ao perceber o ambiente, o agente deve ser capaz de responder as mudanças em tempo hábil para atingir seus objetivos.
3. *Habilidade social*: o agente interage com outros agentes através de uma linguagem de maneira a satisfazer seus objetivos.
4. *Pró-atividade*: O agente deve atuar maximizando o cumprimento de seus objetivos.

Além destas capacidades apresentadas, Fernandes discorre sobre as seguintes características que podem estar presentes nos agentes [Fernandes, 2005]:

1. *Mobilidade* - Alguns agentes podem se mover através de um ambiente.
2. *Cooperação* - Os agentes podem cooperar com outros agentes na medida em que trabalham juntos e visam atingir um objetivo comum.
3. *Comunicabilidade* - Segundo Franklin e Graesser, os agentes podem se comunicar com outros agentes, objetos, o próprio ambiente ou ainda com seres humanos [Franklin and Graesser, 1996].
4. *Aprendizagem* - Os agentes podem aprender. Com o aprendizado podem tomar decisões mais apropriadas em relação a seus objetivos.

Existem diversas classificações para agentes [Sichman, 2003, Wooldridge, 2002] [Fernandes, 2005]. As classificações são determinadas com base nas propriedades dos agentes. Nas classificações encontradas em [Sichman, 2003, Wooldridge, 2002] os agentes são divididos em:

1. *Agentes reativos* - Sichman afirma que os agentes reativos são normalmente baseados em modelos de sociedades biológicas, como sociedades de abelhas e de formigas. Os agentes reativos funcionam segundo um modelo de estímulo-resposta. A representação do ambiente não é explícita. Os agentes não possuem memória e nem planejam o futuro. As sociedades de agentes reativos normalmente possuem um grande número de agentes (ordem de milhares) [Sichman, 2003].
2. *Agentes cognitivos* - De acordo com Sichman os agentes cognitivos são baseados em modelos de sociedades humanas, como grupos, hierarquias e mercados. Possuem uma representação explícita do ambiente, memória do passado e normalmente planejam o futuro. Interagem com outros agentes através de mensagens e podem aprender com os resultados das suas ações. Tipicamente, as sociedades de agentes cognitivos possuem poucos agentes - entre 10 e 20 agentes [Sichman, 2003].

3. *Agentes híbridos* - Os agentes híbridos são agentes que apresentam simultaneamente características de agentes reativos e cognitivos. Esses agentes podem ainda incorporar outras características, como mobilidade.

2.2.2 Engenharia de software orientada a agentes

Atualmente é crescente o desenvolvimento de aplicações baseadas em SMA. Como consequência, a engenharia de software também está se desenvolvendo para atender esse novo paradigma. Recentemente emergiu a área de engenharia de software orientada a agente (ESOA), com a missão de ajudar nas etapas de análise, projeto, especificação e implementação de soluções baseadas em agentes [Jennings and Wooldridge, 1999].

Alguns trabalhos desenvolvidos pela ESOA incluem a construção de *frameworks* conceituais (ontologias) [Silva and Lucena, 2004], especificação de linguagens de modelagem [Bauer et al., 2001, da Silva and de Lucena, 2003], definição de padrões de projeto [Sauvage, 2004], metodologias de desenvolvimento de softwares [Zambonelli et al., 2003], *frameworks* para construção e simulação de sistemas baseados em agentes e outras tecnologias.

Apesar de todo este desenvolvimento da ESOA, a aplicação prática dos estudos desenvolvidos por esta área tem se mostrado pouco expressiva, com excessão dos *frameworks* para construção e simulação de SMA [Amor et al., 2005]. Estes *frameworks* diminuem o trabalho dos desenvolvedores na medida que fornecem bibliotecas de classes que impactam na diminuição da codificação dos sistemas.

2.2.3 Frameworks de simulação multiagentes

Atualmente existem esforços para padronizar o desenvolvimento de SMA. Uma das principais entidades que padroniza modelos de construção de SMA é a *Foundations for Intelligent Physical Agents* (FIPA) [FIPA, 1996]. Esta entidade é uma associação internacional, sem fins lucrativos, formada por organizações que produzem especificações genéricas para tecnologias de agentes. A FIPA não foca em desenvolvimento de tecnologia específica para uma aplicação, mas na especificação de tecnologias genéricas para diversas áreas de aplicação independente da plataforma utilizada de maneira a garantir interoperabilidade entre diferentes plataformas [FIPA, 1996].

As especificações produzidas pela FIPA baseiam-se em duas suposições. A primeira é que o tempo necessário para se projetar e implementar estas especificações não deve ser longo. A segunda é que somente comportamentos externos dos componentes devem ser especificados, deixando detalhes de implementação e arquiteturas internas para o

desenvolvedor dos agentes [FIPA, 1996].

Existem diversos *frameworks* para construção de simulações baseadas em agentes. Pode-se citar os *frameworks*: i) Swarm [Minar et al., 1996]; ii) Jade [Bellifemine et al., 1999]; iii) MASON [Luke et al., 2005]; iv) Repast [Collier, 2000, North et al., 2006]; v) NetLogo [NetLogo, 2007]. Existem dezenas de *frameworks* semelhantes aos citados.

Railsback et al. apresentam um estudo comparativo entre *frameworks* de simulação multiagente [Railsback et al., 2006]. Este estudo compara alguns *frameworks* sob aspectos de desempenho computacional, dificuldade de utilização, disponibilidade de documentação e escalabilidade. Algumas sugestões quanto a escolha de *frameworks* para determinadas situações são apresentadas.

O *framework* *Swarm* foi o primeiro a ser desenvolvido para construção de simulações multiagentes. Inicialmente suas bibliotecas foram escritas na linguagem de programação *Objective-c*. Atualmente, os desenvolvedores do *Swarm* disponibilizam uma versão desta ferramenta na linguagem de programação *Java* em [Swarm Development Group, 2004]. O *Swarm* é distribuído sob licença *GNU* (General Public License), o que obriga os desenvolvedores a tornarem o código de sua aplicação aberto.

Os *frameworks* *MASON* (Multi-Agent Simulator Of Neighborhoods) e *NetLogo* são mais recentes. Estas ferramentas permitem que os desenvolvedores de agentes criem suas aplicações em *alto nível*, ou seja, o desenvolvedor não precisa se preocupar com aspectos muito técnicos de programação. O *framework* *NetLogo* possui uma linguagem de programação própria, enquanto o *MASON* possibilita a criação de agentes na linguagem de programação *Java*.

JADE (Java Agent DEvelopment Framework) é um *framework* baseado nas especificações FIPA. Segundo Bellifemine et al. [Bellifemine et al., 1999], o objetivo desse *framework* é simplificar o desenvolvimento, assegurando o padrão FIPA através de um conjunto de serviços de agentes e ferramentas. Para atingir este objetivo o *framework* *JADE* oferece as seguintes funcionalidades para programadores de agentes:

- Plataforma de agentes básicos, que incluem os agentes: *Agent Management System* (AMS), *Directory Facilitator* (DF) e *Agent Communication Channel* (ACC). Todos esses agentes são baseados na especificação de alguns agentes definidos pela FIPA.
- *Plataforma de agentes distribuídos* - A plataforma de agentes pode ser composta por diversas sub-plataformas. Cada sub-plataforma pode executar uma instância do aplicativo *JADE* em um computador diferente. Os agentes são implementados como uma *thread Java* e eventos são utilizados para efetivar a comunicação entre os agentes de uma mesma plataforma.

- Serviços de “páginas amarelas” podem ser inicializados, de maneira a permitir a comunicação entre os agentes.
- Ferramentas gráficas para facilitar registros e manutenção de agentes de serviços pertencentes a um ou mais domínios.
- Mecanismos de transporte e interface gráfica para envio e recebimento de mensagens de um agente para outro.
- Utilização do protocolo *IIOP* (Internet Inter-ORB Protocol) para conectar diferentes plataformas.
- Registro automático de agentes através do *AMS*.
- Interface gráfica para monitoramento de agentes na plataforma.

A execução de cada agente no *framework JADE* é realizada sob uma *Thread* do sistema operacional. Isto limita a capacidade de escalabilidade das aplicações desenvolvidas utilizando o *JADE*. Desta forma, esse *framework* é aconselhável para o desenvolvimento de aplicações com número reduzido de agentes [Railsback et al., 2006].

O *framework Repast* (The Recursive Porous Agent Simulation Toolkit) herdou vários conceitos do *framework Swarm* [Collier, 2000]. Entretanto, o *Repast* é diferente do *Swarm* em vários aspectos. O *Repast* é disponível em *Java* e *Microsoft.NET*. Este *framework* é distribuído utilizando uma variação da licença *BSD* (Berkeley Software Distribution), que não exige a liberação do código-fonte por parte dos desenvolvedores de aplicações baseadas em agentes. O *Repast* fornece um conjunto de bibliotecas integradas para *redes neurais*, *algoritmos genéticos*, modelagem de simulações sociais e outros tópicos [Collier, 2000, North and Macal, 2005].

O *Repast* foi desenvolvido na Universidade de Chicago (EUA) e é mantido pelo laboratório *Argonne National Laboratory* (EUA). Esse *framework* possui código aberto. North e Macal discorrem sobre as seguintes características do *framework Repast* [North and Macal, 2005]:

- O *Repast* é totalmente orientado a objetos.
- O *Repast* inclui bibliotecas para configuração de escalonamento de eventos discretos.
- O *Repast* oferece recursos de registro de variáveis dinâmicas e exibição gráfica dos valores das mesmas.
- O *Repast* possui bibliotecas para automação de simulação Monte Carlo.

- O *Repast* permite o acesso e modificação de propriedades de agentes e do modelo de simulação em tempo de execução.
- O *Repast* inclui bibliotecas para geração de números aleatórios, regressão linear, algoritmos genéticos e redes neurais.
- O *Repast* possui ferramentas de auxílio a simulações de redes sociais.
- O *Repast* fornece suporte a integração de sistemas de informações geográficas.
- O *Repast* permite implementações de modelos nas linguagens disponíveis no *framework* Microsoft.NET: C#, Managed C++, Visual Basic.NET, Managed Lisp, Managed Prolog, Managed Cobol. Além destas linguagens é possível implementar modelos na linguagem de programação Python scripting.
- O *Repast* é disponível nas plataformas Windows, Mac OS e Linux.

O *Repast* possui duas camadas. A camada “núcleo” é responsável pela execução dos componentes internos, não visíveis para os usuários. A camada mais externa é relacionada ao modelo de simulação desenvolvido pelo usuário do *framework* [North and Macal, 2005]. A figura 2.3 ilustra os componentes envolvidos nas simulações realizadas com o *framework Repast*.

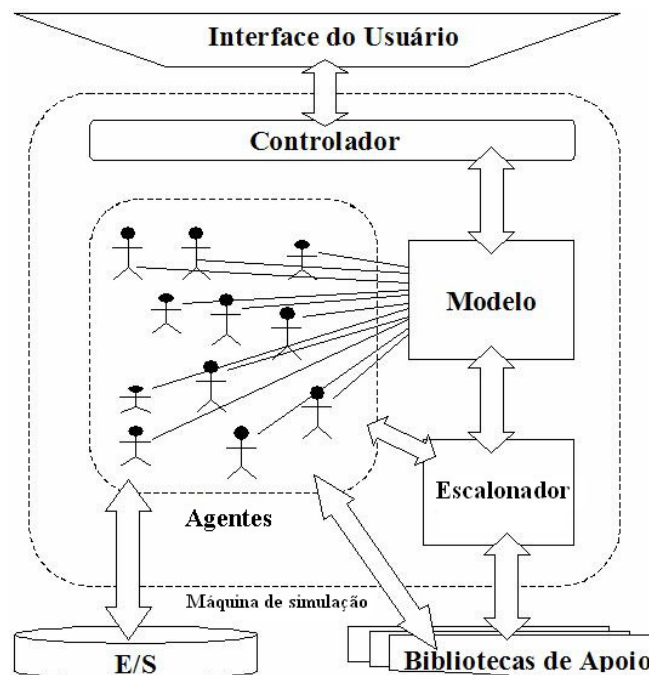


Figura 2.3: Visão geral do *framework Repast* [North and Macal, 2005]

De acordo com North e Macal existem quatro componentes na figura 2.3 que são implementados na camada interna do *Repast* e utilizados pelas camadas externas (modelo de simulação do usuário) [North and Macal, 2005]. Estes componentes são:

1. *Máquina de simulação* - A máquina de simulação ¹ é responsável pela execução das simulações. Esta máquina possui as seguintes partes:
 - (a) *Escalonador*: responsável pela definição das regras de escalonamento de eventos/comportamentos de agentes e objetos.
 - (b) *Modelo*: contém a definição da simulação e seus elementos.
 - (c) *Controlador*: conecta o modelo de simulação ao escalonador da plataforma de simulação.
 - (d) *Agentes*: são definidos pelo usuário e adicionados na simulação.
2. *Entrada e saída de dados* - Este mecanismo é responsável pela atribuição de valores definidos por usuários às propriedades do modelo de simulação. Este mecanismo é responsável também pelo registro de valores de variáveis em memória secundária.
3. *Interface do usuário* - Este componente permite a exibição de resultados do modelo de simulação e permite a interação do usuário com o modelo de simulação.
4. *Bibliotecas de apoio* - O *Repast* fornece bibliotecas de suporte à modelagem e matemática.

A figura 2.4 ilustra a execução de um modelo de simulação conhecido como *HeatBugs* utilizando o *framework Repast* [North et al., 2006].

¹O termo *máquina de simulação* foi traduzido do termo em inglês: *Simulation engine*

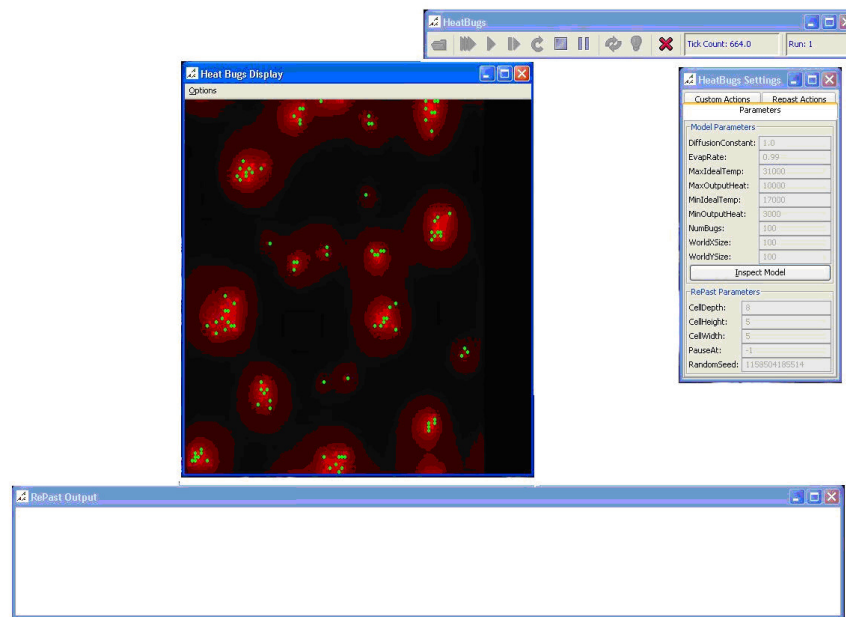


Figura 2.4: Modelo de simulação *HeatBugs*

No modelo de simulação *HeatBugs* alguns insetos são colocados em um cenário inicial. Conforme o tempo passa, os agentes tendem a aproximar-se ou distanciar-se na tentativa de manter uma temperatura média. Como consequência, observa-se grupos de agentes sendo formados [North et al., 2006].

2.3 Trabalhos relacionados

Diferentes modelos que descrevem a dinâmica de populações de insetos e propagação de doenças foram revisados durante o desenvolvimento deste trabalho. Seguindo a classificação de modelos sugerida em [Parunak et al., 1998] e [Grimm and Railsback, 2005], os modelos revisados foram divididos em dois grupos a saber: modelos matemáticos e modelos baseados em indivíduos. A seguir alguns destes modelos são apresentados.

2.3.1 Modelos matemáticos

Um modelo pioneiro que modela a relação entre espécies foi proposto por Vito Volterra e Alfred J. Lotka em 1910 [Flake, 2000]. Este modelo é conhecido como *Lotka-Volterra*. A partir desse trabalho, a modelagem matemática da relação entre espécies começou a se disseminar na ciência. Bailey em [Bailey, 1975] adotando uma abordagem matemática análoga ao modelo *Lotka-Volterra* definiu modelos de propagação de doenças infecto-

contagiosas.

A partir das idéias propostas em [Bailey, 1975] e da teoria de controle ótimo [Swan, 1984], Leonel e Yoneyama propuseram um modelo matemático para o *controle ótimo* e *sub-ótimo* em epidemias de dengue [Leonel and Yoneyama, 2000]. No modelo proposto, investimentos, como campanhas educacionais e utilização de inseticidas são relacionados ao custo de tratamento da dengue. Desta forma, simulações computacionais baseadas em equações diferenciais ordinárias foram realizadas, considerando investimentos variados em educação e inseticidas, e, como consequência, resultados relacionados a índices de infestação de mosquitos *Aedes aegypti* (índice de Breteau [Gomes, 1998]) foram obtidos. Estes índices estão diretamente relacionados aos custos com o tratamento da dengue. O resultado mostra que uma política sub-ótima gera uma redução substancial nos custos com saúde pública, e é pouco inferior à política de controle ótimo. Além disso, Leonel e Yoneyama afirmam que a implantação de uma política de controle ótimo mostraria-se inviável na prática, uma vez que exigiria investimentos elevados e contínuos no combate à doença [Leonel and Yoneyama, 2000].

Derouich et al. desenvolveram modelos matemáticos onde dois tipos de vírus coexistem, possibilitando o surgimento da dengue clássica e da dengue hemorrágica simultaneamente. Um estudo sobre a dinâmica da doença foi realizado através de modelos comportamentais envolvendo equações diferenciais ordinárias para a população de seres humanos e mosquitos *Aedes aegypti*. Algumas estratégias de combate à dengue foram discutidas a partir da análise de simulações realizadas, onde foram considerados diferentes valores de parâmetros [Derouich et al., 2003].

Yang et al. propõem um estudo da dinâmica populacional do mosquito da dengue através do uso de equações diferenciais [Yang et al., 2002]. Foram abordadas fases da vida do mosquito *Aedes aegypti* e mecanismos de controle adotados pelas entidades de saúde. Os autores estudam “a dinâmica da população de mosquitos transmissores da dengue através de modelos matemáticos autônomos. Dependendo da hipótese admitida para a capacidade de oviposição eles mostraram a existência ou não de condições para a eliminação dos mosquitos por meio de controle aplicado na sua população” [Yang et al., 2002].

Um estudo complementar da dinâmica populacional do mosquito *Aedes aegypti* é realizado em [Ferreira and Yang, 2003]. Nesse trabalho mostra-se o impacto de diferentes formas de controle utilizadas pela SUCEN (Superintendência de Controle de Endemias) da cidade de São Paulo - SP. Os mesmos autores possuem outros trabalhos correlacionados. Em [Yang, 2003] é realizado um estudo matemático através de equações diferenciais demonstrando a dinâmica da transmissão da dengue entre seres humanos e o vetor transmissor da dengue, o mosquito *Aedes aegypti*. Resultados indicam o impacto da dengue

na população de seres humanos diante de estratégias adotadas no controle de população de mosquitos.

Takahashi et al. apresentam um modelo contínuo que descreve a dinâmica de dispersão do mosquito *Aedes aegypti*. Esse trabalho teve como objetivo permitir avaliar a influência da dispersão ativa ² e passiva ³ do mosquito frente a diferentes possíveis estratégias [Takahashi et al., 2005].

Marcelo Otero e Schweigmann propuseram um modelo estocástico que descreve a dinâmica da população de mosquitos *Aedes aegypti* em regiões de clima temperado. Utilizando dados climáticos (principalmente temperatura) da capital argentina Buenos Aires e parâmetros ecológicos relacionados ao mosquito *Aedes aegypti* diferentes saídas do modelo desenvolvido são analisadas e distribuições geográficas de populações de mosquitos em função da temperatura são sugeridas [Otero et al., 2006].

Focks et al. desenvolveram um modelo determinístico de simulação, denominado *CIMSiM*, da dinâmica da vida do mosquito *Aedes aegypti* [Focks et al., 1993b]. Este modelo produz estimativas de vários parâmetros para simulações de grupos de mosquitos *Aedes aegypti*, em um ambiente de aproximadamente 1 ha. O modelo *CIMSiM* mantém informações sobre idade, desenvolvimento em relação a temperatura e tamanho, peso, taxa de fecundação e outras características para cada grupo simulado, dependendo do estágio de vida em que os mosquitos se encontram. Baseado nestas informações internas, *CIMSiM* é capaz de fornecer algumas respostas de forma dinâmica, como o número de indivíduos que irão passar para a próxima fase da vida. Por se tratar de um modelo macroscópico, que considera grupos populacionais de mosquitos, detalhes que podem influenciar as simulações não são considerados na modelagem do sistema.

Em [Bar-Zeev, 1957] são conduzidos experimentos laboratoriais para investigar propriedades dos mosquitos *Aedes aegypti* [Focks et al., 1993a]. Estas propriedades relacionam-se às fases antecedentes à idade adulta do mosquito. Diante dos dados obtidos nestes experimentos, a partir do modelo *CIMSiM* simulações foram realizadas e os seus resultados foram comparados com os dados do experimento laboratorial. Os dados reais e as saídas do modelo de simulação *CIMSiM* se mostraram coerentes.

Posteriormente, Focks et al. incrementaram no modelo *CIMSiM* a fase de contaminação por dengue entre mosquitos e seres humanos. O modelo resultante deste incremento é denominado *DENSiM* [Focks et al., 1995]

Utilizando o simulador *CIMSiM*, Maguire et al. apresentam simulações da dinâmica

²Dispersão ativa é um fenômeno que permite a movimentação intencional do mosquito.

³Dispersão passiva é um fenômeno que leva o mosquito a se movimentar sem intenção. Por exemplo, correntes de ar podem mudar a trajetória de um mosquito.

da população de mosquitos *Aedes aegypti* realizadas na cidade de Townville, localizada no estado de Queensland, região Nordeste da Austrália. Os resultados destas simulações indicam o possível efeito da utilização de algumas políticas de controle do mosquito, como aplicação de inseticida e eliminação de reservatórios de água [Maguire et al., 1999].

Dentre os trabalhos apresentados nesta seção relacionados com a dinâmica de população de mosquitos *Aedes aegypti*, somente o trabalho desenvolvido por [Focks et al., 1993b] teve o modelo validado com dados reais.

2.3.2 Modelos baseados em indivíduos

Simulações baseadas em indivíduos estão sendo utilizadas por diversas áreas do conhecimento. Guo et al. propõem um trabalho voltado para a área de modelagem de sistemas imunológicos, onde foi modelado um sistema multiagentes para simular os mecanismos que o vírus HIV utiliza no organismo infectado, permitindo que os observadores abstraíam relações de comportamento do vírus com o sistema imunológico [Guo et al., 2005].

O trabalho proposto por Barrett et al. utiliza modelos baseados em indivíduos para simulação de pessoas e de uma possível epidemia de varíola [Barrett et al., 2005]. O sistema proposto, chamado *EpiSims*, tem por objetivo auxiliar entidades oficiais de saúde pública a identificarem uma melhor resposta sobre uma variedade de condições frente a uma situação de epidemia de varíola originada de um ataque terrorista com armas biológicas. O *EpiSims* foi criado pelo *Los Alamos National Laboratory* a partir do simulador de tráfego de veículos terrestres *TranSims*. Ambos os sistemas simulam a cidade de Portland (Oregon-EUA), que possui 2 milhões de habitantes.

Em 2004, a agricultura de alguns países do leste Asiático foi surpreendida pela morte de milhares de aves causada pelo vírus *H5N1*. Autoridades de saúde mundiais se mobilizaram para o problema, pois existia o risco da doença causada pelo vírus, conhecida como *gripe do frango*, ser transmitida para o ser humano, e com isto, uma epidemia da doença se tornar mundial. Situngkir propõe um estudo da dinâmica espacial epidemiológica da doença. Os mapas de simulações gerados para esse estudo foram baseados em autômatos celulares e indicam possível efeito da dispersão desta doença [Situngkir, 2005].

A malária é uma doença antiga e mortal entre seres humanos. Depinay et al. em apresentam um modelo de simulação do mosquito Africano *Anopheles*, transmissor da malária. Este modelo é baseado em indivíduos e incorpora mecanismos da dinâmica populacional do mosquito *Anopheles* e sua relação com o meio ambiente. Simulações indicam padrões similares aos encontrados “em campo”. Análises de sensibilidade do modelo em relação à temperatura, umidade, competição por nutrientes e predação são

conduzidas [Depinay et al., 2004].

Pessoa et al. descrevem o sistema desenvolvido *EpiCASim* que utiliza autômatos celulares probabilísticos para simulação de propagação da dengue [Pessoa et al., 2004]. No modelo deste sistema considerou-se a transmissão da dengue entre homens e mosquitos *Aedes aegypti*. Internamente foram criadas duas grades (*Grid Cellular Automata*), sendo que uma contém regras de transição dos seres humanos modelados e o outros possui regras de transição dos mosquitos *Aedes aegypti* modelados. A sobreposição das células da primeira grade com as da segunda caracteriza a contaminação do ser humano pelo vírus da dengue, através da picada do mosquito *Aedes aegypti*. As regras de transição adotadas baseiam-se no trabalho desenvolvido por [Alves et al., 2003]. Ao final do trabalho, mostrou-se que o comportamento do *EpiCASim* foi relativamente semelhante ao modelo matemático da transmissão da dengue, proposto por [Newton and Reiter, 1992].

Nesta seção, nenhum modelo de simulação de mosquitos *Aedes aegypti* foi apresentado. Até a presente data nenhum simulador que envolva modelos baseados em indivíduos e mosquitos *Aedes aegypti* foi publicado. Esta é uma das motivações que levou a escolha do tema deste trabalho.

A tabela 2.1 resume os trabalhos revisados nas seções *Modelos matemáticos* e *Modelos baseados em indivíduos*.

Modelos	Modelos Matemáticos	Modelos Baseados em Indivíduos
Modelos biológicos	[Lotka-Volterra, 1910]	[Guo et al., 2005]
Modelos epidemiológicos	[Bailey, 1975] [Maguire et al., 1999]	[Barrett et al., 2005] [Situngkir, 2005] [Depinay et al., 2004]
Modelos epidemiológicos relacionados com a transmissão da dengue	[Newton and Reiter, 1992] [D. A. Focks and Keesling, 1995] [Leonel and Yoneyama, 2000] [Yang, 2003]	[Pessoa et al., 2004]
Modelos epidemiológicos relacionados com a dinâmica de populações de mosquitos <i>Aedes aegypti</i>	[D. A. Focks and Mount, 1993a] [D. A. Focks and Mount, 1993b] [Yang et al., 2002] [Ferreira and Yang, 2003] [Derouich et al., 2003] [Takahashi et al., 2005] [Otero et al., 2006]	

Tabela 2.1: Trabalhos revisados: modelos matemáticos e baseados em indivíduos

2.4 O mosquito *Aedes aegypti* e a dengue

Nesta seção são apresentados alguns dados relacionados à dengue e sua transmissão. Detalhes da biologia do mosquito *Aedes aegypti* são descritos.

2.4.1 A dengue

As doenças infecto-contagiosas sempre foram preocupação para a humanidade. A *peste negra*, transmitida pelas pulgas do rato, causou mortalidade de aproximadamente 25% da população européia durante o século XIV [McNeil, 1976]. Doenças como *dengue*, *malária*, *febre amarela*, *AIDS*, *ebola* são flagelos dos tempos modernos.

Algumas doenças são mais comuns em determinadas regiões, seja por condições climáticas, socioeconômicas ou geográficas. A dengue é uma doença presente em mais de 100 países do mundo, principalmente nos países de clima tropical, como os situados no continente asiático, continente africano e nas Américas [World Health Organization, 2002].

A dengue é transmitida para o homem através do mosquito do gênero *Aedes* e a presença de mosquitos em uma região é motivo de preocupação para entidades de saúde, pois dependendo do número de insetos contaminados, uma epidemia de dengue pode ocorrer [World Health Organization, 2002].

A dengue pode ser encontrada em duas formas. A primeira forma, conhecida por *Dengue Clássica* (DC), caracteriza-se por febres, sem sintomas respiratórios, seguidas por fortes dores de cabeça, náuseas e vômitos. A segunda forma, conhecida por *Dengue Hemorrágica*, caracteriza-se pelos mesmos sintomas da DC e por hemorragia interna. A dengue hemorrágica é fatal quando não tratada devidamente [World Health Organization, 2002].

Um dos maiores problemas relacionados à dengue é o fato da doença poder ser causada por quatro diferentes sorotipos, conhecidos por *DEN1*, *DEN2*, *DEN3* e *DEN4*. Uma pessoa contaminada por um sorotipo da dengue torna-se imune a este sorotipo em contaminações posteriores. Entretanto, se uma pessoa que foi contaminada por um sorotipo da dengue for novamente infectada por outro sorotipo, então esta pessoa torna-se susceptível ao desenvolvimento da dengue na forma hemorrágica [World Health Organization, 2002].

Atualmente, a dengue é endêmica em diversos países. Esse fato possibilita o aparecimento da doença na forma hemorrágica, aumentando o número de mortes na população. Essa doença é motivo de preocupação, não somente pelo aumento do número de óbitos, como também pelo aumento dos gastos públicos com o tratamento e prevenção da doença. Os governos dos países com maior incidência de casos de dengue seguem uma tendência de maiores investimentos em pesquisas e em combate a dengue. Um exemplo é o Governo do Brasil que promove anualmente o “Dia D”, o dia nacional de combate à dengue

[Ministério da Saúde, 2005].

De acordo com a Organização Mundial de Saúde (OMS) [World Health Organization, 2002], em 1995 o número de pessoas que se tornaram expostas ao risco de contaminação chegava a 2,5 bilhões. A cada ano estima-se que 50 milhões de novos casos de dengue ocorrem no mundo. Em 2001, mais de 609 mil casos de dengue ocorreram nas Américas. Isto é o dobro dos casos registrados em 1995. Em 2001, foram registrados cerca de 390 mil casos no Brasil, incluindo mais de 670 casos de dengue hemorrágica.

Somente no primeiro semestre de 2007, o Ministério da Saúde do Brasil contabilizou 438,9 mil casos de dengue em todo território brasileiro [Ministério da Saúde, 2007]. Isto corresponde a um aumento de 45,12 % em relação ao mesmo período do ano de 2006. O número de mortes por dengue hemorrágica aumentou de 77 para 98 óbitos nesse período. Como se percebe, estes números são crescentes.

O fenômeno do aquecimento global também é um fator preocupante em relação à dengue. De acordo com o relatório do Painel Intergovernamental de Mudanças Climáticas (IPCC, em inglês) da Organização das Nações Unidas (ONU), a temperatura média da superfície da Terra pode aumentar até 4 °C nos próximos 80 anos [IPCC, 2007]. Estudos indicam que com esta possível mudança o número de pessoas em situação de risco de contágio da dengue pode aumentar de 2,5 para 5 bilhões de pessoas [Houghton et al., 1997, Peterson, 1998].

2.4.2 A biologia do mosquito *Aedes aegypti*

Segundo Eiras, os mosquitos estão no planeta Terra há cerca de 30 a 54 milhões de anos [Eiras, 2000]. O mosquito *Aedes aegypti* é o principal responsável pela transmissão da dengue para seres humanos. Este inseto passa por algumas fases no seu processo de desenvolvimento biológico até atingir a idade adulta.

O ciclo de vida do mosquito *Aedes aegypti* é composto por quatro fases: ovo, larva, pupa e adulto, com durações variáveis. A fase ovo dura de dois a quatro dias em temperatura média de 26 °C, a fase larva dura de 10 a 20 dias e a fase pupa de um a três dias. Os mosquitos adultos vivem cerca de um a dois meses no verão e até seis meses no inverno [Eiras, 2000].

Um dos principais fatores que afeta o desenvolvimento dos mosquitos é a água [Consoli and Oliveira, 1994]. Quando o mosquito atinge a idade adulta voa até um abrigo, como troncos de árvores, buracos, postes, etc. [Eiras, 2000]. A partir do momento em que o mosquito está desenvolvido e não se encontra mais em seu criadouro, ele toma decisões visando sua sobrevivência e a manutenção da espécie.

Com frequência, a primeira atividade dos mosquitos na fase adulta é procurar por um parceiro do sexo oposto para reprodução. O macho, atraído pelas batidas de asas da fêmea, procura a parceira para acasalar [Eiras, 2000]. No entanto, raras vezes esse acasalamento ocorre depois do primeiro repasto sangüíneo da fêmea. Após seu próprio nascimento é comum o que o macho fique no criadouro aguardando a eclosão da pupa fêmea para acasalar [Consoli and Oliveira, 1994].

Uma diferença entre machos e fêmeas da espécie *Aedes aegypti* está na alimentação. Enquanto os mosquitos machos alimentam-se apenas de açúcares e néctares de plantas, as fêmeas são capazes de picar animais e alimentar-se de sangue. O mosquito adquire o vírus através do sangue contaminado de vítimas ou já nasce contaminado (contaminação vertical) [Gomes, 1998, Consoli and Oliveira, 1994].

Após a fêmea ser fecundada e se alimentar de sangue, ela procura um lugar para a desova. Normalmente a fêmea utiliza o mesmo criadouro em que nasceu para desovar, porém, outros fatores influenciam na escolha do criadouro, como intensidade luminosa, temperatura, grau de salinidade, presença de vegetais e micro-organismos. Em média, cada fêmea coloca de 10 a 30 ovos por criadouro. Estes ovos podem ser contaminados pelo vírus da dengue, se a sua genitora o possuir [Eiras, 2000]. Gomes et al. apresentam um modelo de distribuição temporal de ovos de fêmeas criadas em laboratório [Gomes et al., 2006].

De acordo com Eiras os mosquitos picam suas vítimas (pessoas e animais de sangue quente) dentro e/ou fora das casas, principalmente nos horários entre 7 e 10 horas e entre 16 e 19 horas [Eiras, 2000].

Toda a complexidade biológica do mosquito *Aedes aegypti*, aliada a sua dinâmica de vida, proporciona uma grande adaptabilidade deste inseto aos ambientes em que vive, e, como consequência, o controle de infestação é difícil [Consoli and Oliveira, 1994]. O crescimento dos transportes rodoviários e aéreos de pessoas e cargas e a disponibilidade de criadouros artificiais também proporciona o aumento do grau de distribuição de ovos e mosquitos *Aedes aegypti* no mundo, principalmente em regiões favoráveis para o desenvolvimento do inseto (umidade e temperatura) [Eiras, 2000]. Outros detalhes sobre a biologia do mosquito são discutidos no próximo capítulo: Modelo conceitual.

Capítulo 3

Modelo Conceitual

O modelo de simulação *SimPopMosq* é o núcleo do simulador de população de mosquitos *Aedes aegypti*. Neste capítulo são descritos os elementos do modelo de simulação *SimPopMosq*: agentes, objetos e leis naturais que são utilizados na configuração de diferentes cenários de simulações computacionais.

A descrição é baseada no protocolo *ODD* (*Overview, Design concepts and Details*) proposto em [Grimm et al., 2006]. Este protocolo foi proposto para descrever modelos baseados em indivíduos (MBA/MBI). O *protocolo ODD* é composto por três blocos principais - *Visão geral*, *Conceitos de projeto* e *Detalhes*. A figura 3.1 ilustra a composição do protocolo *ODD*.

Visão geral	Propósito
	Variáveis de estado e escala
	Visão de processo e escalonamento
Conceitos de projeto	Conceitos de projeto
Detalhes	Inicialização
	Entrada
	Sub-modelos

Figura 3.1: Estrutura do protocolo *ODD* [Grimm et al., 2006]

Grimm et al. descrevem vários exemplos de modelos de simulação usando o protocolo *ODD*. Os modelos apresentados incluem um cenário pré-estabelecido [Grimm et al., 2006]. O simulador proposto neste trabalho é capaz de tratar diferentes cenários a simulador. O simulador pode simular uma casa, um bairro. As limitações do simulador são relacionadas aos recursos computacionais disponíveis para as simulações.

A seguir a descrição do modelo de simulação *SimPopMosq* é apresentada empregando-se o protocolo *ODD*.

3.1 Visão geral

A visão geral é composta por três elementos - (i) propósito, (ii) variáveis de estado e escala e (iii) visão de processo e escalonamento. O objetivo é reportar uma visão geral do modelo de simulação proposto em termos conceituais.

3.1.1 Propósito

Ainda não é conhecida uma forma economicamente viável de se medir diretamente a população de mosquitos em uma dada região. Entretanto, a escolha de uma política eficiente de controle de população de mosquitos depende dessa informação.

Diante deste problema foi proposto um modelo de simulação de populações de mosquitos *Aedes aegypti*. O propósito do modelo computacional desenvolvido é analisar a dinâmica populacional de mosquitos *Aedes aegypti* e possui como meta estimar o tamanho da população de mosquitos. Esta análise é realizada considerando a população de mosquitos nas fases ovo e adulta, a disponibilidade de criadouros (reservatórios de água) e de alimentos em ambientes que podem ter áreas de dezenas a centenas de metros quadrados.

O alimento do mosquito é de origem animal ou vegetal. As pessoas e mamíferos presentes no ambiente servem como alimento, na medida em que são picadas pelos mosquitos fêmeas. O sangue ingerido é utilizado para maturação dos ovos da fêmea. A presença de plantas no ambiente serve como alimento para o mosquito adulto (seiva) ou como local de descanso noturno destes insetos.

3.1.2 Variáveis de estado e escalas

As variáveis de estado e escala são as características que determinam o estado em que os elementos se encontram e a escala de representação espaço-temporal dos cenários a serem simulados. A figura 3.2 ilustra o conjunto de elementos, divididos em camadas, que compõem o modelo de simulação *SimPopMosq*.

Observando de forma ascendente os níveis da figura 3.2, no nível mais básico desta hierarquia encontram-se a definição de algumas propriedades básicas do sistema como: escala de tempo, definição do espaço, definição dos deslocamentos e das velocidades. No nível intermediário, a camada “ambiente” possui elementos que podem ser inseridos nos cenários de simulação como: rastro de difusão, variáveis meteorológicas (chuva, radiação

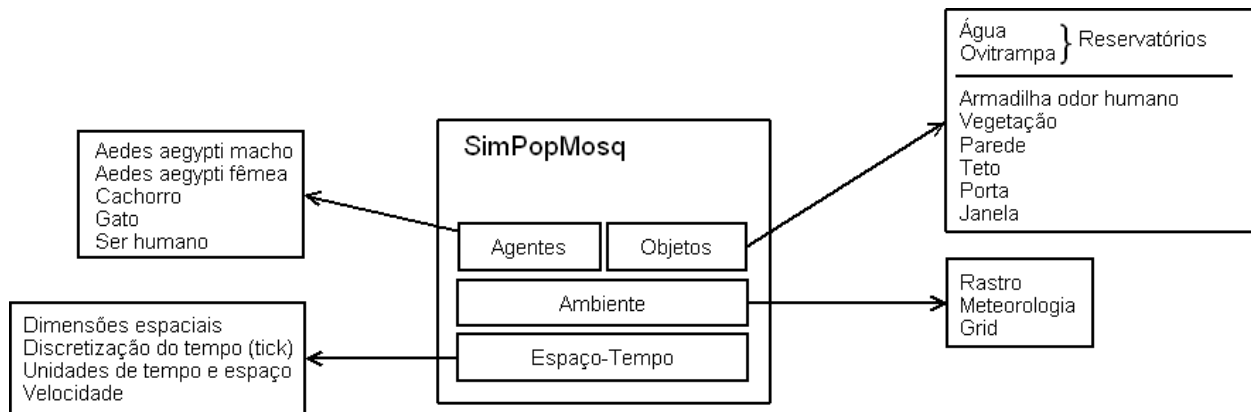


Figura 3.2: Elementos que compõem o modelo de simulação *SimPopMosq*

solar, vento, etc.) e o espaço georeferenciado para inserção de outros elementos como agentes e objetos. No nível superior encontram-se os objetos e agentes que podem ser utilizados nas simulações computacionais. Estes últimos elementos dependem das camadas inferiores para o seu funcionamento. A seguir, são descritas as camadas apresentadas na figura 3.2 com seus respectivos elementos e propriedades.

Espaço-Tempo

Os modelos computacionais dos agentes e objetos que fazem parte da ferramenta de simulação proposta possuem uma referência espaço-temporal. A referência espacial indica onde estes elementos estarão localizados em determinado instante no cenário configurado para a simulação (mundo). Uma localização é compreendida pela combinação dos três pontos cartesianos x , y e z (comprimento, largura e altitude). Os dois primeiros (comprimento e largura) são quantificados *discretamente* enquanto a dimensão z é *contínua*.

Os motivos pela adoção de variáveis contínuas e discretas são relacionados com questões de implementação e desempenho computacional que serão descritas no capítulo 4. Foi convencionalizada a relação de 1 (uma) unidade de espaço discreta para cada 50 centímetros do cenário simulado. Na figura 3.3 esta relação pode ser visualizada.

O tempo é modelado por unidades discretas de execução denominadas *ticks*. O *tick* é uma unidade de tempo artificial adotada pelo *framework Repast* e representa o intervalo de tempo discreto entre reavaliações sucessivas das variáveis do modelo. No simulador desenvolvido convencionou-se que 1 *tick* corresponde a 1 segundo simulado. Para ser simulado 1 segundo no modelo podem ser gastos frações de segundo ou vários segundos de processamento computacional. A noção de tempo se amplia para definições mais complexas, como a existência de datas e horas. As simulações são inicializadas em um instante, representado por uma data e hora, e finalizadas em outro instante. Períodos

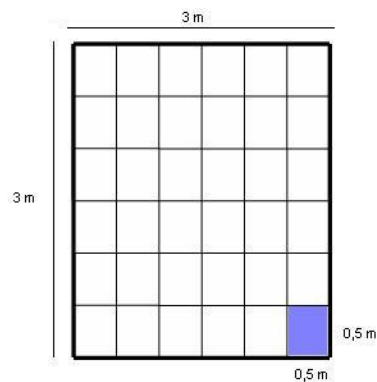


Figura 3.3: Exemplo de espaço discreto formado pelos eixos x e y

como dia e noite também podem ser interpretados a partir do horário da simulação.

A propriedade velocidade dos elementos que são capazes de se locomover é determinada pelo número de unidades de espaço discretas que este elemento pode percorrer durante uma unidade de tempo - *tick*.

O padrão adotado para as unidades de medida de tempo, espaço e velocidade são respectivamente segundos(s), metros(m) e metros por segundo (m/s).

Ambiente

A camada ambiente da ferramenta computacional desenvolvida é composta pelos elementos que podem ser incorporados nas simulações e interagir com os agentes e objetos das camadas superiores.

Rastro de difusão

Os rastros são deixados por agentes e objetos como: *Aedes aegypti*, *mamífero*, *armadilha de odor humano*, *água* e *vegetação*. Esses rastros são utilizados para modelar o campo visual, o campo sonoro ou ainda campos de dispersão de substâncias químicas.

Os elementos (agentes e objetos) que emitem rastro possuem uma propriedade que indica a intensidade do rastro liberado. Cada elemento possui parâmetros que indicam o mínimo e o máximo de rastro a ser emitido. A intensidade do rastro a ser emitida pelo elemento é definida a partir da geração de número aleatório definido entre estes parâmetros. De forma geral, as intensidades de rastro emitidas variam de 500 a 10000 unidades de rastro.

O modelo de difusão do rastro é descrito na seção 3.3.3. A figura 3.4 ilustra o comportamento do rastro de odor de um ser humano em função do seu deslocamento.

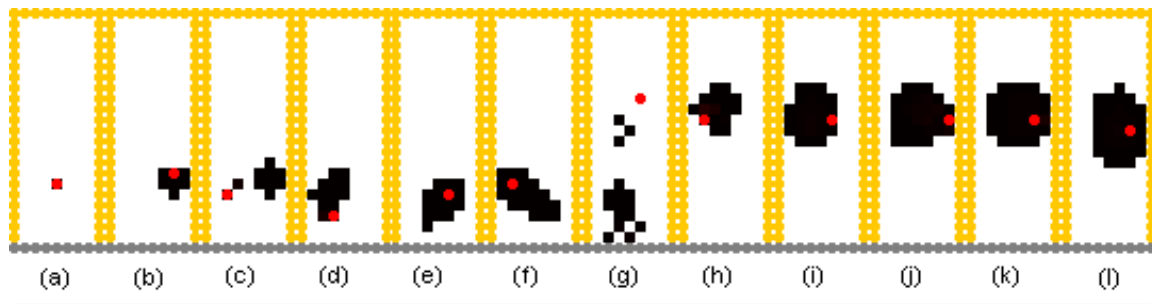


Figura 3.4: Movimento de um ser humano e seu rastro de odor

Clima - Variáveis meteorológicas

Algumas variáveis globais definem a situação climática do ambiente simulado. A influência que estas variáveis exercem é restrita aos processos de evaporação de água e precipitação. As variáveis climáticas consideradas são:

1. *Direção do vento ($^{\circ}$ NV)¹*
2. *Direção do vento na velocidade máxima ($^{\circ}$ NV)*
3. *Precipitação acumulada (mm)*
4. *Radiação solar acumulada (MJ/m^2)*
5. *Radiação solar global (MJ/m^2)*
6. *Temperatura do ar ($^{\circ}C$)*
7. *Temperatura máxima do ar nas últimas 24 horas ($^{\circ}C$)*
8. *Temperatura mínima do ar nas últimas 24 horas ($^{\circ}C$)*
9. *Umidade relativa do ar (%)*
10. *Velocidade do vento (m/s)*
11. *Velocidade máxima do vento (Rajada) (m/s)*

Detalhes sobre estas variáveis, como intervalo de medição e coleta de dados, estão descritos no apêndice A.

¹Norte verdadeiro

Agentes

Todos os agentes presentes no modelo *SimPopMosq* possuem em comum uma variável de localização. Esta variável indica a posição do agente (x , y e z) no cenário simulado.

Agentes *Aedes aegypti*

Os principais agentes presentes na ferramenta de simulação são os agentes computacionais que representam os mosquitos *Aedes aegypti*. O propósito do sistema é relacionado com estes agentes. Os mosquitos machos e fêmeas possuem variáveis em comum. As principais variáveis que determinam o estado do mosquito *Aedes aegypti* são:

1. *Fase* - As fases do agente *Aedes aegypti* determinam o seu comportamento. O agente pode vivenciar as fases de ovo, larva, pupa e adulto.
2. *Nível de energia* - No modelo do agente mosquito o nível de energia influencia nas suas decisões. Esta é uma propriedade variável e pode assumir valores entre 0 e 155520 unidades de energia. O motivo da escolha deste valor máximo é descrito na seção 5.2.3 - *Calibração de parâmetros*.
3. *Intensidade do Rastro* - Todo mosquito possui uma intensidade de rastro, que pode ser percebida por outros agentes. Esta é uma propriedade variável e pode assumir valores aleatórios entre 3000 e 10000 unidades de rastro.
4. *Comportamento/estado em execução* - Os mosquitos possuem diferentes comportamentos. Em cada momento pode assumir um comportamento, como por exemplo: procurar alimento, reproduzir, repousar, alimentar, etc.

Os mosquitos possuem alguns parâmetros que podem influenciar no seu comportamento e nas suas variáveis que definem seu estado. Estes parâmetros são:

1. *Duração das fases* - Cada fase tem uma média de duração, assim cada agente pode viver por um período variável. A duração de cada fase do agente é calculada com base em equações, parâmetros e números aleatórios.

(a) *Duração da fase Ovo*:

$$fase_o = literatura_o + (literatura_o + DistNormalAleatoria(\mu, \sigma))$$

O parâmetro $literatura_o$ corresponde ao valor médio da duração da fase ovo encontrado nas principais literaturas revisadas. A duração da fase ovo é calculada

somando-se $literatura_o$ a um número aleatório obtido a partir da distribuição normal configurada pela média μ e desvio padrão σ . Os valores adotados para $literatura_o$, μ e σ foram respectivamente: 3, 0 e 10 dias.

(b) *Duração da fase Larva:*

$$fase_l = literatura_l + (literatura_l \times DistUniformeAleatoria(min, max))$$

O parâmetro $literatura_l$ corresponde ao valor médio da duração da fase larva encontrado na literatura revisada. min e max são valores limites (inferior e superior) para geração do número com base na distribuição uniforme. Os valores adotados para $literatura_l$, min e max foram respectivamente: 15 dias, -0.2 e 0.2 dias.

(c) *Duração da fase Pupa:*

$$fase_p = literatura_p + (literatura_p \times DistUniformeAleatoria(min, max))$$

A fase pupa é calculada de forma idêntica à fase larva. Os valores adotados para $literatura_p$, min e max foram respectivamente: 3 dias, -0.1 e 0.1 dias.

(d) *Duração da fase Adulto:*

$$fase_a = DistNormalAleatoria(\mu, \sigma)$$

Caso $DistNormalAleatoria(\mu, \sigma)$ seja menor que zero, então utiliza-se a seguinte equação:

$$fase_a = literatura_a - DistNormalAleatoria(\mu, \sigma)$$

O parâmetro $literatura_a$ corresponde ao valor médio da duração da fase adulta encontrado na literatura. A duração da fase é calculada considerando-se um número aleatório obtido a partir da distribuição normal configurada pela média μ e desvio padrão σ . Os valores adotados para $literatura_a$, μ e σ foram respectivamente: 15, 15 e 15 dias.

2. *Risco de morte durante a mudança de fase* - Um mosquito pode morrer durante a transição de uma fase para outra. No modelo desenvolvido são adotadas, de forma *ad hoc*, probabilidades de sobrevivência em cada mudança de fase. Os parâmetros adotados são:

- (a) *Risco de morte na transição da fase Ovo para Larva: 30%*
- (b) *Risco de morte na transição da fase Larva para Pupa: 30%*
- (c) *Risco de morte na transição da fase Pupa para Adulto: 30%*
- (d) *Risco de morte após o atingir o fim da idade Adulta: 90%*

3. *Consumo de energia* - O consumo de energia do agente é caracterizado pela quantidade de energia necessária para que ele execute uma tarefa. Cada tipo de atividade possui como parâmetro um valor de consumo de energia. Em média o valor adotado para este parâmetro foi 0,75 unidades de energia gasta por segundo.
4. *Altura padrão de vôo* - Embora o mosquito possa voar livremente na dimensão z (altitude), assume-se como parâmetro o valor de 1 metro para a altitude padrão de seu vôo.
5. *Percepção da proximidade de vegetação* - Para um agente, perceber se há vegetação próxima é um fator importante. O agente necessita de alimentação para sobreviver e de locais para descanso. Foram definidos os parâmetros intensidade do rastro de vegetação *alta* e *média* para o mosquito verificar se a intensidade do rastro no ponto em que está é alta ou média ou baixa. Diante disto, o mosquito escolhe qual comportamento deverá assumir. Os valores adotados para intensidade de rastro de vegetação *alta* e *média* foram respectivamente 8000 e 6000 unidades de rastro. A figura 3.5 ilustra as intensidades dos níveis *alto* e *médio* do rastro de vegetação para o mosquito *Aedes aegypti*.

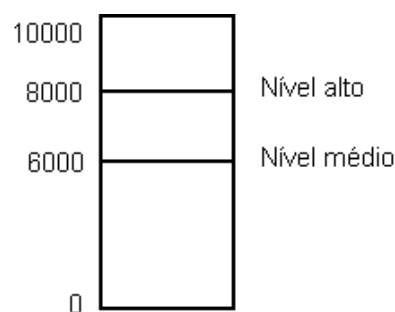


Figura 3.5: Níveis de proximidade de vegetação para o mosquito *Aedes aegypti*

6. *Duração da alimentação de seiva* - O tempo gasto pelo mosquito para se alimentar de seiva é considerado. Adotou-se como parâmetro o valor de 1200 segundos - 20 minutos - para esta alimentação.

7. *Velocidade de deslocamento* - Outra propriedade considerada é a velocidade de deslocamento. A velocidade afeta o consumo de energia do mosquito. Cada tipo de atividade possui um parâmetro relacionado que indica a velocidade que o mosquito desloca executando esta atividade. O valor médio adotado para este parâmetro foi 0,6 m/s.
8. *Velocidade de deslocamento na vertical* - Os agentes *Aedes aegypti* possuem a propriedade velocidade de deslocamento na vertical que é definida por um parâmetro que vale 0,8 m/s. Este valor foi determinado pela valor médio entre a velocidade de subida e descida de um mosquito.

Agente *Aedes aegypti* macho

O modelo computacional do mosquito do sexo masculino possui parâmetros que auxiliam na percepção do nível intensidade do rastro das fêmeas. Estes parâmetros são intensidade de rastro *alta* e *média* e o seu objetivo é similar ao objetivo dos parâmetros apresentados na percepção de vegetação: o mosquito escolhe qual comportamento deverá assumir diante da proximidade da fêmea. Os valores adotados para intensidade de rastro de fêmea *alta* e *média* foram respectivamente 8000 e 6000 unidades de rastro.

Agente *Aedes aegypti* fêmea

Os mosquitos *Aedes aegypti* de sexo feminino possuem diversas diferenças em relação aos machos relacionadas com a reprodução da espécie. Os comportamentos como alimentação sangüínea, fecundação e oviposição estão presentes apenas nas fêmeas. As variáveis que complementam o modelo computacional do mosquito *Aedes aegypti* fêmea e definem seu estado são:

1. *Nível de energia em função do volume de sangue consumido* - A fêmea necessita de uma energia “específica” para maturar seus ovos. Esta energia é oriunda do consumo de sangue animal. Quanto mais sangue ingerido maior a probabilidade dos ovos eclodirem. Esta é uma propriedade variável e pode assumir valores entre 0 e 155520 unidades de energia.
2. *Número de ovos no ovário* - O número de ovos a serem colocados varia para cada gestação. Esta variável pode assumir valores aleatórios entre 80 e 120 ovos. Entretanto, o número de ovos que irão eclodir dependerá proporcionalmente da variável *Nível de energia em função do volume de sangue consumido*.

3. *Duração da reprodução* - A duração da reprodução (coito) é considerada. Esta é uma propriedade variável e pode assumir valores entre 300 segundos (5 minutos) e 600 segundos (10 minutos).

As fêmeas possuem alguns parâmetros que podem influenciar no seu comportamento e nas variáveis que definem seu estado. Estes parâmetros são:

1. *Sucesso de Picadas* - O índice de sucesso de picadas é utilizado para determinar o sucesso da caçada de um mosquito. O sucesso de picada é determinado pela razão entre o número de picadas bem sucedidas pelo número total de tentativas de picadas. Foi considerado o valor de 80% de sucesso de picada para cada investida dos mosquitos nas suas vítimas.
2. *Risco de morte na tentativa de picada* - Probabilidade de um mosquito ser morto ao tentar picar uma vítima. Este risco foi definido como 2%.
3. *Duração da alimentação animal* - O mosquito permanece tentando atacar uma vítima até 900 segundos - 15 minutos. Este parâmetro serve para limitar o tempo em que o mosquito ficará em contato com os mamíferos.
4. *Quantidade máxima de sangue ingerido por picada* - Um mosquito ingere de uma vítima até 90% do volume máximo de sangue possível de ser ingerido.
5. *Nível de Percepção da proximidade de mamíferos* - As fêmeas são capazes de perceber a proximidade de um mamífero. Esta percepção influencia a decisão de realizar um ataque. Foram definidos os parâmetros intensidade do rastro de mamífero *alta* e *média* para o mosquito verificar se a intensidade do rastro no ponto em que está é alta ou média ou baixa. Diante disto, o mosquito escolhe qual comportamento deverá assumir. Os valores adotados para intensidade de rastro de mamífero *alta* e *média* foram respectivamente 8000 e 6000 unidades de rastros.
6. *Velocidade de oviposição* - A velocidade de oviposição indica quantos ovos serão colocados por unidade de tempo. Adotou-se o valor de 0,02 ovos por segundo.
7. *Padrão temporal de distribuição de ovos* - A distribuição de ovos ao longo do tempo é um fator importante para a sobrevivência da espécie do mosquito *Aedes aegypti*. No padrão de distribuição de ovos adotado, uma fêmea tende a colocar seus ovos aleatoriamente no ambiente e em períodos e quantidades diferentes. A distribuição adotada foi baseada no modelo proposto em [Gomes et al., 2006]. Esta distribuição é descrita no apêndice B - *Distribuição de Ovos do Mosquito Aedes aegypti*.

Agentes Mamíferos: Cachorro, gato e ser humano

Os outros agentes que existem no modelo *SimPopMosq* são os mamíferos: cachorro, gato e ser humano. Apesar de serem complexos, no modelo *SimPopMosq* a representação de seus comportamentos foi simplificada. Foram considerados apenas fatores que podem influenciar no estudo da dinâmica populacional do mosquito *Aedes aegypti*.

Estes três tipos de mamíferos possuem em comum quatro variáveis: comportamento em execução (que pode influenciar na velocidade do agente), intensidade de rastros (que pode influenciar na atração das fêmeas *Aedes aegypti*), altitude e tamanho.

O agente ser humano possui uma agenda de tarefas que podem ser executadas. Estas tarefas podem ser: ir para um determinado local, permanecer parado e andar aleatoriamente. Todas essas tarefas podem ser definidas e agendadas na construção das simulações a serem executadas. Outra variável presente no ser humano é a permissão para abrir e fechar determinados acessos como portas e janelas.

Objetos

Um vez descritas as variáveis de estados e propriedades dos agentes computacionais o mesmo é feito para os objetos do modelo *SimPopMosq*. Os objetos possuem em comum localização espacial (x , y e z) e formas geométricas, que podem ser cubos ou cilindros.

Reservatórios

A água é representada por reservatórios que são capazes de armazená-la. O objeto ovitrampa (armadilha para captura de ovos) também é representado como um reservatório. Este objetos possuem as seguintes propriedades:

1. *Percentual de líquido* - Esta propriedade é variável e indica o percentual de líquido dentro do reservatório.
2. *Volatilidade do líquido* - Certos líquidos podem ser mais voláteis do que outros. Logo, pode-se definir diferentes percentuais de volatilidade.
3. *Percentual de exposição* - Determinados reservatórios podem ter seu conteúdo mais exposto ao meio ambiente do que outros.
4. *Rastro*

O que diferencia estes objetos - água e ovitrampa - é intensidade do rastro emitido. Além disso, os ovos depositados pelas fêmeas na ovitrampa são eliminados.

Armadilha Odor Humano

Um segundo tipo de armadilha, denominado “Armadilha Odor Humano”, é definido. Esta armadilha emite um rastro similar ao dos agentes mamíferos e os mosquitos adultos são atraídos pela armadilha. Ao entrar na armadilha estes mosquitos são capturados e eliminados.

Vegetação

O objeto vegetação representa vegetação nos cenários simulados. Cada objeto vegetação possui um rastro próprio capaz de atrair mosquitos *Aedes aegypti* que estão em busca de alimentação à base de seiva ou de um local para descansar.

Teto e parede

Outros objetos disponíveis são os objetos que representam paredes e tetos. Estes objetos são utilizados na definição do cenário geográfico a ser simulado (casas, muros e lotes). Eles possuem as características citadas anteriormente (localização e dimensões de tamanho). Estes objetos podem ser utilizados pelos mosquitos como um local de descanso.

Acessos

Acessos a espaços restritos podem ser representados nos cenários de simulação. Os acessos disponíveis para representação são portas e janelas. As variáveis de estado que estes objetos possuem são localização, dimensões e se estão abertos ou fechados.

Uma visão alternativa dos elementos descritos nesta seção pode ser obtida através da observação dos diagramas de classes da aplicação desenvolvida. Este diagrama é discutido na seção 4.

As variáveis de estado do modelo de simulação *SimPopMosq* foram descritas nesta seção. Alguns parâmetros menos relevantes, que influenciam as variáveis definidas, não foram apresentados. Encontram-se disponíveis no apêndice D - seção D.1 - todos os parâmetros definidos e adotados no modelo e simulação *SimPopMosq*.

3.1.3 Visão de processo e escalonamento

Nesta subseção uma visão geral dos processos existentes no modelo de simulação *SimPopMosq* é apresentada. Os processos podem ser compreendidos como tudo que é executado em determinado instante pelo mecanismo central de execução da ferramenta desenvolvida.

Visão geral	Propósito	✓
	Variáveis de estado e escala	✓
	Visão de processo e escalonamento	←
Conceitos de projeto	Conceitos de projeto	
Detalhes	Inicialização	
	Entrada	
	Sub-modelos	

Figura 3.6: Protocolo *ODD* - Visão de processo e escalonamento

Ambiente

Rastro

O rastro deixado pelos agentes e objetos é atualizado através de um processo de difusão que considera vários parâmetros definidos. Este processo de difusão incorpora um modelo matemático descrito na seção 3.3.3. A execução deste processo se dá em intervalos de tempos previamente definidos ou após a locomoção do agente.

As variáveis climáticas do modelo de simulação descritas anteriormente são atualizadas ordinariamente através de dados definidos na configuração da simulação proposta pelo usuário da ferramenta. Um processo de atualização automática busca informações climáticas em bases de dados e define os valores para estas variáveis, considerando a data e hora do cenário em simulação. Os detalhes deste processo estão descritos no apêndice A.

Agentes

Os processos de agentes são executados a partir da decisão tomada por cada agente (processo de tomada de decisão). Já os processos independentes de agentes são executados em intervalos de tempos previamente definidos nos parâmetros do modelo de simulação.

Agentes *Aedes aegypti*

Os agentes computacionais que representam o mosquito *Aedes aegypti* basicamente possuem os seguintes processos:

1. *Nascimento ou emergência* - Processo que inicializa o agente e suas características básicas.
2. *Metamorfose* - Processo responsável pela mudança de fase do agente. O ciclo de vida do agente pode ser visto na figura 3.7.

3. *Morte* - Este processo se encarrega de finalizar a execução dos processos do agente em execução, bem como desativá-lo do cenário de simulação.

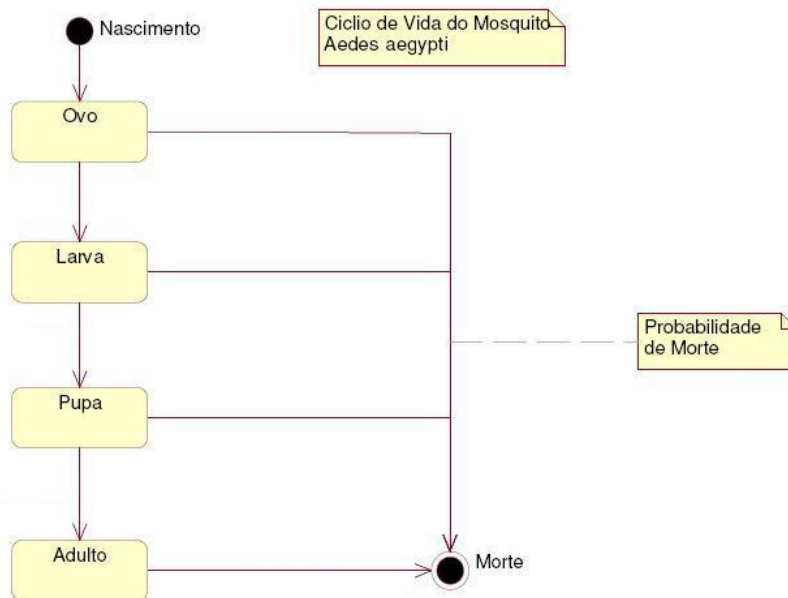


Figura 3.7: Ciclo de vida do agente mosquito *Aedes aegypti*

Cada agente é capaz de tomar decisões, através de um processo que avalia um conjunto de estados e regras de decisões estabelecidas. Este processo é executado em intervalos de tempo previamente definidos.

Agente *Aedes aegypti* macho

O agente que representa o mosquito *Aedes aegypti* do sexo masculino apresenta as seguintes ações que podem ser inicializadas a partir da decisão escolhida:

1. *Voar aleatoriamente* - Permite que o agente voe aleatoriamente, respeitando o espaço ocupado por outros elementos.
2. *Procurar sossego* - O agente pode voar a procura de algum local para o repouso. Este processo é executado quando o período diurno é finalizado, assim o agente localiza um local para repousar a noite.
3. *Procurar alimento seiva* - O mosquito macho se alimenta de seiva.
4. *Alimentar* - Após encontrar alimento o mosquito pode iniciar sua alimentação.
5. *Procurar Parceira* - Procura por uma parceira para reprodução.
6. *Reproduzir* - Encontrada a parceira, inicializa-se o ato de reprodução dos agentes.

Agente *Aedes aegypti* fêmea

A fêmea do mosquito *Aedes aegypti* apresenta os seguintes processos que podem ser executados a partir das suas decisões:

1. *Voar aleatoriamente* - Permite que o agente voe aleatoriamente respeitando o espaço ocupado por outros elementos.
2. *Procurar sossego* - O agente voa a procura de algum local para descanso. Normalmente, este processo é executado quando o período fotofásico é finalizado.
3. *Procurar alimento animal* - A fêmea necessita de alimentação sangüínea para auxiliar no processo de maturação dos ovos.
4. *Procurar alimento seiva* - Quando a fêmea não estiver fecundada ou necessitar de alimentação urgente este tipo de alimentação será preferido.
5. *Alimentar* - Após encontrar alimento o mosquito inicia sua alimentação.
6. *Reproduzir* - Se a fêmea é encontrada pelo macho inicializa-se o ato de reprodução dos agentes.
7. *Procurar criadouro para desova*. A fêmea *Aedes aegypti* é diferente do macho no aspecto da gestação. Esta diferença torna necessária a definição de um processo onde a fêmea é capaz de procurar um criadouro para depositar seus ovos maduros.
8. *Desovar*. Após encontrar um local para desova a fêmea inicia a desova.

A ordem e condição de execução destes processos estão definidas nos processos de tomada de decisão do agente. Os processos de tomada de decisão dos agentes *Aedes aegypti* macho e fêmea são sintetizados nos diagramas apresentados respectivamente nas figuras 3.13 e 3.14.

Agentes Mamíferos: Cachorro, gato e ser humano

Os agentes mamíferos possuem o processo *Deslocar aleatoriamente*. Este processo faz com que o agente se locomova em diferentes direções, sentidos e aleatórios, mas com velocidades típicas. Estes agentes se deslocam sem invadir espaços previamente ocupados.

Agente ser humano

O agente computacional que representa o ser humano tem a possibilidade de verificar sua agenda de tarefas, através do processo *Verificar agenda*. Um processo que pode ter sua execução agendada é o processo *Deslocar ao Destino* e que permite que o agente se desloque de um ponto a outro.

Objetos

Reservatórios

Os objetos que armazenam líquidos, como os objetos que representam a água e a ovitrampa possuem processos que realizam a atualização do nível de líquido do seu reservatório. Esta atualização ocorre através da execução de um modelo matemático que considera condições climáticas do cenário simulado, como umidade relativa do ar e nível de precipitação. Este modelo é detalhado na seção 3.3.3.

3.2 Conceitos de projeto

Grimm et al. definem um *checklist* para identificação de conceitos que podem auxiliar na fase de projeto dos modelos baseados em indivíduos, facilitando o desenvolvimento e integração [Grimm et al., 2006]. A seguir são descritos os conceitos de projeto do modelo de simulação proposto.

Visão geral	Propósito	✓
	Variáveis de estado e escala	✓
	Visão de processo e escalonamento	✓
Conceitos de projeto	Conceitos de projeto	←
Detalhes	Inicialização	
	Entrada	
	Sub-modelos	

Figura 3.8: Protocolo *ODD* - Conceitos de projeto

3.2.1 Fenômenos emergentes

A dinâmica do tamanho da população de mosquitos *Aedes aegypti* pode ser considerada uma propriedade emergente do modelo de simulação. A população, em todas as fases do mosquito é determinada a partir da interação entre agentes, objetos e o ambiente.

3.2.2 Adaptabilidade

Os agentes computacionais que representam os mosquitos *Aedes aegypti* - macho e fêmea - são os únicos elementos capazes de ter o comportamento adaptado dependendo de suas condições internas ou do ambiente.

As fêmeas *Aedes aegypti* podem adaptar seu comportamento em função das seguintes condições:

1. *Período fotofásico*² - Durante a noite o comportamento da fêmea é diferente do seu comportamento diurno. Normalmente a fêmea utiliza a noite para desovar. Caso a fêmea não esteja em fase de desova, então utiliza este período para o repouso.
2. *Nível de energia* - Durante o dia o comportamento da fêmea depende do seu nível de energia atual. A tendência é que se a fêmea estiver com nível energético baixo ou médio procure mais alimento para se nutrir e se estiver grávida nutrir seus ovos com alimentação sangüínea.

O agente computacional do mosquito *Aedes aegypti* macho, embora seja mais simples que a fêmea, apresenta os seguintes comportamentos adaptativos:

1. *Período fotofásico* - Durante a noite o comportamento do macho é diferente do seu comportamento diurno. Normalmente o macho utiliza a noite para o descanso.
2. *Nível de energia* - Durante o dia o comportamento do macho irá depender do seu nível de energia. Se o macho estiver com nível energético baixo ou médio procura alimento à base de plantas para se nutrir.
3. *Nível de proximidade de fêmeas* - Quando o macho percebe a presença de fêmea muda seu comportamento para o acasalamento, aproximando-se desta fêmea.

Os modelos de tomada de decisão dos mosquitos machos e fêmeas podem ser vistos respectivamente através das figuras 3.13 e 3.14.

3.2.3 *Fitness*

Grimm et al. definem *Fitness* como a conseqüência de um comportamento e *fitness-seeking* como uma fenômeno de procura por melhores escolhas de comportamentos, de forma a elevar a sobrevivência do agente [Grimm et al., 2006].

O *fitness-seeking* dos agentes computacionais modelados - *Aedes aegypti* - é implícito. Embora os agentes não calculem o quanto uma decisão pode ser mais eficiente do que outra, implicitamente escolhem os comportamentos mais coerentes com sua realidade (Ex. falta de água, de alimentação sangüínea, etc.).

Os agentes *Aedes aegypti* fêmea quando estão em fase de oviposição distribuem os ovos de forma aleatória e em quantidades e momentos diferentes. Esta estratégia de distribuição não uniforme tende a minimizar o efeito da possível eliminação de criadouros,

²O período fotofásico compreende o intervalo em que os raios solares são presente no ambiente. Foi considerado que este período inicializa-se as 5 horas e finaliza as 19 horas.

como o extermínio de ovos. A distribuição adotada foi baseada no modelo proposto em [Gomes et al., 2006]. Essa distribuição é descrita no apêndice B.

Os mosquitos *Aedes aegypti* apresentam estratégias que aproveitam oportunidades. Por exemplo, se o macho percebe a presença de uma fêmea, dá prioridade para o comportamento que o leva a tentar se acasalar. Em alguns casos, se a fêmea possui ovos no seu ovário e percebe um mamífero, tentará aproveitar esta oportunidade e picar o mamífero, independente do seu nível de energia já ser suficiente para nutrir os ovos.

3.2.4 Predição

Os agentes presentes no modelo de simulação *SimPopMosq* proposto não agem com predição, ou seja, não decidem pensando nas conseqüências futuras da escolha eleita. Além disto, esses agentes não possuem memória e não aprendem.

3.2.5 Sensibilidade

A principal forma dos agentes perceberem a presença de determinado agente ou objeto é através da percepção de intensidade de rastro emitido pelo elemento investigado. Os mosquitos machos, por exemplo, percebem as fêmea através da intensidade de rastro emitida por estas, que simula o som emitido pela batida de asas da fêmea.

3.2.6 Interação

Os objetos, agentes e ambiente possuem elementos que compõem o modelo de simulação proposto. Nas simulações configuradas todos os elementos interagem com o mundo simulado, uma vez que estão geo-referenciados neste mundo e ocupam um espaço físico. Estes elementos também interagem entre si e as principais interações são:

1. *Objetos e ambiente* - Os objetos localizados no mundo simulado podem interagir com a ambiente, na medida que liberam rastros de sua existência. Os objetos reservatórios de água deixam rastros de umidade no ambiente e as vegetações liberam rastros de seiva. Além do rastro, o objetos reservatórios de água são capazes de acumular água proveniente de chuvas.
2. *Agente e ambiente* - Assim como os objetos, os agentes podem interagir com o ambiente através dos rastros liberados.
3. *Agente e objetos* - Os agentes computacionais podem interagir com os objetos do modelo de simulação *SimPopMosq*. Esta interação ocorre através do contato físico

entre estes elementos (mesma posição geográfica). Os mosquitos *Aedes aegypti* podem interagir com a vegetação na medida em que pousam para alimentação ou descanso, com os reservatórios de água no processo de oviposição e desenvolvimento larval e com objetos físicos como paredes, tetos, portas e janelas. Estes últimos objetos podem servir como local de descanso para os mosquitos. Os mamíferos interagem com os objetos, na medida que estes servem de obstáculos. O agente ser humano pode, ainda, acessar determinadas portas quando estiverem abertas ou quando o agente possuir acesso (chave).

4. *Mosquito Aedes aegypti macho e fêmea* - Os mosquitos macho e fêmea podem interagir no momento de acasalamento. Quando o macho detecta o rastro da fêmea, então aproxima-se e tenta o acasalamento. Se a fêmea estiver disposta a acasalar, então durante um período os agentes permanecem juntos até a conclusão do cruzamento.
5. *Mosquito Aedes aegypti fêmea e mamíferos* - Como forma de garantir o desenvolvimento de seus ovos, a fêmea *Aedes aegypti* necessita de sangue. Direcionada pelo rastro do mamífero (que simula o rastro de odor emitido) a fêmea tenta picar o animal. Caso tenha sucesso, permanece por um período de tempo se alimentando.

3.2.7 Estocasticidade

O modelo de simulação proposto apresenta comportamento estocástico devido às diversas transições probabilísticas definidas e configuradas a partir de números pseudo-aleatórios gerados. Os principais pontos de aleatoriedade encontrados no modelo proposto são indicados a seguir:

1. *Intensidade dos rastros dos elementos* - Os elementos que emitem algum tipo de rastro os emitem em uma determinada intensidade. Esta intensidade é aleatoriamente escolhida entre uma faixa de valores estabelecida para cada tipo de elemento.
2. *Deslocamento aleatório* - Os agentes mosquitos *Aedes aegypti* e mamíferos podem se deslocar aleatoriamente. Os mosquitos podem ainda se deslocar no eixo z .
3. *Ciclo de vida dos mosquitos Aedes aegypti* - Nas transições das fases da vida do mosquito *Aedes aegypti* são associadas probabilidades que podem aumentar ou diminuir o período em que o mosquito permanece em determinada fase.
4. *Mosquito Aedes aegypti fêmea* - A fêmea do mosquito *Aedes aegypti* possui um grau de aleatoriedade na reprodução (determinar o número de ovos a serem nutridos), oviposição (determinar o sexo dos mosquitos que irão nascer) e alimentação

(determinar o sucesso de picada e a probabilidade de morte, no caso de insucesso da picada).

3.2.8 Coletividade

No modelo de simulação proposto o conceito de coletividade não é presente, porque cada indivíduo possui uma representação computacional única e não possui objetivos comuns com outros agentes no sentido coletivo. No mundo real, os mosquitos *Aedes aegypti* não formam enxames. Esta característica foi incorporada no modelo de simulação. O máximo que ocorre é dois agentes *Aedes aegypti* (macho e fêmea) se juntarem para acasalamento.

3.2.9 Observação

Uma das principais questões que o modelo desenvolvido se propõe a responder é: *Qual é o tamanho da população de mosquitos em determinado momento ?* O tamanho das populações de mosquitos nas suas diferentes fases são observados (registrados), assim como outras variáveis que contribuem para a análise da dinâmica da população do mosquito. As principais variáveis que podem ser observadas e registradas no modelo proposto são:

1. *Número de mosquitos Aedes aegypti na fase ovo.*
2. *Número de mosquitos Aedes aegypti na fase larva.*
3. *Número de mosquitos Aedes aegypti na fase pupa.*
4. *Número de mosquitos Aedes aegypti na fase adulta.*
5. *Número de mosquitos Aedes aegypti macho.*
6. *Número de mosquitos Aedes aegypti fêmea.*
7. *Número de mosquitos Aedes aegypti na fase ovo produzidos nas últimas 24 horas.*
8. *Número de mosquitos Aedes aegypti na fase adulto produzidos nas últimas 24 horas.*
9. *Taxa de Picadas*
10. *Índice de positividade de ovitrampa*
11. *Índice de densidade de ovitrampa*
12. *Índice de positividade de larvas e pupas em ovitrampa*

3.3 Detalhes

Grimm et al. discorrem sobre a importância de se descrever detalhes do modelo de simulação. Esta descrição é importante para que os experimentos possam ser reproduzidos [Grimm et al., 2006].

3.3.1 Inicialização

A figura 3.9 ilustra as etapas do modelo *ODD* descritas até este ponto.

Visão geral	Propósito	✓
	Variáveis de estado e escala	✓
	Visão de processo e escalonamento	✓
Conceitos de projeto	Conceitos de projeto	✓
Detalhes	Inicialização	←
	Entrada	
	Sub-modelos	

Figura 3.9: Protocolo *ODD* - Inicialização

A inicialização da simulação depende da especificação formal de alguns dados. A figura 3.10 apresenta uma síntese dos principais conjuntos de dados necessários para definição da simulação.

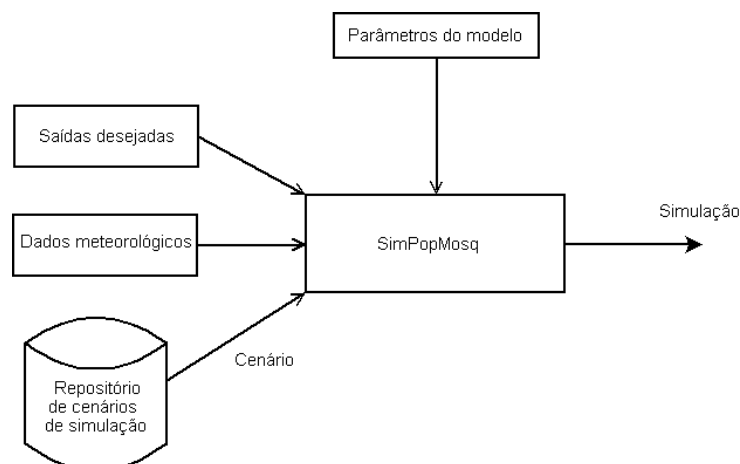


Figura 3.10: Inicialização da simulação

Os dados necessários para inicialização do modelo são *cenário de simulação*, *dados meteorológicos*, *saídas desejadas* e *parâmetros do modelo*. Uma breve descrição de cada dado é apresentada a seguir:

- *Cenário de simulação* - No simulador proposto é possível criar e simular diferentes cenários. Para isto, basta que o usuário da ferramenta adicione no repositório de cenários de simulação o cenário desejado. A especificação do cenário consiste na configuração dos agentes e objetos a serem adicionados no ambiente.
- *Dados meteorológicos* - Os cenários a serem simulados podem contar com variáveis climáticas, que influenciam o *mundo* a ser simulado. Estas variáveis são atualizadas através de dados importados de bases de dados climáticas e adaptadas a um padrão previamente definido. O padrão adotado é baseado nos dados disponibilizados pelo *Centro de Previsão de Tempo e Estudos Climáticos do Instituto Nacional de Pesquisas Espaciais - CPTEC/INPE*. Os detalhes sobre este padrão são descritos no apêndice A.
- *Saídas desejadas* - A especificação das saídas desejadas da simulação é opcional. Se um usuário deseja saber como determinada variável se comporta ao longo da simulação, poderá indicar em uma especificação formal (arquivo XML) a variável que deseja observar. O simulador irá registrar os valores dessa variável. Os valores dos parâmetros adotados foram obtidos ou estimados a partir dos dados bibliográficos consultados [Templin, 2000, Eiras, 2000, Consoli and Oliveira, 1994, Minar et al., 1996] [Focks et al., 1993b, Focks et al., 1995]. A etapa de atribuição e calibração dos parâmetros do modelo é crítica.
- *Parâmetros do modelo* - O modelo de simulação proposto possui um conjunto de *parâmetros* de controle (mais de 130 parâmetros) que determinam o funcionamento das simulações.

3.3.2 Entrada

Cenários

Os cenários de simulação podem ser definidos e armazenados em um repositório. Os cenários a serem simulados são definidos através de uma estrutura de dados (classe) da linguagem de programação *Java*, obedecendo a um conjunto de regras (interfaces e classes abstratas). As seguintes operações devem ser definidas para a construção do cenário de simulação:

1. *Definição do espaço* - As dimensões do mundo a ser simulado são informações obrigatórias para a existência das simulações. Define-se a largura e comprimento do

mundo virtual. A altura deste ambiente não é definida. A dimensão z assume valores contínuos e “ilimitados”.

2. *Definição do tempo* - Definir o tempo no cenário de simulação significa que a simulação terá uma data e hora virtual. As principais datas/hora existentes no mundo virtual simulado são: *início da simulação*, *fim da simulação* e *data/hora atual*. A *data/hora atual* é calculada considerando a data de *início da simulação* e o tempo decorrido a partir do início da simulação.
3. *Criação de objetos*. A configuração da instalação dos objetos no cenário a ser simulado é importante. Nesta configuração define-se a posição de cada objeto no mundo virtual e as propriedades particulares de cada objeto, como por exemplo o nível de água de um reservatório ou o tamanho de uma parede.
4. *Criação de agentes*. Os agentes que irão participar da simulação devem ser definidos. Estes agentes podem ser mosquitos *Aedes aegypti* ou mamíferos. No caso do agente *ser humano* a trajetória de deslocamento e sua dinâmica podem ser previamente definidas.

Dados meteorológicos

Os dados meteorológicos adotados nas simulações seguem o padrão dos dados do CP-TEC/INPE. Embora muitos dados estejam disponíveis no cenário de simulação, os atuais modelos de objetos e agentes desenvolvidos não os utilizam completamente. Os principais dados utilizados no modelos são: *temperatura do ar*, *precipitação acumulada* e *umidade relativa do ar*.

A descrição deste padrão, dados, unidades de medida e alguns detalhes técnicos pode ser visualizada no apêndice A.

Saídas desejadas

Na seção 3.2.9 estão listadas as principais variáveis atualmente incluídas no modelo de simulação, como saídas desejadas. Estas variáveis serão discutidas no capítulo 5 - *Experimentos computacionais* - sob a perspectiva de resultados obtidos a partir de simulações computacionais realizadas.

Parâmetros do modelo

Os principais parâmetros do modelo de simulação foram apresentados na seção 3.1.2. As faixas de valores que as variáveis podem assumir nas simulações estão definidas no

arquivo de parâmetros da simulação *ParametrosEntrada.XML*. Este arquivo encontra-se no apêndice D deste trabalho.

3.3.3 Sub-modelos

O modelo de simulação *SimPopMosq* é composto por diversos sub-modelos. A seguir são detalhados estes sub-modelos.

Visão geral	Propósito	✓
	Variáveis de estado e escala	✓
	Visão de processo e escalonamento	✓
Conceitos de projeto	Conceitos de projeto	✓
Detalhes	Inicialização	✓
	Entrada	✓
	Sub-modelos	←

Figura 3.11: Protocolo ODD - Sub-modelos

Rastro de difusão

O modelo computacional do *rastro de difusão* foi inspirado no modelo disponível no *framework* de simulação multiagentes *Swarm* [Swarm Development Group, 2004] citeSwarm.2. Este modelo é um *autômato celular* onde cada célula possui uma intensidade de rastro. A figura 3.12 ilustra esta idéia.

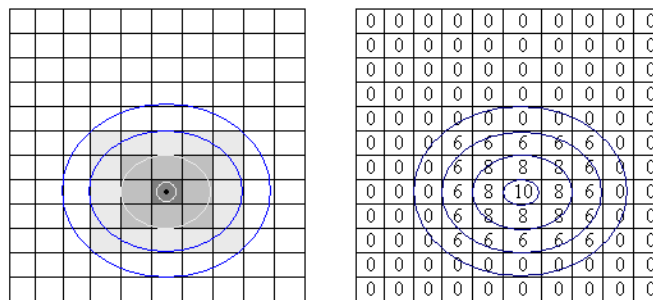


Figura 3.12: Rastro de difusão: Autômato celular

Ao final de cada iteração da simulação (1 *tick*), a propagação do rastro no cenário de simulação é realizada. Esta propagação se dá pela atualização do valor da intensidade do rastro em cada célula do *autômato celular*, através da seguinte regra de transição:

$$i_n = t_e \times (i_a + c_d \times (m_v - i_a))$$

Esta equação demonstra o cálculo do valor da *nova intensidade* (i_n) de rastro em uma célula n . O cálculo é realizado considerando a *intensidade atual* do rastro na célula (i_a), a *taxa de evaporação* (t_e) e *constante de difusão* (c_d) do rastro, além do valor médio de intensidade de rastro das oito células vizinhas (m_v).

Os desenvolvedores do *framework* de simulação multiagente *Swarm*[Minar et al., 1996] sugerem os valores 1 e 0,99 para as constantes c_d e t_e , respectivamente.

Variação de líquidos em reservatórios

Os reservatórios disponíveis no modelo de simulação proposto - água e ovitrampa - podem sofrer variação no nível do líquido que possuem. Esta variação ocorre em função de mudanças climáticas. As principais mudanças são: i) aumento do volume de líquido por causa da precipitação; ii) diminuição do volume de líquido por causa da evaporação.

O aumento do volume de líquido consiste no incremento do valor da variável *percentual de líquido* do objeto reservatório, de acordo com a variável climática *precipitação acumulada*.

A diminuição do volume de líquido dos reservatórios - evaporação - é um processo mais complexo do que a precipitação. Adotou-se o modelo de evaporação da água proposto em [Focks et al., 1993a], como segue:

$$v_e = 0,93 + (0.28 \times e_s) - (0.01 \times u_r)$$

Este modelo considera as variáveis *Percentual de exposição ao sol* (e_s), que varia entre 0 e 1, e umidade relativa do ar (u_r). O volume evaporado de água (v_e) é calculado em centímetros evaporados por dia.

Mosquito *Aedes aegypti* macho

Uma das principais contribuições deste trabalho está na definição dos modelos computacionais de tomada de decisão e comportamento dos mosquitos *Aedes aegypti*.

O modelo de tomada de decisão do mosquito *Aedes aegypti* macho pode ser visualizado na figura 3.13. Nesta figura, o modelo foi representado através de um *questionário*. O agente é autônomo e utiliza este questionário para controlar suas ações.

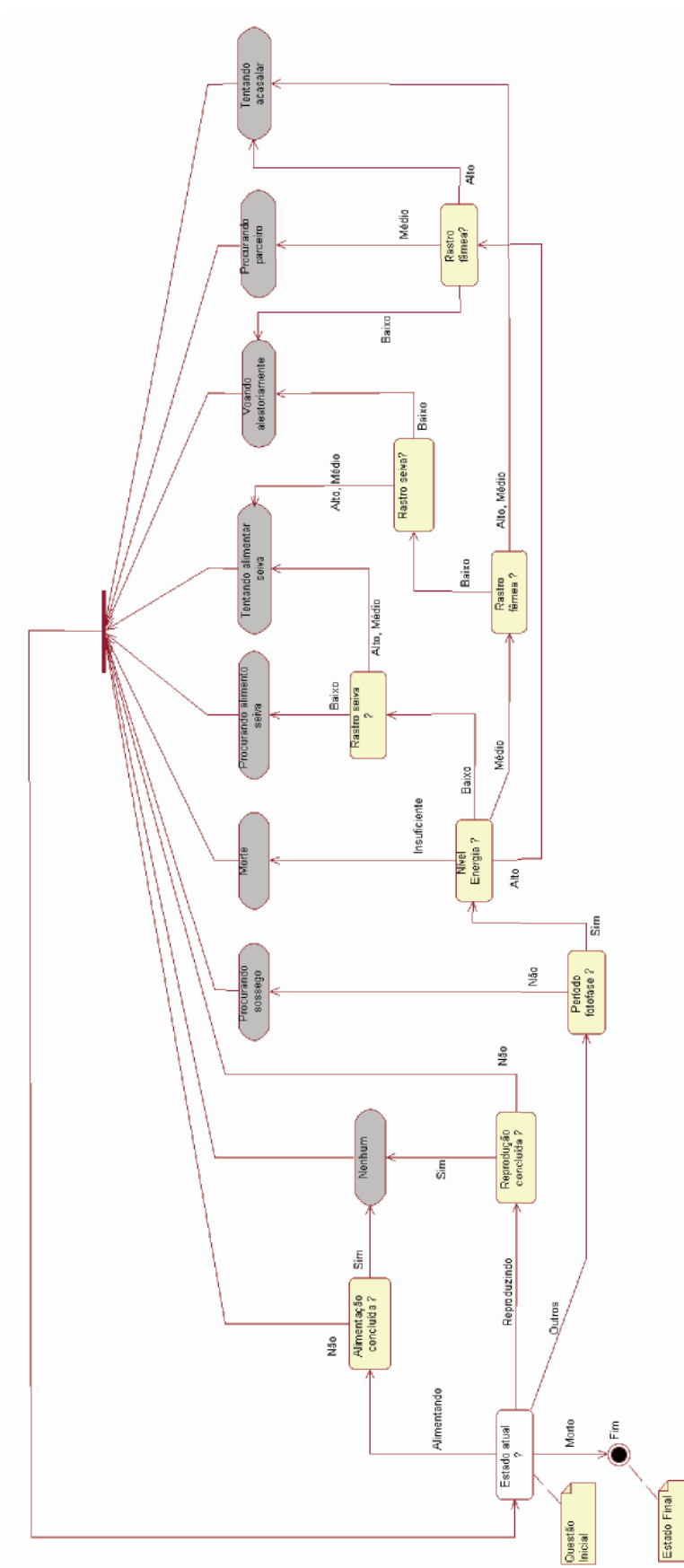


Figura 3.13: Questionário do modelo de comportamento do mosquito *Aedes aegypti* macho

No *questionário* da figura 3.13 observa-se a existência de um momento importante. Este momento denominado *Questão inicial* trata do principal ponto de decisão do modelo (caixa de fundo branco). Neste momento, uma pergunta sobre o comportamento em atividade do mosquito é realizada. De acordo com a resposta outros questionamentos são feitos (caixas de fundo amarelo claro).

Respondida a questão inicial e as demais questões um estado de comportamento é definido (caixa de fundo cinza). O mosquito *Aedes aegypti* executa as ações referentes ao estado escolhido. O processo de tomada de decisão é iterativo (reinício da questão inicial). Estas iterações são executadas em intervalos de tempo pré-definidos. No caso do agente *Aedes aegypti* macho este intervalo de tomada de decisão equivale a 10 segundos, ou 10 *ticks*.

A questão inicial sobre o comportamento em atividade possui quatro diferentes saídas. As duas primeiras são relacionadas às atividades em execução como: alimentação e reprodução. Caso o mosquito esteja no estado “Alimentação” ou “Reprodução”, e esta atividade não tenha sido finalizada, então este agente deve continuar a atividade. Por outro lado, caso a alimentação ou a reprodução tenha finalizado, então o mosquito fica livre para novas escolhas. A terceira possível resposta da questão inicial está relacionada ao fato do mosquito ter tido seu estado alterado para *Morto*. Neste caso o processo de tomada de decisão deste agente é finalizado.

Se a resposta à pergunta principal for um comportamento em atividade diferente dos estados em alimentação, reprodução ou morto, então outras questões devem ser respondidas para se chegar a escolha de um novo estado de comportamento a ser assumido pelo mosquito. As perguntas adotadas relacionam-se ao *período fotofásico* (dia ou noite), *nível de energia atual do mosquito*, *existência de seiva de plantas para alimentação* e *fêmeas para reprodução* nas proximidade.

De acordo com as respostas às perguntas os seguintes estados de comportamento podem ser alcançados: *Procurar sossego*, *Morrer*, *Procurar alimento à base de seiva*, *Tentar se alimentar de seiva*, *Voar aleatoriamente*, *Procurar parceiro* e *Tentar acasalar*.

A fim de ilustrar este modelo de tomada de decisão foram realizados alguns experimentos computacionais. A ilustração, descrição e execução destes experimentos são detalhados na seção 5.1 do capítulo 5 - *Experimentos computacionais*.

Mosquito *Aedes aegypti* fêmea

O agente computacional mosquito *Aedes aegypti* fêmea é mais complexo do que o agente macho. Isto acontece pelo fato da fêmea ter o papel de conduzir a gestação dos descendentes da espécie, que inclui alimentação sangüínea para maturação dos ovos e

procura de um local adequado para postura dos ovos.

Esta complexidade superior do comportamento dos agentes *Aedes aegypti* fêmeas em relação aos machos pode ser visualizada através da figura 3.14, que ilustra o questionário que define o comportamento da fêmea. Observa-se que o número de perguntas a serem respondidas na decisão da fêmea é maior do que o número de perguntas encontradas na tomada de decisão do macho, ilustrada na figura 3.13.

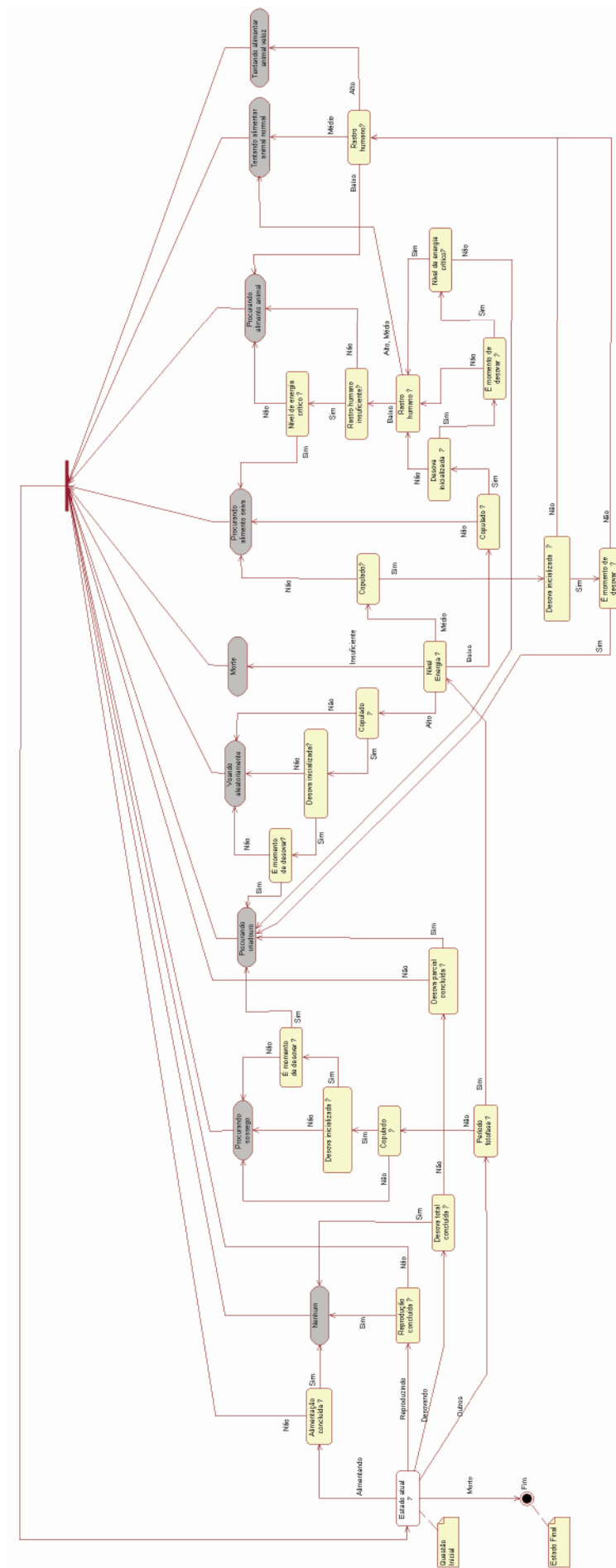


Figura 3.14: Questionário do modelo de comportamento do mosquito *Aedes aegypti* fêmea

O modelo da fêmea possui uma pergunta central, relacionada ao comportamento em atividade do mosquito. Esta pergunta possui cinco respostas relacionadas aos seguintes comportamentos: alimentação, reprodução, morte, oviposição e demais estados. Neste ponto, a novidade em relação ao mosquito *Aedes aegypti* macho é a resposta vinculada ao comportamento de oviposição. Caso a fêmea esteja colocando seus ovos e o processo de oviposição não tenha completado, então o mosquito deve continuar colocando os ovos. Caso a oviposição termine parcialmente então o mosquito passa a procurar um novo criadouro para completar a desova. Quando a oviposição termina, o mosquito está livre para assumir novos comportamentos.

Se o mosquito não estiver executando nenhuma das atividades previstas anteriormente (alimentação, reprodução, oviposição) ou não estiver morto as questões relacionadas aos seguintes itens devem ser respondidas: *período fotofásico atual, estado de cópula, início da desova, verificação do momento de desovar, nível de energia atual, existência de seres humanos ou vegetação* nas proximidades. De acordo com as respostas à estas perguntas, os seguintes estados de comportamento podem ser elegidos: *Procurar sossego, Procurar criadouro para desova, Voar aleatoriamente, Morrer, Procurar alimento à base de seiva, Procurar mamífero para ingerir sangue, Tentar picar um mamífero e Tentar picar um mamífero rapidamente.*

A fim de ilustrar este modelo de tomada de decisão foram realizados alguns experimentos computacionais. A ilustração, descrição e execução destes experimentos são detalhados na seção 5.1 do capítulo 5 - *Experimentos computacionais.*

Agentes Mamíferos

Os mamíferos cachorros e gatos disponíveis no modelo de simulação *SimPopMosq* são agentes que inseridos em um determinado cenário de simulação são capazes de se locomover *aleatoriamente*, obedecendo às imposições físicas do ambiente. Basicamente, a diferença entre os cachorros e os gatos está na intensidade do odor, sendo que foi assumido que os cachorros liberam maior intensidade de odor, assim atraindo mais mosquitos *Aedes aegypti* do que os gatos.

Agente ser humano

O agente ser humano, além do comportamento de locomoção aleatória, possui em seu modelo, uma agenda capaz de registrar atividades para este agente. Uma atividade que pode ser agendada é o deslocamento entre dois pontos no cenário de simulação, como por exemplo sair de uma casa e ir até outra. O agente ser humano pode ter registrado

o conjunto de portas que este tem acesso (chave para abertura e fechamento). Estas características permitem que seres humanos transitem e consigam abrir e fechar portas do cenário de simulação.

Deslocamento do agente ser humano

O deslocamento do agente que representa o ser humano pode ser orientado a objetivos, diferentemente dos outros agentes. No modelo de simulação proposto, um sub-modelo de deslocamento baseado na teoria de grafos e na utilização de algoritmos de *menor caminho* é adotado. Assim, o agente humano é capaz de se deslocar entre dois pontos, sempre pelo menor caminho.

Modelo de transponibilidade

Um modelo de transponibilidade de corpos foi desenvolvido. Através de uma matriz centralizada - *Matriz de transponibilidade* - define-se quais objetos podem ser transpostos. Quando um agente decide se locomover em uma direção é verificado se o mesmo pode ocupar o espaço desejado. Isto inclui a análise da transponibilidade de eventuais elementos que estejam no espaço desejado. Caso, o agente não possa ocupar o espaço desejado, outras células vizinhas devem ser analisadas, até encontrar uma célula para a qual o agente seja capaz de se locomover.

Capítulo 4

Modelo de Implementação

Neste capítulo são apresentados os aspectos de implementação do simulador de população de mosquitos *Aedes aegypti*. O simulador é um aplicativo computacional formado pela combinação de modelos computacionais, inspirados em objetos, agentes e leis naturais do mundo real.

A estrutura do simulador é dividida em alguns elementos. A figura 4.1 ilustra os elementos que compõem esta estrutura.

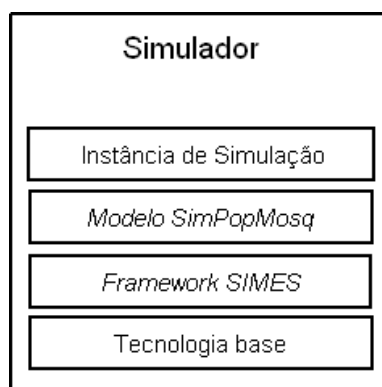


Figura 4.1: Composição tecnológica do simulador

Observando de forma ascendente os níveis da figura 4.1, verifica-se que no nível mais básico desta hierarquia, encontra-se o item *Tecnologia base*. Este item envolve os aspectos tecnológicos utilizados na construção e execução do simulador de população de mosquitos.

A camada *Framework SIMES* refere-se ao *framework* desenvolvido que dá suporte para desenvolvimento de modelos de simulação de sistemas ecológicos.

Em um nível mais acima encontra-se o modelo de simulação *SimPopMosq*. Este item foi apresentado no capítulo 3 em termos conceituais. Seus aspectos de implementação são discutidos neste capítulo.

No nível superior encontra-se o item *Instância de Simulação* que se refere à forma de construção dos cenários a serem simulados.

A seguir são descritas as camadas da figura 4.1, com seus respectivos elementos e propriedades.

4.1 Tecnologia base

Esta seção descreve as tecnologias utilizadas no desenvolvimento do simulador, bem como os ambientes computacionais necessários para a execução deste aplicativo e as conseqüências destas escolhas em termos de recursos computacionais.

4.1.1 Desenvolvimento de sistemas

Durante o desenvolvimento do simulador de população de mosquitos alguns *frameworks* para construção e simulação de sistemas multiagentes foram analisados com o objetivo de auxiliar na implementação do simulador de população de mosquitos. Entre os *framework* avaliados pode-se citar: i) Swarm [Minar et al., 1996, Swarm Development Group, 2004]; ii) Jade [Bellifemine et al., 1999]; iii) Mason [Luke et al., 2005]; iv) Repast [Collier, 2000, North et al., 2006]; v) NetLogo [NetLogo, 2007].

O simulador foi desenvolvido utilizando o *framework Repast* - versão 3.0 - como base auxiliar para a construção dos modelos de agentes. Vários fatores contribuíram para esta escolha, e pode-se destacar a modernidade da estrutura de orientação a objetos deste *framework*. A versão adotada do *framework Repast* foi escrita na linguagem de programação *Java*, criada pela empresa *Sun Microsystems* [Sun Microsystems, 2007]. Com isto, o simulador foi desenvolvido através da linguagem de programação *Java*.

Outro fator importante na escolha do *framework Repast* e conseqüentemente na escolha da linguagem de programação *Java* está relacionado ao fato das aplicações feitas em *Java* serem independentes de plataforma. Desta forma, o simulador de população de mosquitos *Aedes aegypti* pode ser executado em sistemas operacionais que sejam compatíveis com a máquina virtual *Java* sem a necessidade de alterações no código da aplicação. Durante a fase de testes o simulador foi executado nos sistemas operacionais *Microsoft Windows 2000 e XP*, *Linux Fedora Core 4*, *Linux Ubuntu 6.1* e *Linux Kurumin 5.0 e 6.0*.

O desenvolvimento do simulador de população de mosquitos *Aedes aegypti* ocorreu sobre a plataforma de desenvolvimento *Eclipse 3.2* [The Eclipse Foundation, 2007]. A tecnologia de controle de versão *Subversion* criada pela comunidade online *openCollabNet* [openCollabNet, 2007] foi adotada para controle interno no desenvolvimento do simulador.

A linguagem *UML*¹ foi adotada para modelar a implementação do simulador. Os principais recursos utilizados da linguagem *UML* foram os diagramas de *pacote*, *classe* e *seqüência*. Ferramentas diversas, como *Dia for Windows 0.95-1*, *JUDE Community 2.4.4* [Change Vision Inc, 2007], *Microsoft Visio 2000* [Microsoft Corporation, 2007] e *Rational Rose 2002* foram utilizadas para criar estes diagramas e outros fluxogramas apresentados neste texto.

4.1.2 Recursos computacionais

O desempenho do simulador desenvolvido depende do hardware adotado para sua execução e da complexidade do cenário a ser simulado. Experimentos realizados com o simulador em diferentes cenários levaram diversas horas para serem concluídos. O número de agentes e o tamanho do cenário simulado são os principais fatores que determinam a duração das simulações.

A arquitetura interna do aplicativo desenvolvido foi definida de maneira que as simulações são executadas de forma centralizada, ou seja, utilizam apenas um núcleo de processamento disponível no hardware em uso.

A quantidade de memória primária utilizada pelo simulador é relacionada com o tamanho do cenário a ser simulado e o número de elementos contidos neste cenário. Em média, ao simular os cenários apresentadas no capítulo 5 - *Experimentos computacionais* - o simulador utilizou menos de 128 *mega bytes* de memória primária.

4.2 Framework SIMES

Inicialmente foi proposto que o simulador de populações de mosquitos utilizasse diretamente o *framework* de simulações multiagentes *Repast* como infra-estrutura que auxiliasse a implementação dos seus agentes computacionais. Durante a construção do simulador percebeu-se que a arquitetura de classes, objetos e interfaces do *framework Repast* não era ideal, em termos de reutilização de código, para a construção de um complexo simulador que fosse escalável e flexível na construção de cenários de simulação. Diante disto, optou-se por estender o *framework Repast*.

A extensão do *framework Repast* é denominada *framework SIMES*². O *framework SIMES* induz os desenvolvedores a criarem o código da sua aplicação mais padronizado, e como consequência reutilizável. O *framework SIMES* pode ser utilizado como base para

¹Do inglês - Unified Modelling Language.

²O nome *SIMES* é uma abreviação das palavras em inglês: *Simulator*, *Ecological* e *Systems*.

a construção de outros modelos de simulação que possuam características desejadas e disponíveis neste *framework*.

Segundo Johnson um *framework*, quanto ao propósito, é um esqueleto de uma aplicação que pode ser herdado e personalizado por um desenvolvedor de sistemas [Johnson, 1997]. Johnson também define que quanto à estrutura, um *framework* é um projeto reutilizável total ou parcialmente e é representado por um conjunto de classes abstratas e interação entre suas instâncias. Considerando este raciocínio, o módulo *SIMES* desenvolvido durante a construção do simulador pode ser considerado um *framework* pois pode ser personalizado e/ou utilizado como infra-estrutura base na construção de modelos de simulação de outros agentes. Quanto à estrutura, o *framework SIMES* foi desenvolvido de forma flexível e parametrizada e é fundamentado em arquitetura de *Engenharia de Software Orientada a Agentes (ESOA)*³, o que permitirá que outros desenvolvedores criem seus modelos de simulação de maneira rápida e flexível.

O *framework SIMES* foi construído a partir de um conjunto de classes, objetos e interfaces disponíveis nas bibliotecas da máquina virtual *Java* e estendidas do *framework Repast*.

De forma geral, os modelos voltados para simulações de sistemas ecológicos, como por exemplo, os modelos de simulação de populações de insetos podem ser desenvolvidos sobre o *framework SIMES*. Outros problemas podem ser modelados e desenvolvidos usando o *framework SIMES*. É importante ressaltar que o *framework SIMES* é uma extensão do *framework Repast* e não uma substituição ou concorrência. O desenvolvimento do *framework SIMES* é um resultado complementar deste trabalho.

4.2.1 Itens estendidos do *framework Repast*

O *framework Repast* possui um conjunto de funcionalidades que auxiliam desenvolvedores na criação de seus modelos de simulação baseados em agentes. A seguir, as principais funcionalidades disponíveis e estendidas no *framework SIMES* são apresentadas:

1. *Ambiente 3D* - O *framework Repast* permite simular ambientes com apenas as dimensões x e y através de uma grade computacional com espaços discretos entre os elementos. Entretanto, as bibliotecas e classes desse *framework* não permitem simular mais de um agente/objeto na mesma célula. Diante disso, foi implementado no *framework SIMES* uma estrutura tridimensional para inserir agentes e objetos nos eixos x , y e z . Esta estrutura utiliza as classes da estrutura bidimensional

³Do inglês - Agent Oriented Software Engineering - AOSE.

do *framework Repast* e incorpora uma terceira dimensão, capaz de assumir valores contínuos.

2. *Rastro 3D* - As bibliotecas que auxiliam a simulação de difusão de substâncias no *framework Repast* se limitam a simular a difusão em um cenário bidimensional. Isto porque os cenários disponíveis neste *framework* são bidimensionais. Diante desta limitação é proposta uma extensão das classes do *framework Repast* que permite simular a difusão de substâncias em cenários tridimensionais (eixos x, y e z). Otimizações também foram realizadas nesta biblioteca, de maneira a tornar a execução das simulações mais rápida.
3. *Mecanismo de registro de saída de dados* - O *framework Repast* possui bibliotecas que permitem a criação dinâmica de gráficos que registram o comportamento de variáveis definidas. Estas bibliotecas permitem também registrar estes dados em arquivos especificados (memória secundária). Foi desenvolvido um mecanismo que acopla a leitura de parâmetros via arquivos XML, registro de variáveis em arquivos em disco e graficamente. Esse mecanismo torna mais flexível a especificação das variáveis a serem observadas e dispensa o trabalho de desenvolvimento de código, por parte do usuário, para gerar os dados de saída.

Algumas funcionalidades do *framework Repast* foram encapsuladas em classes do *framework SIMES*. O objetivo desta encapsulação foi aumentar a padronização e o reaproveitamento de código das aplicações desenvolvidas sobre o *framework SIMES*. Essas extensões envolvem os seguintes itens:

1. *Definição de interface gráfica 2D*.
2. *Leitura automática de parâmetros em arquivo XML*.
3. *Especificação dos escalonamentos de processos*.
4. *Definição de grafo para deslocamento de agentes*.

Algumas funcionalidades e definições desejáveis não são previstas no *framework Repast*. Diante disto, foram inseridas no *framework SIMES* as seguintes funcionalidades e definições:

1. *Definição de data e hora* - Não existe no *framework Repast* o conceito de data (ano, mês e dia) e hora (hora, minuto e segundo). Estes conceitos foram implementados através da conversão da unidade de tempo discreta *tick* do *framework Repast*.

2. *Diferenciação de agentes e objetos* - No *framework Repast* agentes e objetos são as mesmas coisas. Entretanto, no *framework SIMES* existe esta diferença e está relacionada ao fato dos agentes serem escalonados e os objetos não.
3. *Criação de agendas e tarefas* - Foi incorporado ao *framework SIMES* bibliotecas que permitem a criação de agenda de tarefas para agentes. Os agentes podem ter tarefas/atividades agendados para períodos especificados. As tarefas podem possuir pré-condições de execução.
4. *Matriz de transponibilidade* - Alguns elementos podem ocupar o mesmo espaço de outros. Diante disso, foi definido o conceito de transponibilidade e implementada no *framework SIMES* uma classe que permite especificar quais elementos podem ser transpostos.
5. *Geometria computacional* - Algumas classes para construção de objetos lineares e circulares em simulações como muros, paredes e lagos foram incorporadas ao *framework SIMES*. Estas classes possuem algoritmos geométricos, como o algoritmo de *Bresenham* para traçado de linhas [Bresenham, 1965].

4.2.2 Classes

A implementação do *framework SIMES* é complexa. Diversas perspectivas podem ser adotadas para ilustrar a arquitetura de um *framework*. Os seguintes artefatos são apresentados na tentativa de transmitir a lógica de implementação do *framework SIMES*:

1. *Descrição dos pacotes de classes e seus interesses*
2. *Divisão das classes nos pacotes*
3. *Descrição do objetivo das classes*
4. *Descrição dos relacionamentos entre as classes*
5. *Descrição dos relacionamentos entre as classes do framework SIMES e as classes do framework Repast*

O *framework SIMES* possui um conjunto de classes e interfaces que são agrupadas por pacotes. Cada pacote é relacionado a um interesse específico da aplicação. Os pacotes do *framework SIMES* podem ser visualizados na figura 4.2.

Os interesses de cada pacote do *framework SIMES* são descritos a seguir:

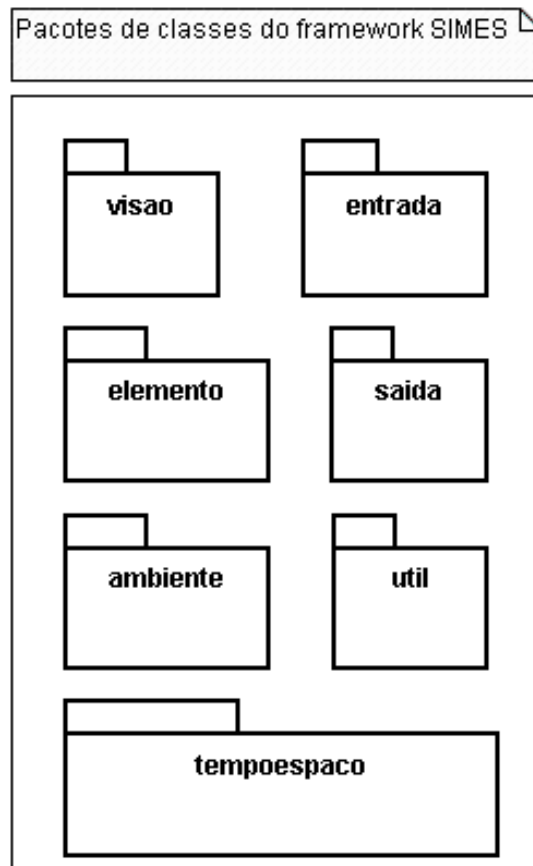


Figura 4.2: Diagrama de pacotes do *framework SIMES*

1. *tempoespaco* - Este pacote agrupa as classes relacionadas à definição de tempo, espaço e deslocamento. Os mecanismos de conversão da unidade de tempo discreta *tick* e data-hora de simulação são definidos. Define-se também os mecanismos de conversão do espaço discreto - células da *grade* - e a unidade de medida *metro*.
2. *ambiente* - As classes relacionadas ao ambiente (mundo) das simulações são definidas neste pacote. A *grade* para inserção dos agentes e objetos, modelo de difusão de rastro, e matriz de transponibilidade são conceitos implementados nas classes deste pacote.
3. *elemento* - Os elementos do simulador desenvolvido podem ser divididos em agentes e objetos. As classes abstratas que dão suporte à implementação de agentes e objetos estão inseridas neste pacote.
4. *visao* - Este pacote agrupa um conjunto de classes que padroniza a representação gráfica dos elementos do ambiente de simulação, através de classes abstratas.

5. *util* - Este pacote incorpora as classes que compreendem os requisitos não-funcionais do simulador.
6. *saida* - O mecanismo para coletar dados de variáveis, gerar gráficos dinâmicos e registrar as saídas em arquivos texto é implementando através das classes que compõem este pacote.
7. *entrada* - Os dados que alimentam o simulador e o modelo de simulação são definidos em classes que obedecem à padronização definida nas classes abstratas deste pacote.

Esta divisão de pacotes foi inspirada no padrão de arquitetura de software *MVC* (Model-view-controller) [Krasner and Pope, 1988]. Uma visão alternativa e aprofundada das classes e sub-pacotes dos pacotes apresentados anteriormente do *framework SIMES* é apresentada na figura 4.3.

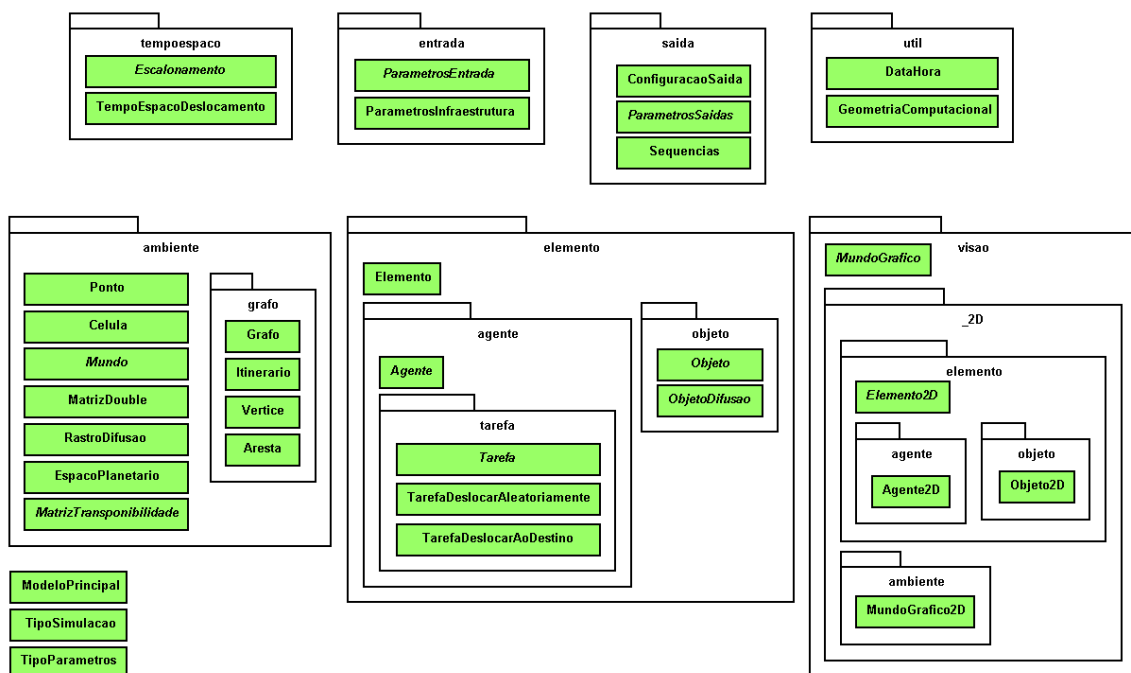


Figura 4.3: Diagrama de pacotes detalhado do *framework SIMES*

As classes pertencentes a este *framework* são descritas a seguir:

1. Pacote *tempoespaco*

- (a) *Escalonamento* - Classe abstrata desenvolvida para padronização de configurações de regras de escalonamentos de processos de agentes e objetos.

- (b) *TempoEspacoDeslocamento* - Define o tempo, espaço e deslocamento do simulador e seus elementos.

2. Pacote ambiente

- (a) *Ponto* - Estrutura de dados utilizada para representação de um ponto formado pelos eixos x e y .
- (b) *Celula* - Esta estrutura de dados é inserida em cada posição do “espaço planetário simulado”. Possui uma variável z que indica a sua altitude e uma lista que pode armazenar elementos de simulação, como agentes e objetos.
- (c) *EspacoPlanetario* - O espaço a ser simulado, representado por esta classe, é formado por uma grade, que por sua vez possui um conjunto de células em cada posição discreta da grade. Esta estrutura encapsula um objeto da classe *Object2DTorus* do framework *Repast*.
- (d) *Mundo* - Classe abstrata que padroniza os “mundos de simulação” a serem desenvolvidos. Dentre as principais padronizações está a definição do período de simulação, dimensões do mundo simulado e listas de elementos e rastros a serem inseridos nas simulações. Define também a utilização da estrutura de dados *EspacoPlanetario*.
- (e) *MatrizDouble* - Esta classe representa uma matriz do tipo “double” que possui otimizações voltadas para maior eficiência na difusão de rastros.
- (f) *RastroDifusao* - Classe responsável pela difusão de rastro no cenário de simulação. Esta classe estende a classe *Diffuse2d* disponível no framework *Repast*.
- (g) *MatrizTransponibilidade* - Estrutura de dados abstrata utilizada para definição da transposição entre elementos dos modelos de simulação.

3. Pacote ambiente.grafo

- (a) *Vertice* - Representação computacional de vértice de grafo.
- (b) *Aresta* - Representação computacional de aresta de grafo.
- (c) *Grafo* - Estrutura de dados que representa grafo direcionado. Internamente utiliza uma lista de vértices e uma lista de arestas.
- (d) *Itinerario* - Estrutura de dados que manipula grafos para encontrar itinerário válido entre pontos estabelecidos.

4. *Pacote elemento* - A classe abstrata *Elemento* padroniza a definição de elementos no *framework SIMES*.
5. *Pacote elemento.agente* - A classe abstrata *Agente* padroniza a definição de agentes nos modelos de simulação.
6. *Pacote ambiente.agente.tarefa*
 - (a) *Tarefa* - Classe abstrata que padroniza a definição de tarefas no *framework SIMES*.
 - (b) *TarefaDeslocarAleatoriamente* - Esta classe representa a tarefa *deslocar aleatoriamente*, que pode ser utilizada pelos agentes dos modelos de simulação.
 - (c) *TarefaDeslocarAoDestino* - Representação da tarefa *deslocar ao destino*, que pode ser utilizada pelos agentes dos modelos de simulação.
7. *Pacote ambiente.objeto*
 - (a) *Objeto* - Classe abstrata que padroniza a definição de objetos nos modelos de simulação.
 - (b) *ObjetoDifusao* - Extensão da classe *Objetos* que padroniza a definição de objetos capazes de difundir rastros.
8. *Pacote visao* - A classe abstrata *MundoGrafico* foi desenvolvida para padronização de modelos gráficos de simulação.
9. *Pacote visao.2D.ambiente*
 - (a) *MundoGrafico2D* - Especialização da classe *MundoGrafico* que encapsula objetos de classes do *framework Repast* para simulação bidimensional.
10. *Pacote visao.2D.elemento* - *Elemento2D* é uma estrutura de dados abstrata desenvolvida para padronização de representações gráficas dos elementos de simulação.
11. *Pacote visao.2D.elemento.agente* - *Agente2D* é uma estrutura de dados para representação gráfica dos agentes dos modelos de simulação.
12. *Pacote visao.2D.elemento.objeto* - *Objeto2D* é uma estrutura de dados para representação gráfica dos objetos dos modelos de simulação.
13. *Pacote util*

- (a) *DataHora* - Representação computacional de data e hora. Utiliza o formato “aaaa/mm/dd hh:mm:ss”.
- (b) *GeometriaComputacional* - Conjunto de algoritmos para definição de formas geométricas, como linhas, elipses, círculos e outros.

14. Pacote saída

- (a) *ConfiguracaoSaida* - Estrutura de dados enumerável que classifica a situação da “saídas de dados”.
- (b) *ParametrosSaidas* - Classe abstrata que padroniza a definição de “saídas de dados” e manipula estas saídas, armazenando-as em um *arquivo XML*.
- (c) *Sequencias* - Classe utilizada para encapsular a manipulação de gráficos de seqüência do *framework Repast*.

15. Pacote entrada

- (a) *ParametrosEntrada* - Classe abstrata que padroniza a definição de “entradas de dados” dos modelos de simulação.
- (b) *ParametrosInfraestrutura* - Estrutura de dados responsável pela manipulação dos parâmetros de infra-estrutura do *framework SIMES*.

Além das classes apresentadas pertencentes aos pacotes do *framework SIMES*, a classe abstrata *ModeloSimulacao* deve ser considerada. Esta classe é responsável pela padronização e definição de modelos de simulação especificados pelo usuário do *framework*. Todo modelo de simulação deve herdar esta classe e implementar seus métodos abstratos.

O relacionamento entre as classes do *framework SIMES* pode ser visualizado na figura 4.4.

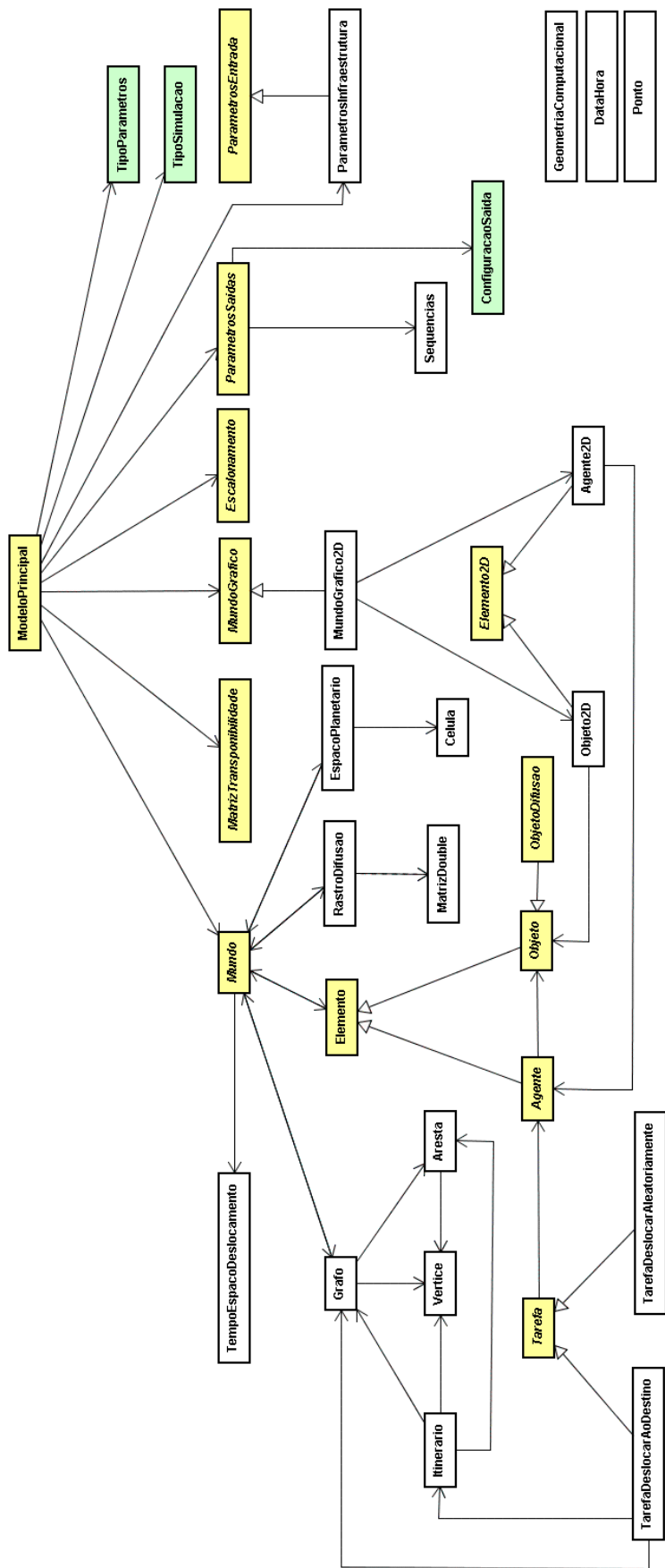


Figura 4.4: Diagrama de classes do *framework SIMES*

Observa-se na figura 4.4 que a classe *ModeloSimulacao* associa-se a um conjunto de classes abstratas necessárias para o desenvolvimento do modelo de simulação, que são: *Mundo*, *MatrizTransponibilidade*, *MundoGrafico*, *Escalonamento*, *ParametrosSaida* e *ParametrosEntrada*. Dentre estas classes, a classe *Mundo* é a mais importante, pois define os elementos, rastros e espaço planetário da simulação (grade).

Os retângulos de cor amarela representam as classes abstratas, os azuis representam as classes enumeráveis e os demais (brancos) representam as classes concretas do *framework SIMES*.

As classes do *framework SIMES* que encapsulam ou relacionam-se com classes e objetos do *framework Repast* podem ser visualizadas na figura 4.5. O objetivo desta figura é ilustrar o relacionamento entre as classes destes dois *frameworks*.

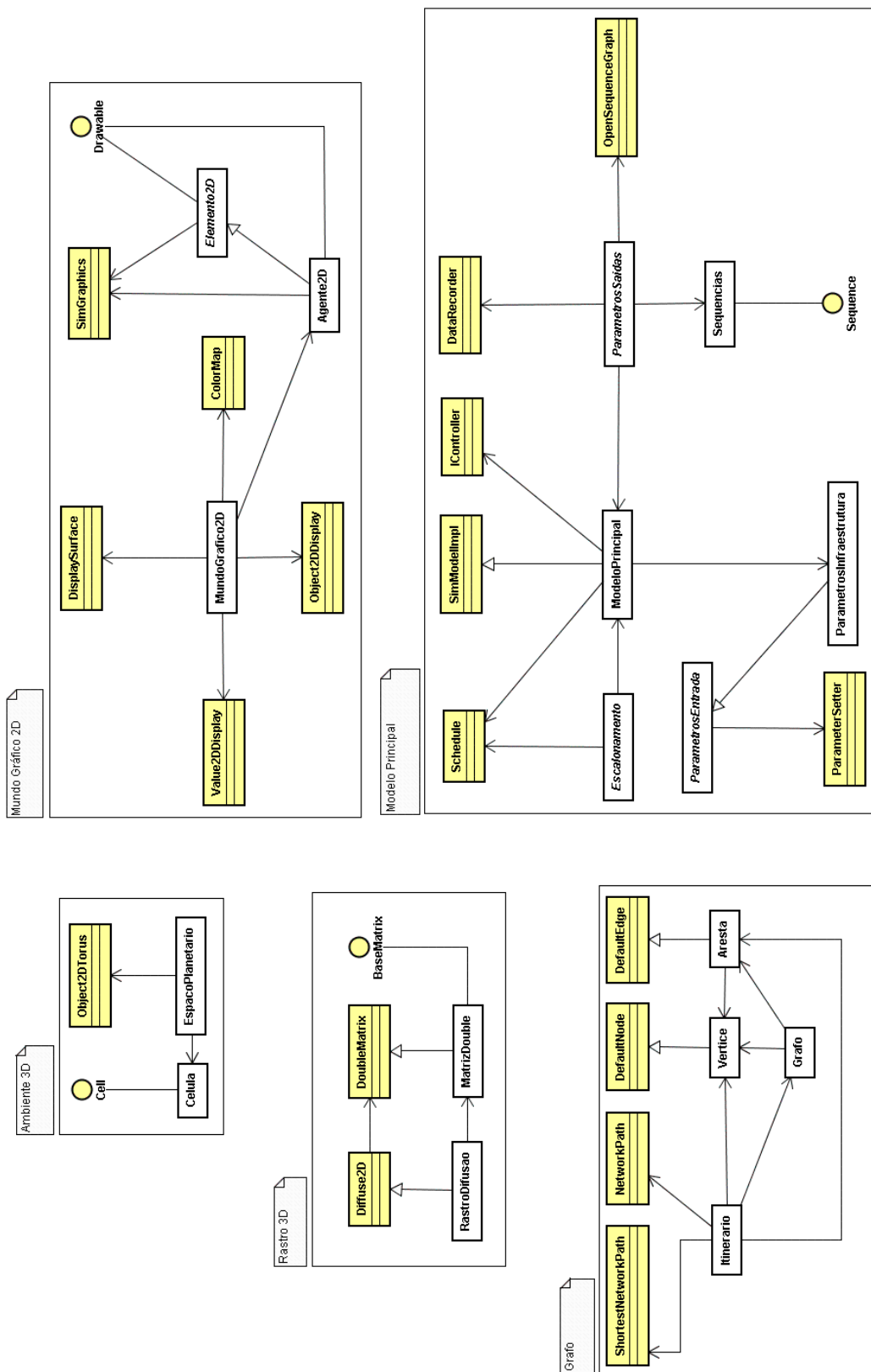


Figura 4.5: Diagrama de classes: Relacionamento entre as classes dos frameworks *Repast* e *SIMES*

Na figura 4.5, as classes de cor amarela pertencem ao *framework Repast*, enquanto as demais pertencem ao *framework SIMES*. Observa-se que o contato entre esses *frameworks* se dá nos seguintes itens: i) Representação do ambiente; ii) Representação gráfica; iii) Definição do rastro; iv) Estrutura de grafo; v) Padronização do modelo de simulação.

O *framework SIMES* é uma extensão do *framework Repast*. Esta extensão padroniza o desenvolvimento de modelos de simulação tornando mais rápido seu desenvolvimento e aumentando o grau de reusabilidade de códigos.

4.3 Modelo de simulação *SimPopMosq*

Dentre os elementos necessários para o funcionamento do simulador de população de mosquitos destaca-se o modelo de simulação *SimPopMosq*. O propósito do simulador é relacionado com os requisitos funcionais deste modelo. O modelo *SimPopMosq* consiste em um conjunto de classes implementadas na linguagem de programação *Java* sobre os padrões dos *frameworks SIMES e Repast*.

A figura 4.1 ilustra os itens relevantes e adotados no desenvolvimento do simulador proposto. Os itens discutidos *Tecnologia base e framework SIMES* servem como base para o funcionamento do modelo de simulação *SimPopMosq*. A maioria das classes do modelo *SimPopMosq* são representações computacionais dos agentes e objetos do modelo de simulação. O principal benefício obtido a partir do uso da arquitetura de software adotada (ou seja, modelo de simulação desenvolvidos sobre o *framework SIMES*) é permitir que os modelos de simulação sejam construídos com foco maior na lógica do modelo de simulação e menos foco em requisitos não-funcionais.

A lógica de programação das classes do modelo *SimPopMosq* foi definida de acordo com o modelo conceitual descrito no capítulo 3 deste trabalho. A seguir são apresentadas essas classes.

4.3.1 Classes

A descrição da implementação do modelo *SimPopMosq* se resume na apresentação do seguintes itens:

1. *Descrição dos pacotes de classes e seus interesses*
2. *Divisão das classes nos pacotes*
3. *Descrição dos relacionamentos entre as classes*

4. Descrição dos relacionamentos entre as classes do modelo *SimPopMosq* com as classes do framework *SIMES*

As classes do modelo *SimPopMosq* foram agrupadas em pacotes e podem ser visualizadas na figura 4.6. O critério adotado para o agrupamento das classes foi o mesmo adotado nas classes do *framework SIMES*.

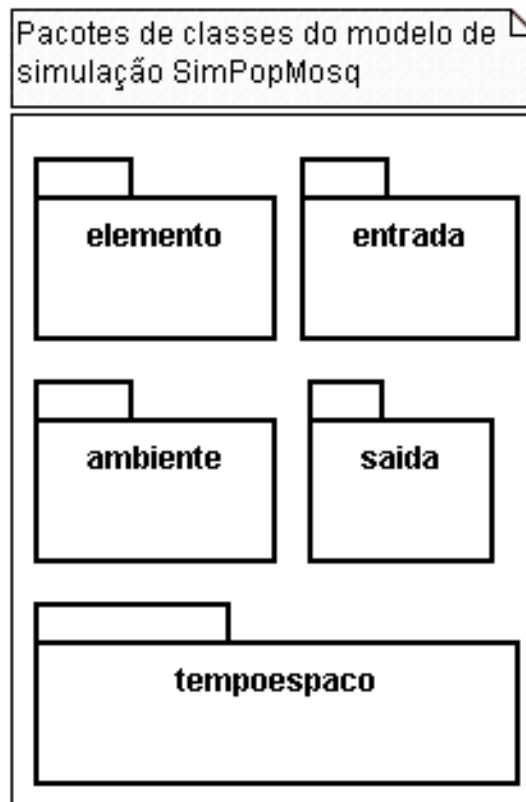


Figura 4.6: Diagrama de pacotes do modelo *SimPopMosq*

Uma visão alternativa e aprofundada das classes e subpacotes dos pacotes apresentados na figura 4.6 *framework SIMES* é definida na figura 4.7.

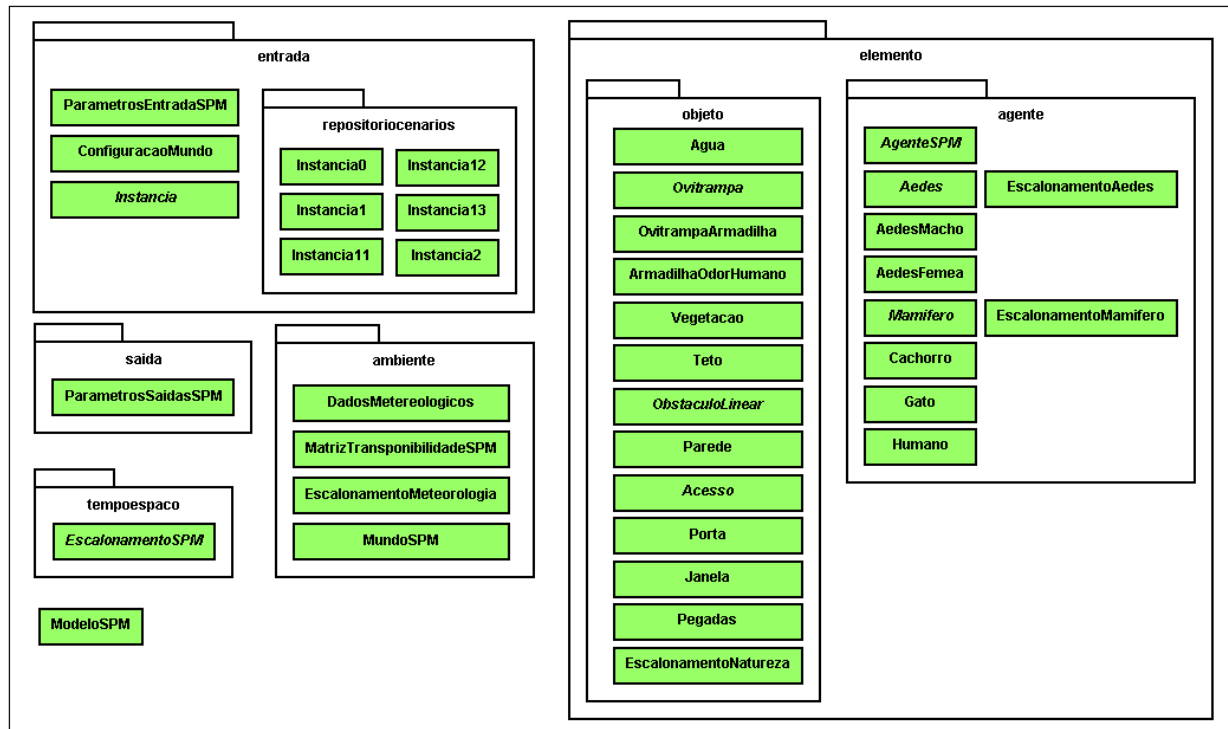


Figura 4.7: Diagrama de pacotes detalhado do modelo *SimPopMosq*

Observa-se na figura 4.7 o novo pacote *repositoriocenarios*. Este pacote é utilizado para agrupar as classes que representam cenários de simulações. O pacote *objeto* agrupa as classes que representam os objetos da simulação e o pacote *agente* agrupa os agentes. As classes dos demais pacotes são classes concretas que implementam os métodos das suas respectivas classes abstratas (base) definidas no *framework SIMES*.

A figura 4.8 apresenta o diagrama de classes - versão simplificada - do modelo de simulação *SimPopMosq*.

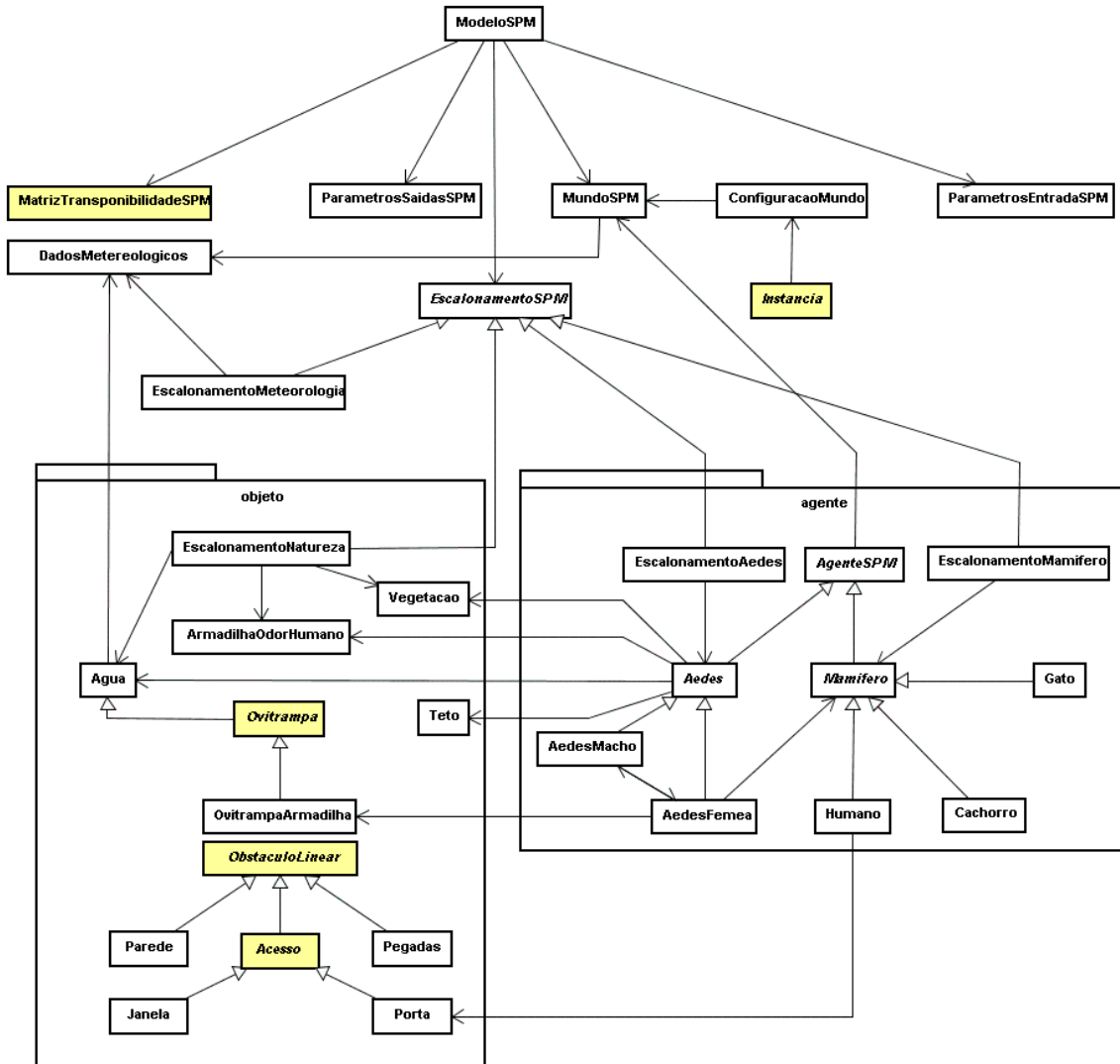


Figura 4.8: Diagrama de classes do modelo *SimPopMosq*

Nota-se no diagrama da figura 4.8 que os principais relacionamentos ocorrem entre os agentes e objetos, principalmente os agentes que representam os mosquitos *Aedes aegypti*. Algumas heranças são definidas entre algumas classes, como as classes *Agua*, *Obstaculo-Linear* e suas especializações. As classes *Cachorro*, *Gato* e *Humano* são especializações da classe *Mamifero*. Outra característica nítida neste diagrama é a relação entre as classes *Agua*, *Mundo* e *DadosMeteorologicos*.

A figura 4.9 ilustra o relacionamento entre as classes do modelo *SimPopMosq* e as classes do *framework SIMES*.

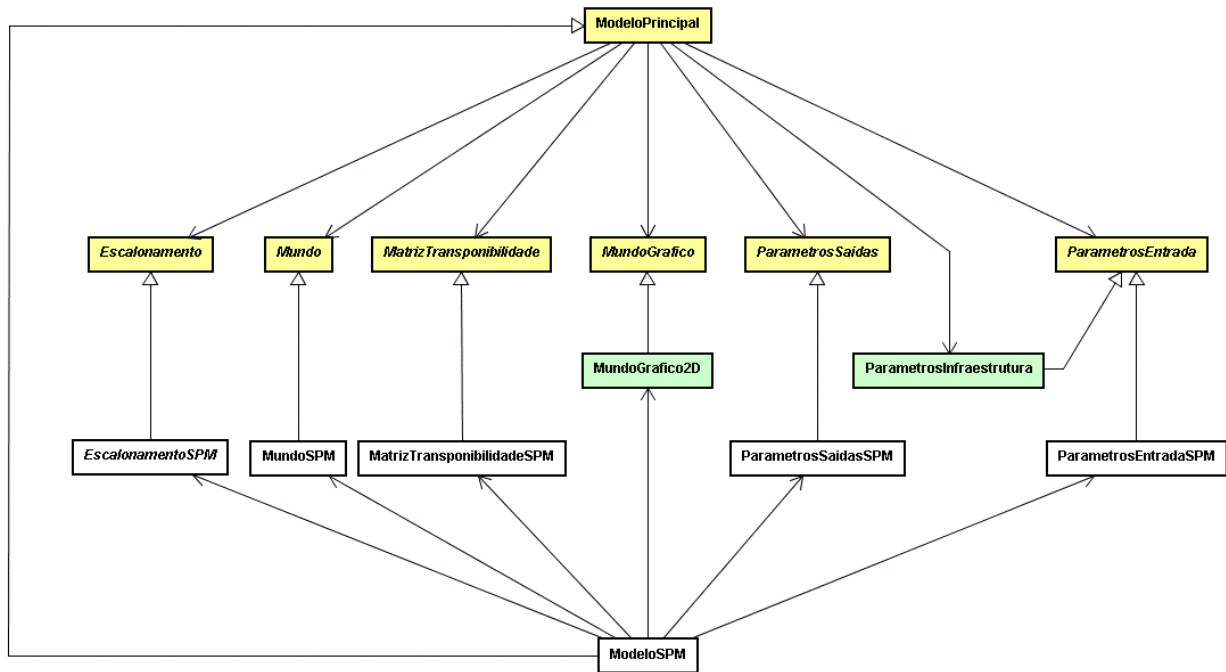


Figura 4.9: Diagrama de classes: Relacionamento entre as classes dos *SIMES* e o modelo de simulação *SimPopMosq*

As principais relações entre as classes da figura 4.9 são baseadas em herança. No modelo implementado as heranças entre as classes estabelecem um padrão de desenvolvimento forçando os desenvolvedores de modelos de simulação se adequarem aos métodos previamente definidos nas classes base.

Os retângulo de cor amarela representam as classes abstratas do *framework Repast*, os azuis representam as classes concretas e os demais (brancos) representam as classes do modelo de simulação *SimPopMosq*.

Existem diversos outros relacionamentos entre as classes do simulador proposto mas somente os relacionamentos mais importantes foram definidos nos diagramas de classes. Caso todos os relacionamentos, atributos e métodos fossem adicionados nos diagramas, estes diagramas se tornariam complexos, de forma que a sua visualização em um plano bidimensional seria difícil.

4.4 Simulação

A figura 4.10 ilustra de forma geral o funcionamento do simulador de população de mosquitos.

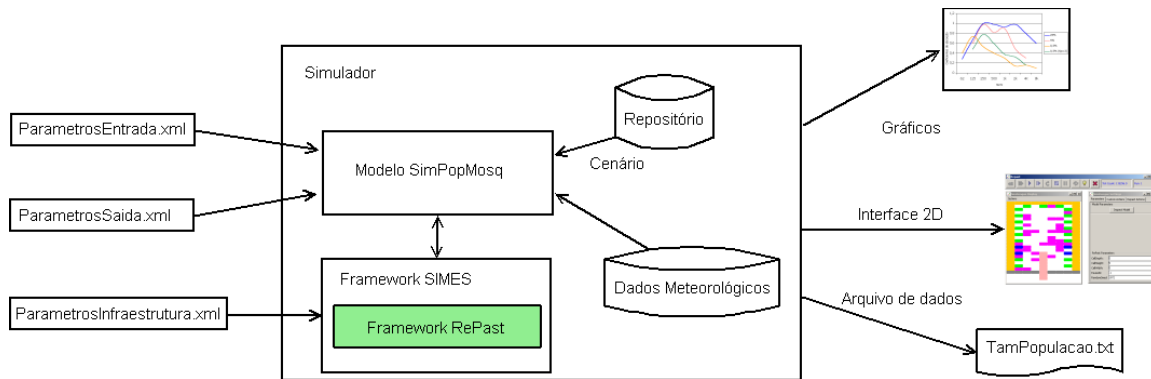


Figura 4.10: Visão geral do simulador de população de mosquitos

Através da figura 4.10 observa-se que o simulador é composto basicamente pelo *framework SIMES* e pelo modelo de simulação *SimPopMosq*. Os itens necessários para o funcionamento do modelo *SimPopMosq* são: *ParametrosEntrada.xml*, *ParametrosSaida.xml*, *repositório* e dados meteorológicos (formato XML). Estes itens já foram descritos no capítulo 3. Os parâmetros do modelo de simulação são definidos no arquivo *ParametrosEntrada.xml* e as saídas desejadas (a serem observadas) são definidas no arquivo *ParametrosSaida.xml*. Nota-se a existência de um arquivo chamado *ParametrosInfraestrutura.xml*. Neste arquivo são armazenados dados relativos à infra-estrutura de funcionamento do *framework SIMES*.

O simulador de população de mosquitos é capaz de registrar o comportamento do modelo de simulação através de gráficos *online* (gerados em tempo de execução), interface gráfica *2D* com representação dos elementos em simulação, e por fim, através de arquivos textos (*txt*) que armazenam numericamente os dados observados. Estes arquivos podem ser importados por planilhas eletrônicas e seus dados analisados.

Para inicializar o modelo de simulação *SimPopMosq* sobre o *framework SIMES*, uma classe chamada *Principal* foi definida. Esta classe possui um método *main* e recebe como parâmetro os arquivos XML a serem utilizados pelo simulador e então inicializa a simulação.

4.4.1 Classes

Algumas classes envolvidas na configuração de cenários e execução de simulações podem ser visualizadas no diagrama de classes da figura 4.11.

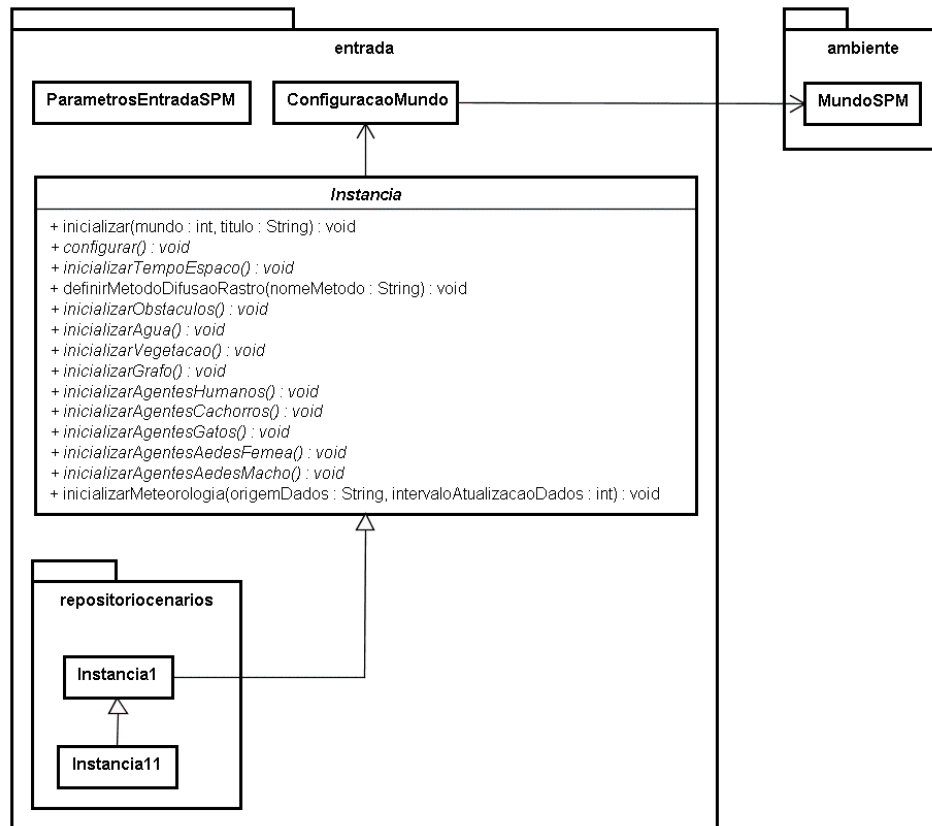


Figura 4.11: Diagrama de classes: Instância de simulação

Na figura 4.11 nota-se a presença dos pacotes *entrada* e *ambiente*. O pacote *entrada* contém classes que manipulam a criação de cenários de simulação. O sub-pacote *repositoriocenarios* possui as classes que representam os cenários de simulação.

Todo cenário de simulação a ser desenvolvido deve ser estendido da classe *Instancia*, e conseqüentemente deve implementar seus métodos abstratos. Observa-se que cada método desta classe define elementos a serem inseridos no cenário de simulação, como por exemplo: *inicializarAgua*, *inicializarAgentesAedes*, *inicializarVegetacao*, etc.

Capítulo 5

Experimentos Computacionais

Foram realizados experimentos computacionais com o objetivo de ajustar os parâmetros, validar e aplicar o modelo de simulação desenvolvido. Três grupos de experimentos foram feitos, a saber: experimentos qualitativos, experimentos básicos de calibração e validação e estudos de caso.

5.1 Experimentos qualitativos - Ilustração do modelo

Este grupo de experimentos foi feito para ilustrar o funcionamento dos modelos de racionalidade dos agentes *Aedes aegypti* macho e fêmea.

5.1.1 Questionário de decisão do mosquito *Aedes aegypti* macho

Os agentes mosquitos *Aedes aegypti* macho possuem os seguintes comportamentos:

1. Procurar alimento.
2. Procurar um local para descanso.
3. Procurar uma fêmea para se acasalar.

Este experimento foi realizado para ilustrar a tomada de decisão do agente computacional que representa o mosquito *Aedes aegypti* macho. As seguintes características são consideradas:

1. A simulação ocorreu no período virtual entre 18 horas e 30 minutos e 19 horas e 1 segundo, totalizando 30 minutos e 1 segundo virtuais de experimentação.

2. O cenário simulador é composto por:

- (a) Um quarto fechado (paredes e teto) com as dimensões: 10 m de largura, 10 m de comprimento e 3,3 m de altura.
- (b) Um mosquito *Aedes aegypti* macho - O mosquito macho foi adicionado no cenário com pouca energia. Este valor foi definido como “baixo” para ilustrar o interesse do mosquito em relação à alimentação.
- (c) Um mosquito *Aedes aegypti* fêmea - Inicialmente adicionou-se uma fêmea do mosquito *Aedes aegypti* com o nível de energia “alto” recém transformada em adulta. Desta maneira, espera-se que a fêmea fique alternando entre os comportamentos “Voar aleatoriamente” (enquanto o nível de energia estiver alto) e “Procurar alimento à base de seiva” (quando o nível de energia estiver médio ou baixo).
- (d) Uma vegetação - Adicionou-se vegetação no chão do quarto simulado com as dimensões: 0,5 m de comprimento, 0,5 m de largura e 0,5 m de altura. Esta vegetação serve de alimento para os agentes.

Definidas as condições iniciais, a simulação é inicializada. Na figura 5.1 é possível visualizar a movimentação dos mosquitos no cenário ao longo do tempo. O ponto de cor cinza representa o mosquito *Aedes aegypti* macho, o ponto de cor rosa representa a fêmea e o ponto de cor verde representa a vegetação. Os pontos de cor amarela representam as paredes do cenário simulado.

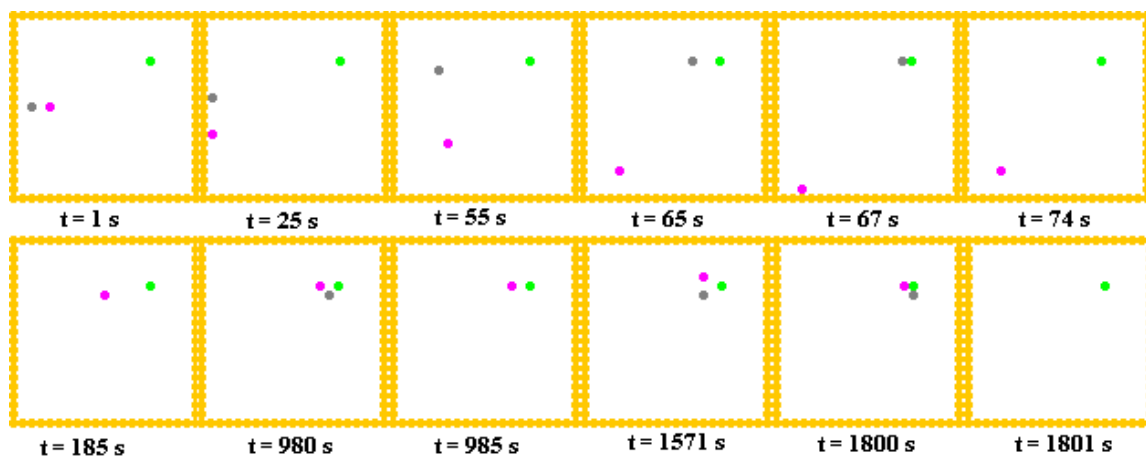


Figura 5.1: Simulação do mosquito *Aedes aegypti* macho: execução detalhada

A figura 5.1 apresenta diferentes instantes de simulação (t). No instante $t = 1$ s os elementos são posicionados no cenários de simulação.

O foco deste experimento está no mosquito *Aedes aegypti* macho. Logo, a seguinte pergunta é formulada: “o que o mosquito *Aedes aegypti* macho irá fazer nos próximos passos?”. Como descrito na seção 3.3.3 o mosquito *Aedes aegypti* macho escolhe o comportamento a ser executado baseando-se em um conjunto de regras e condições externas. O questionário de decisão do mosquito, ilustrado na figura 3.13, incorpora esta lógica através de perguntas. As respostas a estas perguntas determinam o comportamento que o mosquito executará.

No questionário de decisão do mosquito a primeira pergunta se refere ao atual comportamento (estado) do mosquito. A resposta para esta pergunta é “Nenhum”, ou seja, o mosquito inicialmente não está fazendo nada. Diante desta resposta o fluxo do modelo leva à segunda pergunta: “O período atual é fotofásico (dia)?”. Como definido na configuração deste experimento o período de simulação inicia-se às 18 horas e 30 minutos. Este período é considerado fotofásico. Com esta resposta a seguinte pergunta deve ser respondida: “Qual é o nível atual de energia do mosquito?”. Foi definido inicialmente que o nível de energia do mosquito é “baixo”. A quarta e última pergunta é realizada: “Qual é a intensidade do rastro deixado no ar pelas seivas de plantas?”. Pela distância do mosquito em relação à vegetação do cenário, a intensidade do rastro desta vegetação no ponto em que se encontra o mosquito é “baixa”. Com esta última resposta o mosquito *Aedes aegypti* macho muda seu comportamento (estado) para “Procurando alimento à base de seiva”.

A figura 5.2 ilustra a seqüência de perguntas feitas ao mosquito *Aedes aegypti* macho neste primeiro momento.

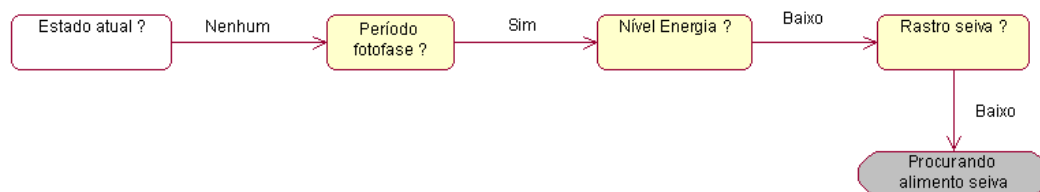


Figura 5.2: Simulação do mosquito *Aedes aegypti* macho: primeiro momento

Diante do comportamento assumido pelo mosquito *Aedes aegypti* macho, observa-se na figura 5.1 que este mosquito se aproxima da vegetação entre os instantes $t = 25$ s e $t = 74$ s, quando efetivamente começa sua alimentação à base de seiva na vegetação disponível no cenário simulado. Nota-se que o mosquito preferiu alimentar-se a acasalar

com a fêmea, mesmo a fêmea estando mais próxima do macho do que a própria vegetação.

A cada 10 segundos o questionário de decisão dos mosquitos *Aedes aegypti* é “executado”. No instante $t = 970$ s a execução das perguntas do questionário resultam na escolha do comportamento/estado “Nenhum”. Isto porque neste instante o mosquito acaba de finalizar sua alimentação. No instante $t = 980$ s o mosquito tem seu questionário de decisão novamente executado. Assim, a primeira pergunta neste instante é: “Qual é o estado atual do agente?”. A resposta é “Nenhum” (comportamento definido no instante $t = 970$ s). Diante desta resposta o fluxo do modelo leva à segunda pergunta: “O período atual é fotofásico (dia)?”. A resposta é “Sim” pois neste instante ($t = 980$ s) são aproximadamente 18 horas e 46 minutos (horário inferior às 19 horas). Com esta resposta a seguinte pergunta deve ser respondida: “Qual é o nível atual de energia do mosquito?”. Como o mosquito acabou de se alimentar, a resposta é “Alto”. A quarta e última pergunta é realizada: “Qual é a intensidade do rastro sonoro emitido pelas batidas de asas da fêmea?”. Pela distância do mosquito em relação à fêmea, a intensidade do rastro sonoro desta fêmea no ponto em que se encontra o mosquito é “alta”. Com esta última resposta o mosquito *Aedes aegypti* macho muda seu comportamento (estado) para “Tentando acasalar”.

A figura 5.3 ilustra a seqüência de perguntas feitas ao mosquito *Aedes aegypti* macho neste segundo momento.

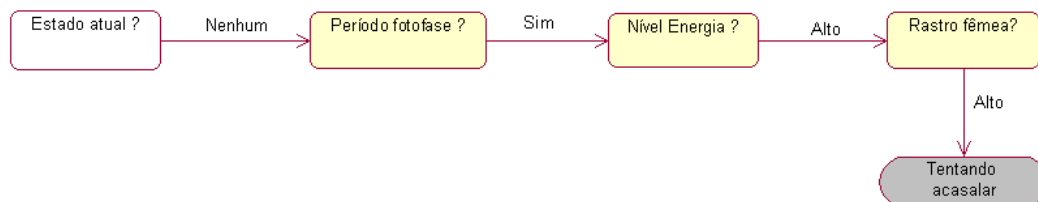


Figura 5.3: Simulação do mosquito *Aedes aegypti* macho: segundo momento

5.1.2 Questionário de decisão do mosquito *Aedes aegypti* fêmea

De maneira similar à apresentada na seção anterior, o questionário de decisão do mosquito *Aedes aegypti* fêmea será apresentado nesta seção. A fêmea do *Aedes aegypti* possui os seguinte comportamentos:

1. Procurar alimento.
2. Procurar um local para descanso.

3. Acasalar.
4. Procurar criadouro para desova.

Foi realizado um experimento computacional para ilustrar a tomada de decisão dos agentes que representam os mosquitos *Aedes aegypti* fêmea. Neste experimento as seguintes características são consideradas:

1. A simulação ocorreu no período virtual entre 9 e 10 horas da manhã, totalizando 3600 segundos virtuais de experimentação em período fotofásico (dia).
2. O cenário simulado é composto por:
 - (a) Um quarto fechado (paredes e teto) com as dimensões de 10 *m* de largura, 10 *m* de comprimento e 3,3 *m* de altura. No centro do quarto adicionou-se uma parede (obstáculo).
 - (b) Um mosquito *Aedes aegypti* macho - O mosquito macho foi adicionado no cenário com nível de energia elevado. Este valor foi definido “alto” para ilustrar o desinteresse imediato deste mosquito em relação à alimentação.
 - (c) Um mosquito *Aedes aegypti* fêmea - Adicionou-se uma fêmea do mosquito *Aedes aegypti* com o nível de energia “médio” recém transformada em adulta.
 - (d) Uma vegetação - Adicionou-se vegetação no chão do quarto simulado com as dimensões: 0,5 *m* de comprimento, 0,5 *m* de largura e 0,5 *m* de altura. Esta vegetação serve de alimento para os agentes.
 - (e) Um reservatório de água - Adicionou-se um reservatório de água no chão do quarto simulado com as dimensões: 0,5 *m* de comprimento, 0,5 *m* de largura e 0,5 *m* de altura.
 - (f) Um ser humano com altura igual a 1,8 *m* foi adicionado ao cenário de simulação. Este ser humano mantém-se estático no cenário e serve como alimento para a fêmea do mosquito *Aedes aegypti*.

Definidas as condições iniciais, a simulação é inicializada. Na figura 5.4 é possível visualizar a movimentação dos mosquitos ao longo do tempo. O ponto de cor cinza representa o mosquito *Aedes aegypti* macho, o ponto de cor rosa representa a fêmea, o ponto de cor verde representa a vegetação, o ponto de cor azul representa o reservatório de água e o ponto de cor vermelha representa o ser humano. Os pontos de cor amarela representam as paredes (obstáculos) do cenário simulado.

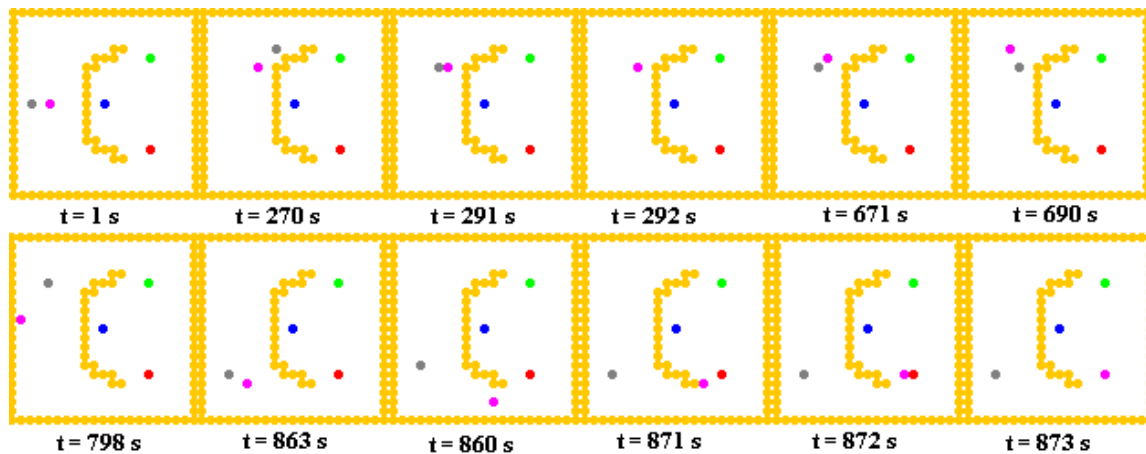


Figura 5.4: Simulação do mosquito *Aedes aegypti* fêmea: execução detalhada

A figura 5.4 apresenta diferentes instantes de simulação (t). No instante $t = 1 s$ os elementos nos cenários de simulação são posicionados. A partir deste instante e considerando as condições iniciais do experimento “como a fêmea do mosquito *Aedes aegypti* irá se comportar nos próximos instantes?”. Conforme descrito na seção 3.3.3 o mosquito *Aedes aegypti* do sexo feminino escolhe o comportamento a ser executado baseando-se em um conjunto regras e condições externas. O questionário de decisão do mosquito, ilustrado na figura 3.14, incorpora esta lógica através de um questionário com perguntas e respostas. As respostas determinam o comportamento que o mosquito executará.

No questionário de decisão do mosquito *Aedes aegypti* fêmea a primeira pergunta se refere ao atual comportamento (estado) do mosquito. A resposta para esta pergunta é “Nenhum” pois inicialmente o mosquito não está fazendo nada. Diante desta resposta o fluxo do modelo leva à segunda pergunta: “O período atual é fotofásico (dia)?”. Como definido na configuração deste experimento o período de simulação inicia-se às 9 horas e finaliza às 10 horas. Este período é considerado fotofásico. Com esta resposta a seguinte pergunta deve ser respondida: “Qual é o nível atual de energia do mosquito?”. Foi definido inicialmente que o nível de energia do mosquito é “médio”. A quarta e última pergunta é realizada: “A fêmea foi copulada (está grávida)?”. A resposta é “Não”, pois a fêmea ainda não teve contato com o macho. Com esta última resposta o mosquito *Aedes aegypti* fêmea muda seu comportamento (estado) para “Procurando alimento à base de seiva”.

A figura 5.5 ilustra a seqüência de perguntas feitas ao mosquito *Aedes aegypti* fêmea neste primeiro momento.

O mosquito *Aedes aegypti* fêmea procura por vegetação para se alimentar. No entanto, o mosquito macho assumiu o comportamento “Tentar acasalar” e com isto, insiste em perseguir a fêmea para se acasalar. A fêmea cede às pressões do macho e interrompe a

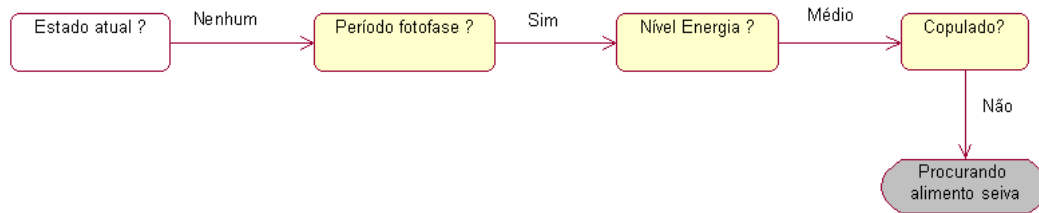


Figura 5.5: Simulação do mosquito *Aedes aegypti* fêmea: primeiro momento

procura por alimentação à base de seiva para se acasalar. O ato sexual dos mosquitos se inicializa no instante $t = 292$ s. A fêmea e o macho assumem o comportamento “Reproduzindo”.

A cada 10 segundos o questionário de decisão dos mosquitos *Aedes aegypti* é “executado”. No instante $t = 660$ s a execução das perguntas do questionário resultam na escolha do comportamento/estado “Nenhum”. Isto porque neste instante os mosquitos finalizaram o coito e não estão fazendo nada. No instante $t = 670$ s os mosquitos têm seu questionário de decisão executado novamente. A primeira pergunta neste instante sobre o mosquito *Aedes aegypti* fêmea é: “Qual é o estado atual do agente?”. O questionário leva à segunda pergunta: “O período atual é fotofásico (dia)?”. A resposta é “Sim” pois o experimento foi realizado no período fotofásico e com esta resposta a seguinte pergunta deve ser respondida: “Qual é o nível atual de energia do mosquito *Aedes aegypti* fêmea?”. Neste instante o nível de energia do mosquito é “médio”. A quarta pergunta é realizada: “A fêmea está grávida?”. A resposta é “Sim”. Outras perguntas/respostas relacionadas à inicialização do processo de desova e intensidade de rastro de odor humano estabelecem o comportamento “Procurar alimento animal”. Este comportamento leva a fêmea a procura por alimentação à base de sangue. A procura é finalizada no instante $t = 873$ s, quando a fêmea pica o ser humano. A partir deste instante, o comportamento “Alimentando” é assumido pela fêmea.

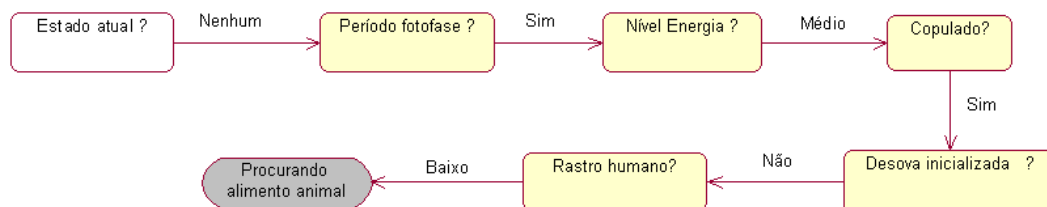


Figura 5.6: Simulação do mosquito *Aedes aegypti* fêmea: segundo momento

A figura 5.6 ilustra a seqüência de perguntas feitas ao mosquito *Aedes aegypti* fêmea neste segundo momento.

Algumas perguntas no questionário de decisão da fêmea relacionadas à desova são consideradas. Em algumas partes do questionário existe a seguinte pergunta: “A desova foi inicializada?”. Se a resposta para esta pergunta for “Sim”, então uma segunda pergunta é feita: “É momento de desovar?”. A resposta para esta pergunta é baseada no modelo de distribuição de ovos adotado (descrito no apêndice B) e o número de ovos colocados pelo mosquito *Aedes aegypti* fêmea. Caso seja momento de desovar então o agente irá assumir o comportamento “Procurar criadouro para desova”.

5.2 Experimento *Green-House* - Validação do modelo

Os experimentos realizados para calibrar e validar o modelo de simulação proposto foram baseados em dados obtidos em um experimento biológico real. Os dados referentes a este experimento foram fornecidos pelo professor doutor Álvaro Eduardo Eiras, pesquisador do departamento de Parasitologia do Instituto de Ciências Biológicas (ICB) da Universidade Federal de Minas Gerais (UFMG). Este experimento é referenciado neste texto como *Green-House*, nome originalmente dado pelo biólogos que o realizaram.

5.2.1 Dados experimentais reais

Visão geral

O experimento *Green-House* foi realizado na Universidade de Regensburg (Alemanha) durante 159 dias em uma estufa (casa), com plantas, recipientes de água e mosquitos *Aedes aegypti*. A configuração da casa foi proposta de forma a aproximar-se de um ambiente tropical. Durante este tempo, um aluno participante do projeto de pesquisa *Green-House*, se dispôs a alimentar as fêmeas do mosquito *Aedes aegypti* com o próprio sangue, através de picadas destes insetos ¹ durante a realização do experimento. Neste período de experimentação foram coletados diversos dados relacionados ao desenvolvimento da população de mosquitos, bem como a eficácia de uma armadilha no controle populacional dos mosquitos.

¹Os mosquitos não estavam contaminados com dengue e nem uma outra doença.

Detalhes do Experimento

Inicialmente foram soltas dentro da casa experimental (*Green-House*) 50 fêmeas do mosquito *Aedes aegypti* famintas por sangue. Quase todos os dias o aluno experimentador entrava duas vezes na casa:

i) Na parte da manhã entre 9 e 11 horas, o voluntário permitia que algumas fêmeas se alimentassem do seu sangue. O estudante permanecia de 5 a 30 minutos dependendo da quantidade de picadas que este conseguia tomar. No máximo 15 mosquitos se alimentaram de sangue por dia.

ii) No período da tarde, entre 15 e 19 horas, o voluntário entrava na casa para inferir a taxa de picadas dos insetos ² durante um período de 5 a 30 minutos. As fêmeas que pousavam sobre o ser humano eram imediatamente coletadas antes mesmo de picar e eram soltas depois na *Green-House*. Desta maneira, foi possível calcular a taxa de picadas - inferida a partir da taxa de pousos - sem a necessidade do ser humano tomar novas picadas. Após calcular estes valores, os reservatórios de água - criadouros dos mosquitos - foram analisados, contabilizando o número de ovos e exúvias ³. Assim, estes ovos e exúvias eram removidos da *Green-House*, sendo que uma semana depois os ovos eram re-introduzidos na casa e colocados em um recipiente com água. No período vespertino, as larvas do mosquito *Aedes aegypti* eram alimentadas com comida de peixe.

Uma foto da estufa utilizada para realização do experimento pode ser visualizada através da figura 5.7. A *Green-House* possui 11 m de comprimento e 3,8 m de largura. A sua altura no menor ponto mede 2,8 m e no maior ponto 3,8 m. Durante os experimentos a temperatura variou entre 22 e 30 °C e a umidade relativa do ar permaneceu entre 58 e 79 %. Cerca de cinco recipientes de água foram instalados no interior da casa. Diversas plantas foram distribuídas no ambiente.

Dados coletados

Os dados experimentais obtidos fornecem informações sobre a produção de ovos, a produção de exúvias, a taxa de picadas e o número de picadas diárias⁴. Estes dados foram usados como referência para calibração e validação do modelo de simulação proposto.

²Nos experimentos realizados, a taxa de picadas corresponde ao número de picadas que o indivíduo sofre durante um minuto.

³Exúvias são restos materiais que sobram de um mosquito quando este no estado de pupa se transforma em adulto. O número de exúvias coletadas serve indiretamente como uma medida de quantos adultos foram produzidos.

⁴Embora se tenha o valor da taxa de picadas e número de picadas no experimento da *Green-House*, a duração do período de picadas é uma informação não disponível.



Figura 5.7: Green-House: O cenário de experimentos reais

Períodos de experimentação

O experimento *Green-House* foi executado durante 159 dias, no período entre *15 de dezembro de 2005* e *22 de maio de 2006*. Este experimento foi dividido em duas etapas: o período *pré-armadilha* e o período *com-armadilha*.

O período *pré-armadilha* do experimento *Green-House* compreende o tempo desde o início do experimento - *15 de dezembro de 2005* - até o dia *5 de abril de 2006*, totalizando 112 dias. O período *com-armadilha* compreende o período que se inicializa após a instalação de uma armadilha - *dia 06 abril de 2006* - e finaliza com o término dos experimentos, no dia *22 maio de 2006*, totalizando 47 dias.

O critério utilizado para instalação da armadilha no experimento real foi baseado na taxa de picadas. Quando o ser humano entra na estufa e contabiliza uma taxa de picadas igual ou superior a cinco picadas por minuto durante três dias consecutivos, isto indica que a população de mosquitos é “alta”, então é instalada a armadilha para controlar a população de mosquitos. As taxas de picadas nos dias *3, 4 e 5 de abril de 2006* foram respectivamente de 12, 11 e 15 picadas por minuto. A armadilha para capturar mosquitos

foi instalada no dia 06 de abril de 2006.

5.2.2 Simulações computacionais

A apresentação das simulações computacionais do experimento *Green-House* foi dividida em duas partes. A primeira parte apresenta apenas o período *pré-armadilha*, e a segunda parte apresenta o período *com-armadilha* do experimento. Os dados do experimento real referentes a estes períodos de simulação foram adotados para ajuste de parâmetros e validação do modelo de simulação.

Os resultados referentes às simulações computacionais indicados nos gráficos desta seção são valores *médios* obtidos a partir de cinco diferentes simulações. As diferenças entre estas simulações são ocasionadas pelas *sementes* utilizadas na geração de números aleatórios. Estas *sementes* são adotadas nas transições probabilísticas do modelo proposto. Os principais dados comparados entre o experimento da *Green-House* e a simulação computacional foram: produção de ovos, produção de adultos ⁵ e taxa de picadas.

A figura 5.8 ilustra a configuração do cenário de simulação da *Green-House*

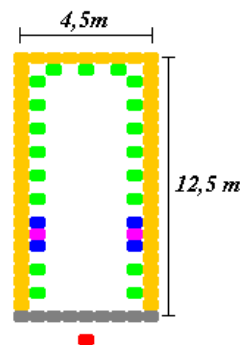


Figura 5.8: Green-House: O cenário de experimentos virtuais

Na figura 5.8 os pontos de cor amarela representam as paredes do cenário de simulação. Os pontos de cor verde representam a vegetação espalhada pelo cenário. Os pontos de cor rosa representam os agentes mosquitos *Aedes aegypti* fêmeas. Inicialmente definiu-se 25 agentes de cada lado, totalizando 50 mosquitos fêmeas. A junção dos pontos de cor cinza representa a porta de acesso à *Green-House*. Os pontos de cor azul representam os reservatórios de água. Definiu-se quatro reservatório de água para este experimento. O ponto vermelho, externo à casa, representa o ser humano.

⁵A produção de adultos medida no experimento da *Green-House* foi medida diretamente através do número de exúvias encontradas em filtros de papeis deixados no ambiente.

As simulações computacionais foram realizadas em computadores do Laboratório de Computação Distribuída da Pontifícia Universidade Católica de Minas (PUC Minas). Em média as simulações da *Green-House* no período *pré-armadilha* foram executadas em 24 horas. Considerando o período total (período *pré-armadilha* e *com-armadilha*) as simulações foram executadas em aproximadamente 30 horas.

5.2.3 Calibração de parâmetros

Os valores dos parâmetros do modelo de simulação foram encontrados na literatura ou foram estimados. Os seguintes parâmetros foram ajustados:

1. *Duração das fases dos mosquitos Aedes aegypti*: os valores adotados para as duração das fases estão descritos na seção 3.1.2.
2. *Sucesso de Picadas*: 20%.
3. *Risco de morte na tentativa de picada*: 2%.
4. *Risco de morte durante a mudança de fase*: 30%.
5. *Duração da alimentação à base de plantas*: 1200 s.
6. *Duração da alimentação animal*: 900 s.
7. *Duração da reprodução - ato sexual*: valores entre 300 e 600 s.
8. *Quantidade máxima de sangue ingerido por picada*: 90%.
9. *Velocidade de oviposição*: 0,02 ovos / s.
10. *Nível máximo de energia*: 155520 unidades de energias.
11. *Nível de energia alto, médio e baixo*: 155520, 77760 e 0 unidades de energias, respectivamente.
12. *Nível de intensidade de rastro alto, médio e baixo*: 8000, 6000 e 0 unidades de rastros, respectivamente.

Estes valores foram obtidos através da versão “manual” do método de otimização de coordenadas descendentes [Luenberger, 1989]. O procedimento de aplicação deste método consistiu em fixar alguns valores de parâmetros e variar outros dentro de uma região de soluções viáveis, seguindo uma direção apropriada.

O parâmetro *Nível máximo de energia* foi um dos principais parâmetros calibrados neste trabalho. Este parâmetro determina o máximo de energia que um mosquito pode ter. O valor calculado para este parâmetro foi 155520 unidades de energia e foi estabelecido a partir do seguinte cálculo:

$$max_e = 86400 \times ndssa \times pece_d$$

Nesta equação a máxima energia que um mosquito por ter (max_e) é calculada considerando o número de dias que o mosquito sobrevive sem alimentação ($ndssa$) e o percentual efetivo de consumo de energia por dia ($pece_d$). O valor é calculado em função do número de segundos em um dia (86400). Os valores adotados para $ndssa$ e max_e foram respectivamente 3 e 60%.

Estes parâmetros foram adotados considerando que um mosquito consome em média uma unidade de energia por segundo em atividade e permanece 60% do seu tempo em atividade. Considerou-se que o mosquito sobrevive até três dias sem alimentação [Templin, 2000].

A fase de calibração do modelo usou apenas dados do período *pré-armadilha*. Esta escolha é relacionada ao fato de que este é o maior intervalo de tempo em que as condições do experimento “real” permanecem constantes.

5.2.4 Validação do modelo

A validação do modelo de simulação proposto consiste na análise quantitativa e qualitativa dos dados obtidos.

Período pré-armadilha - Avaliação geral

Os resultados dos experimentos realizados no período *pré-armadilha* são sintetizados na tabela 5.1. Esta tabela 5.1 indica a correlação entre os dados obtidos do experimento real e o dados das simulações realizadas.

Os resultados apresentados na tabela 5.1 são divididos em três partes: correlação das produções de ovos (A), correlação das produções de mosquitos adultos (B) e correlação das taxas de picadas (C). Alguns dos conceitos adotados nos itens desta tabela são descritos a seguir:

1. *Produção diária* - A produção diária indica quantos ovos/adultos foram produzidos em um dia.

Id.	Dados comparados	Coefficiente de correlação
A	Correlação das produções de ovos	
A.1	Produção diária de ovos	0,2092
A.2	Produção diária de ovos - Dados totalizados	0,3758
A.3	Produção diária acumulada de ovos	0,9779
A.4	Produção diária acumulada de ovos - Dados totalizados	0,9801
A.5	Produção semanal de ovos - Janela fixa	0,6548
A.6	Produção semanal de ovos - Janela deslizante	0,6337
A.7	Produção quinzenal de ovos - Janela fixa	0,7475
A.8	Produção quinzenal de ovos - Janela deslizante	0,8041
B	Correlação das produções de adultos	
B.1	Produção diária de adultos	0,5142
B.2	Produção diária de adultos - Dados totalizados	0,5162
B.3	Produção diária acumulada de adultos	0,9816
B.4	Produção diária acumulada de adultos - Dados totalizados	0,9777
B.5	Produção semanal de adultos - Janela fixa	0,6021
B.6	Produção semanal de adultos - Janela deslizante	0,5669
B.7	Produção quinzenal de adultos - Janela fixa	0,5483
B.8	Produção quinzenal de adultos - Janela deslizante	0,7043
C	Correlação das taxas de picadas	
C.1	Taxa de picadas	0,3129
C.2	Taxa de picadas (até 10 dias antes da instalação armadilha)	0,7731

Tabela 5.1: Análise de correlação no período *pré-armadilha*

2. *Produção diária acumulada* - A produção diária acumulada indica quantos ovos/adultos foram produzidos até um determinado dia.
3. *Dados totalizados* - Os valores coletados no experimento real da *Green-House* não foram coletados respeitando uma frequência uniforme (ex. dia-a-dia ou semana-a-semana), enquanto os dados das simulações da *Green-House* foram coletados dia-a-dia. Esta diferença dos instantes e frequências de coleta dos dados causa variações no comportamento das análises comparativas. Uma maneira de tratar esta diferença é totalizar o número de ovos produzidos na simulação computacional em função dos dias que o ser humano entrou na estufa para coletar ovos.
4. *Produção com janela Fixa* - A janela fixa serve como um totalizador parcial de produções. Por exemplo, considerando um período de 15 dias de experimento e uma janela fixa de tamanho igual a três dias, serão totalizadas as produções de ovos a cada três dias. Assim, cinco totalizações indicarão o comportamento das produções analisadas. Nas comparações realizadas foram utilizadas janelas fixas de tamanho 7 e 15 dias para as análises semanais e quinzenais, respectivamente.
5. *Produção com janela Deslizante* - O conceito de janela deslizante aplicado no contexto deste trabalho refere-se à soma dos números obtidos nos experimentos em um período de n dias, sendo n o tamanho da janela desejada. Para cada dia de experimento a janela “desliza uma posição” e soma os n elementos anteriores. Nas comparações realizadas foram utilizadas janelas deslizantes de 7 e 15 dias para as análises semanais e quinzenais, respectivamente.
6. *Taxa de picadas* - A taxa de picadas corresponde ao número de picadas que o indivíduo sofre durante um minuto.

Na tabela 5.1 as produções diárias e acumuladas de ovos e adultos são apresentadas em dois formatos: dados não totalizados (comparações de identificação A.1, A.3, B.1 e B.3) e dados totalizados (comparações de identificação A.2, A.4, B.2 e B.4).

Analisando os coeficientes de correlação das produções diárias de ovos (comparações de identificação A.1, A.2), conclui-se que sob esta perspectiva os resultados de simulação não foram semelhantes em relação aos dados reais. O maior coeficiente de correlação para este tipo de análise foi de aproximadamente 37%. Entretanto, observando os dados acumulados (comparações de identificação A.3, A.4) observa-se que os resultados de simulação são próximos aos resultados do experimento real. Obteve-se mais de 97% de similaridade nesta análise. Outro resultado considerável é a comparação entre as produções de ovos

sob uma perspectiva quinzenal com janela deslizante (comparação de identificação A.8). Nesta comparação obteve-se mais de 80% de similaridade entre os resultados.

Analisando os coeficiente de correlação das produções diárias de mosquitos na fase adulta (comparações de identificação B.1, B.2), conclui-se que sob esta perspectiva os resultados de simulação são razoavelmente semelhantes em relação aos dados reais. O coeficiente de correlação para este tipo de análise foi de aproximadamente 51%. Contudo, em uma perspectiva acumulada (comparações de identificação B.3, B.4) observa-se que os resultados de simulação são próximos aos resultados do experimento real. Obteve-se mais de 97% de similaridade nesta análise. Outro resultado é relacionado às comparações entre as produções de mosquitos adultos sob uma perspectiva quinzenal com janela deslizante (comparação de identificação B.8). Nesta comparação obteve-se mais de 70% de similaridade entre os resultados.

As taxas de picadas do experimento real e das simulações computacionais não se comportam de forma similar no período *pré-armadilha*. O coeficiente de correlação é de aproximadamente 31% (comparação de identificação C.1). No entanto, se for analisado o coeficiente de correlação entre os dados reais e os dados de simulação, em um período que vai do início dos experimentos até 10 dias antes da instalação da armadilha na *Green-House*, há um aumento significativo neste coeficiente (comparação de identificação C.2). Considerando esta comparação o coeficiente de correlação é de aproximadamente 77%.

Alguns gráficos que ilustram a evolução dos experimentos são apresentados nas próximas subseções. Alguns gráficos complementares a estes são apresentados no apêndice C - *Resultados complementares*.

Período pré-armadilha - Evolução da produção de ovos

O número de ovos produzidos semanalmente no interior da *Green-House* foi contabilizado pelos pesquisadores responsáveis pelo experimento. Com o objetivo de avaliar a dinâmica da população de mosquitos é realizada uma comparação entre o número de ovos contabilizados no experimento real e o número de ovos existentes na simulação.

A interpretação de dados diários nem sempre é a melhor forma de se chegar a conclusões sobre comportamentos, considerando uma perspectiva de longo prazo. Assim, uma alternativa para este tipo de análise é a avaliação de dados acumulados ao longo do tempo. A figura 5.9 apresenta o gráfico com as curvas relacionadas ao número acumulado de ovos produzidos no decorrer do período de experimento e simulação.

A curva da simulação computacional (azul) se aproximou significativamente da curva dos dados coletados no experimento real da *Green-House* (verde). Após a fase de ajustes de parâmetros, conseguiu-se que ao final deste período *pré-armadilha*, o número acumu-

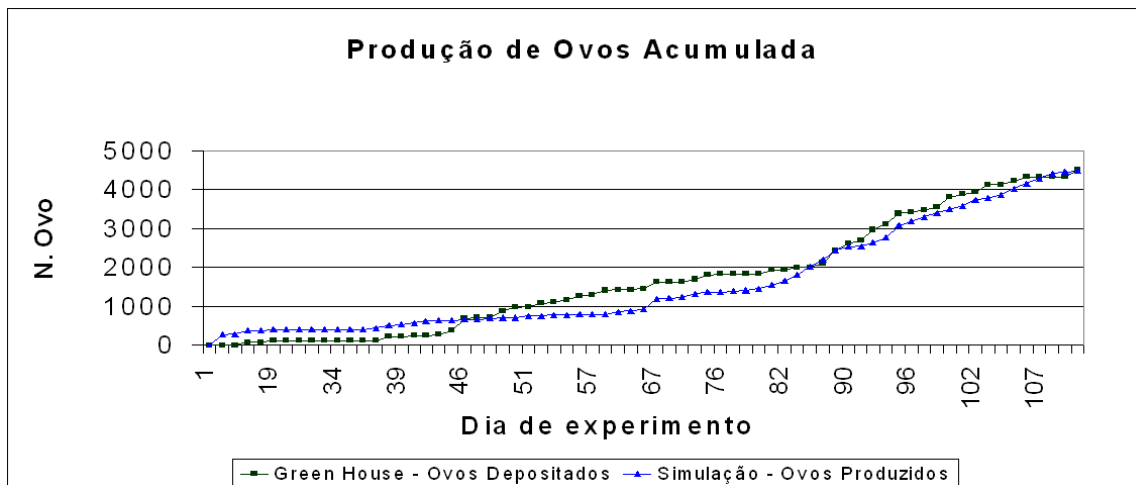


Figura 5.9: Produção acumulada de ovos no período *pré-armadilha*

lado de ovos produzidos fossem muito próximos no experimentos (biológico e computacional). Enquanto no experimento real o número de ovos ao final do período *pré-armadilha* é de 4506 ovos, no experimento computacional este número médio foi de 4494 ovos produzidos. Esta variação corresponde a menos de 1 %.

Outra forma de visualizar a dinâmica da produção de ovos nos experimentos realizados é através da utilização de gráfico com totalizações quinzenais das produções de ovos. A figura 5.10 apresenta um gráfico com produção quinzenal de ovos. Cada ponto do gráfico representa a soma do número de ovos produzidos nos últimos 15 dias. Estes últimos 15 dias compõem uma “janela que desliza” no decorrer do gráfico até o último dia de simulação.

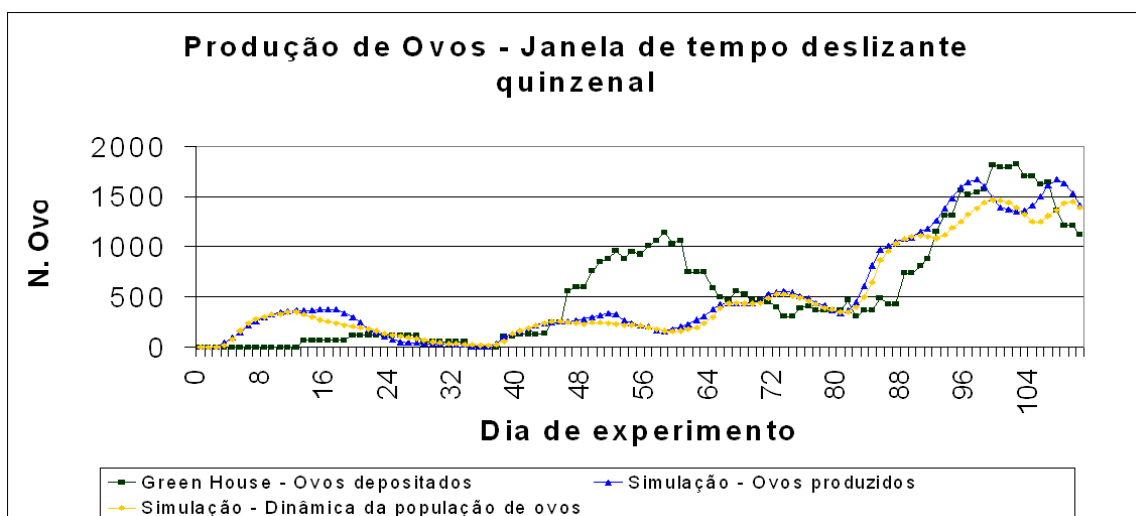


Figura 5.10: Produção quinzenal de ovos no período *pré-armadilha* - Janela deslizante

No gráfico da figura 5.10 a curva verde indica o número de ovos coletados no experimento da *Green-House* e a curva azul indica a produção de ovos nas simulações. Os pontos destas duas curvas representam a soma dos últimos 15 dias. Foi inserida uma curva amarela da dinâmica da população de ovos da simulação. Esta curva ilustra a coerência entre a produção de ovos quinzenal (janela deslizante) do sistema e a dinâmica dia-a-dia da produção de ovos.

Neste gráfico é possível observar uma diferença considerável entre a curva do experimento real (verde) e a curva da simulação computacional (azul), entre o 45° e o 65° dia de experimento. Esta diferença pode ter ocorrido por motivos relacionados a detalhes do experimento real que não são considerados no modelo de simulação proposto. A principal hipótese está relacionada ao modelo de distribuição temporal de ovos adotado, ou seja, o modelo adotado talvez seja mais simples e determinístico do que o modelo real de distribuição de ovos dos mosquitos *Aedes aegypti*. Contudo, mesmo existindo esta variação entre as curvas, elas apresentam comportamentos similares.

Período pré-armadilha - Evolução da produção de adultos

O estudo da evolução da produção de mosquitos adultos é importante. Na vida real o controle de infestação é feito a partir desta informação. O número de exúvias coletadas no interior da *Green-House* é comparado, através de alguns gráficos, com o número de adultos produzidos nas simulações computacionais.

Da mesma forma, que foi apresentada anteriormente uma avaliação acumulativa da produção de ovos nos experimentos, o gráfico da figura 5.11 apresenta a coleta de exúvias acumulada no tempo (curva verde) e a produção acumulada de adultos do experimento computacional (curva azul).



Figura 5.11: Produção acumulada de adultos no período pré-armadilha

As curvas do gráfico da figura 5.11 são similares. O início da produção efetiva de ovos em ambas as curvas ocorre aproximadamente no 29º dia. Um outro fator que indica coerência entre os resultados obtidos está no número acumulado de exúvias e adultos produzidos ao final do experimento. No último dia deste período *pré-armadilha*, o número acumulado de exúvias foi 526, enquanto o número médio acumulado de ovos produzidos foi 532. Embora o comportamento das curvas seja diferente em alguns intervalos, ao final do experimento (visão macro) os números acumulados são próximos.

A fim de complementar análise de produção adultos/exúvias, na figura 5.12 é apresentado gráfico com totalizações quinzenais das produções de adultos/exúvias. Cada ponto do gráfico representa a soma do número de adultos/exúvias produzidos/coletados nos últimos 15 dias. Ou seja, tem-se uma janela de tempo de 15 dias que “desliza” ao longo da realização do experimento.

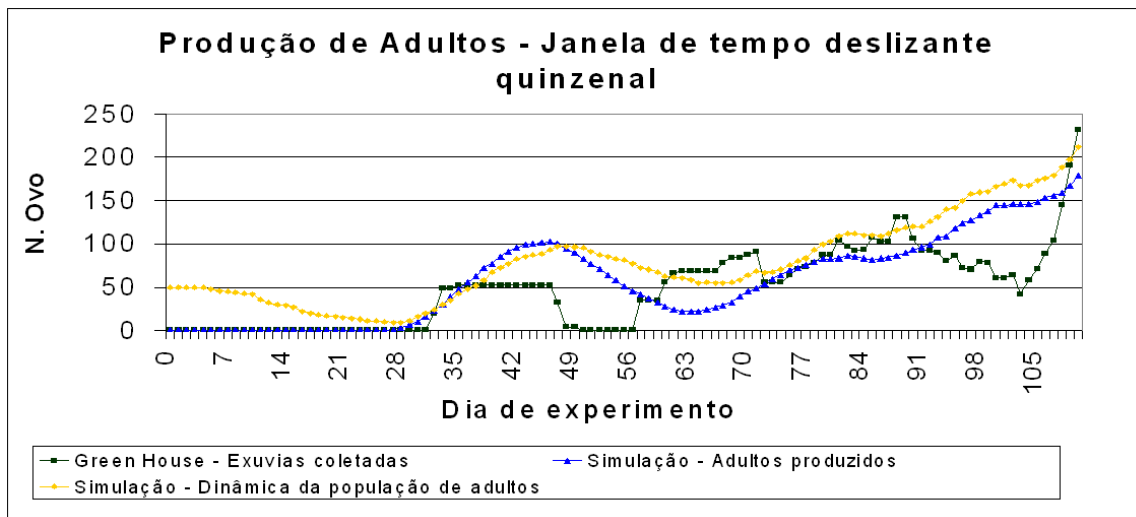


Figura 5.12: Produção quinzenal de adultos no período *pré-armadilha* - Janela deslizante

No gráfico da figura 5.12 a curva verde indica o número de exúvias coletados no experimento da *Green-House* e a curva azul indica a produção de adultos nas simulações. Os pontos destas duas curvas representam a soma dos últimos 15 dias. Foi inserida uma curva amarela da dinâmica da população de adultos da simulação. A curva amarela ilustra a coerência entre a produção de adultos quinzenal (janela deslizante) do sistema e a dinâmica dia-a-dia da produção de adultos. Observa-se que neste gráfico os valores obtidos sempre permanecem entre 0 e 250 adultos/exúvias produzidos.

Estudo de caso - Período *com-armadilha* - Avaliação geral

Para complementar a etapa de validação do modelo de simulação alguns experimentos foram realizados considerando a instalação de uma armadilha no centro da estufa. A armadilha emite um rastro de odor característico de um mamífero. A fêmea do mosquito *Aedes aegypti* é atraída e capturada.

Após a instalação da armadilha na *Green-House* o aluno voluntário entrou algumas vezes na estufa. Neste período os mosquitos atacaram menos o aluno, devido a população de mosquitos ter diminuído com a instalação desta armadilha.

Os resultados dos experimentos realizados no período *com-armadilha* são sintetizados na tabela 5.2. Esta tabela indica a correlação entre os dados obtidos do experimento real e o dados das simulações realizadas, considerando o período posterior à instalação da armadilha.

Id.	Dados comparados	Coefficiente de correlação
A	Correlação das produções de ovos	
A.1	Produção diária acumulada de ovos	0,9478
A.2	Produção quinzenal de ovos - Janela deslizante	0,9515
B	Correlação das produções de adultos	
B.1	Produção diária acumulada de adultos	0,9900
B.2	Produção quinzenal de adultos - Janela deslizante	0,7211

Tabela 5.2: Análise de correlação no período *com-armadilha*

Os resultados apresentados na tabela 5.2 são divididos em duas partes: correlação das produções de ovos (A) e correlação das produções de mosquitos adultos (B). A partir destes coeficientes de correlação, observa-se elevada similaridade entre os experimentos reais e o experimento da *Green-House* no período pós-armadilha.

Além da análise quantitativa das simulações, alguns aspectos qualitativos podem ser observados. Diferentes disponibilidades de alimentação sanguínea foram consideradas nas simulações durante o período *com-armadilha*. As simulações foram divididas em dois grupos: o primeiro grupo considera a alimentação restrita (diminuída), ou seja, permitiu-se que os mosquitos picassem o agente ser humano na mesma proporção do experimento real. No segundo tipo de simulação a disponibilidade de alimentação tornou-se mais abundante. Permitiu-se que o ser humano tomasse até 15 picadas todos os dias.

Considerando estas variações nas simulações, os seguintes experimentos foram realizados e comparados com os dados reais do experimento da *Green-House*:

1. Simulação sem instalação de armadilha e alimentação abundante para os mosquitos.

2. Simulação com instalação de armadilha e alimentação abundante para os mosquitos.
3. Simulação sem instalação de armadilha e alimentação restrita para os mosquitos.
4. Simulação com instalação de armadilha e alimentação restrita para os mosquitos.

Os resultados destes experimentos ilustram a dinâmica da população de mosquitos no cenário simulado diante de restrições de alimentação e da instalação de uma armadilha.

As análises de correlações apresentada na tabela 5.2 consideraram os dados do experimento real da *Green-House* e as simulação com instalação de armadilha e alimentação abundante para os mosquitos.

Estudo de caso - Período com-armadilha - Evolução da produção de ovos

A figura 5.13 ilustra a produção acumulada de ovos no período *com-armadilha*.

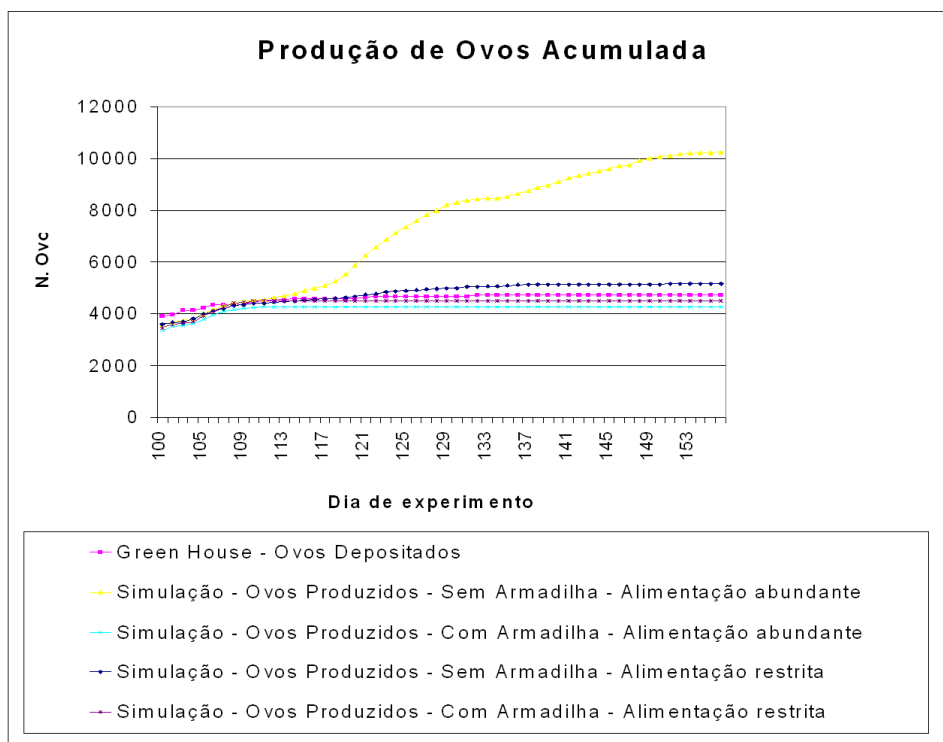


Figura 5.13: Produção acumulada de ovos no período *com-armadilha*

A armadilha foi instalada no 112º dia de experimento. Após este dia o número de ovos produzidos se comporta de forma diferente para as simulações (curvas) realizadas. Com exceção da curva da simulação sem armadilha e com alimentação abundante os demais experimentos mantiveram a mesma tendência que o experimento real. O número de ovos acumulados nas simulações que possuem armadilha instalada se manteve abaixo

do número de ovos coletados no experimento real. As simulações sem a instalação da armadilha produziram um número de ovos acumulados superior ao experimento real.

A figura 5.14 ilustra a produção de ovos no período *com-armadilha* com janela deslizante quinzenal. Cada ponto do gráfico representa a soma do número de ovos produzidos nos últimos 15 dias. Ou seja, tem-se uma janela de tempo de 15 dias que “desliza” ao longo da realização do experimento.

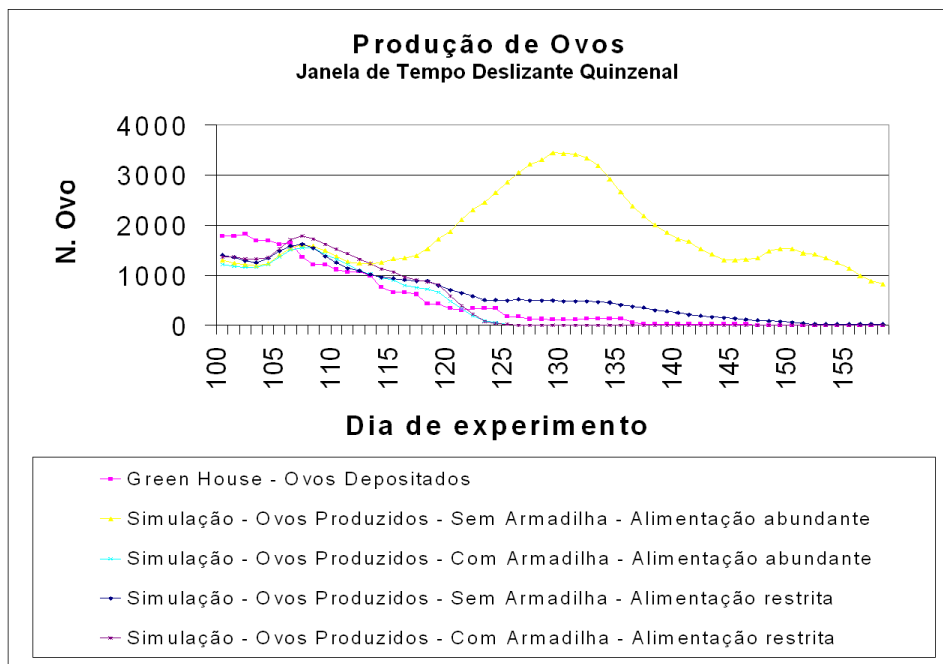


Figura 5.14: Produção quinzenal de ovos no período *com-armadilha* - Janela deslizante

Nota-se na figura 5.14 que após o 112º dia do experimento a produção de ovos diminuiu no experimento real. Nas simulações que possuem armadilha esta queda é acentuada. Na simulação com alimentação restrita e sem armadilha (curva azul) a produção de ovos diminuiu. A alimentação restrita afeta a produção de ovos. Na simulação com alimentação abundante e sem armadilha (curva amarela) o número de ovos produzidos é oscilante, sendo que atinge o número máximo de 3425 ovos no 130º dia de simulação.

Estudo de caso - Período com-armadilha - Evolução da produção de adultos

A figura 5.15 ilustra a produção acumulada de mosquitos adultos no período *com-armadilha*.

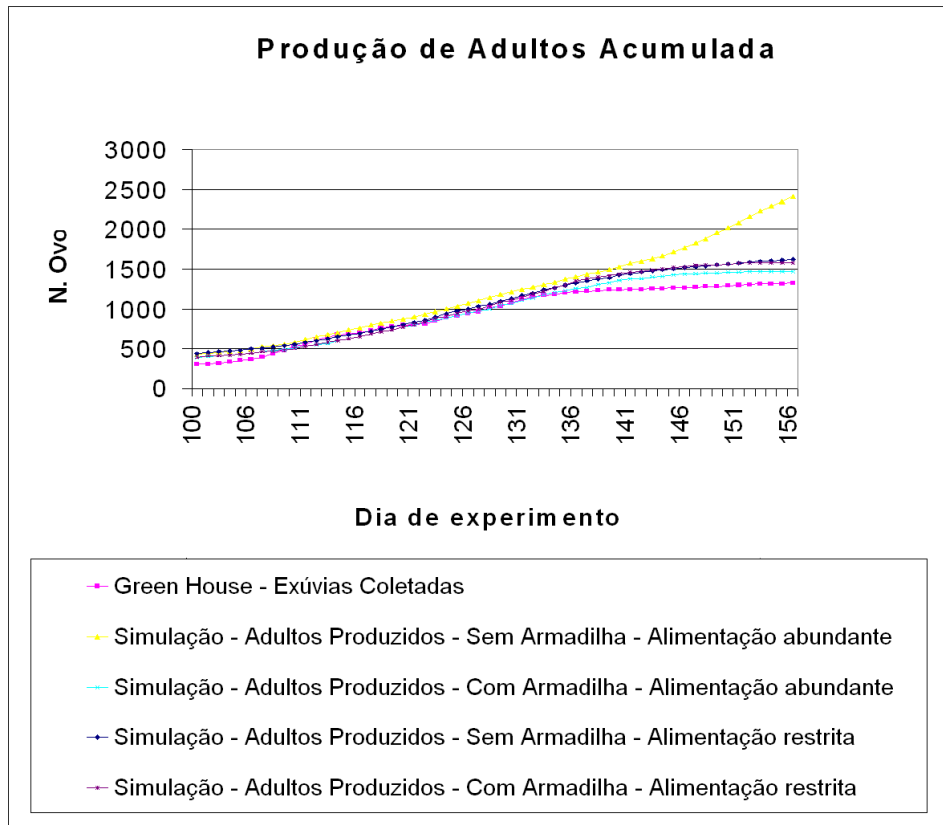


Figura 5.15: Produção acumulada de adultos no período *com-armadilha*

Entre o 112º e o 136º dia do experimento as simulações e o experimento real (curva rosa) mantiveram quase o mesmo valor. Após o 136º dia de experimento o número de mosquitos adultos produzidos nas simulações manteve a mesma tendência do experimento real, com exceção da curva da simulação sem armadilha e com alimentação abundante (curva amarela). O número de adultos produzidos nas simulações sempre se manteve superior ao número de exúvias coletadas no experimento real.

A figura 5.16 ilustra a produção de mosquitos adultos no período *com-armadilha* com janela deslizante quinzenal. Cada ponto do gráfico representa a soma do número de adultos produzidos nos últimos 15 dias. Ou seja, tem-se uma janela de tempo de 15 dias que “desliza” ao longo da realização do experimento.

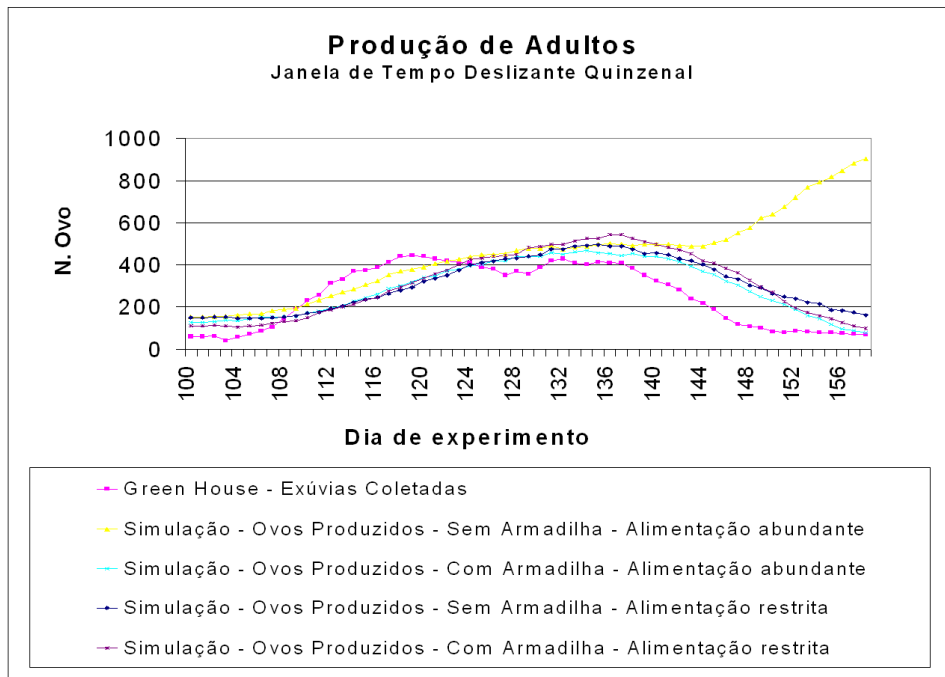


Figura 5.16: Produção quinzenal de adultos no período *com-armadilha* - Janela deslizante

O gráfico da figura 5.16 indica que após o 112º dia do experimento a população de mosquitos adultos continua crescendo e depois de alguns dias inicia tendência de queda. Isto ocorre no experimento real e nas simulações computacionais, com exceção da simulação que considera alimentação abundante para os mosquitos e não instalação de armadilha.

5.3 Estudo de caso - Belo Horizonte

Um terceiro grupo de experimentos realizados consistiu em simular uma região correspondente a um quarteirão urbano. Este quarteirão foi modelado a partir de dados disponibilizados pelo Centro de Controle de Zoonoses da Prefeitura de Belo Horizonte - MG. Outros dados auxiliaram a modelagem, como imagens aéreas obtidas através do software *Google Earth* [Google Inc., 2007] e dados meteorológicos obtidos do *Centro de Previsão de Tempo e Estudos Climáticos do Instituto Nacional de Pesquisas Espaciais - CPTEC/INPE*.

O objetivo deste experimento é verificar a consistência do modelo de simulação em cenários mais complexos. Os dados obtidos nas simulações são comparados com os dados disponibilizados pelo centro de controle de zoonoses.

5.3.1 Cenário real

O quarteirão simulado, denominado 3545, localiza-se no bairro Vista Alegre, região oeste da cidade de Belo Horizonte. A partir da fotografia aérea da figura 5.17 é possível visualizar o bairro Vista Alegre e o quarteirão modelado (região retangular limitada).



Figura 5.17: Bairro Vista Alegre de Belo Horizonte - MG, Brasil

O Centro de Controle de Zoonoses da Prefeitura de Belo Horizonte forneceu diversos dados referentes à região oeste como: perfil e número de moradores da região; número de residências, comércios e animais em cada quarteirão; número de ovos de mosquitos *Aedes aegypti* capturados a cada 15 dias em ovitrampas (armadilhas) durante os anos de 2004 e 2005. Também foram disponibilizadas as localizações de pontos estratégicos onde foram instaladas as ovitrampas.

5.3.2 Simulações computacionais

O quarteirão 3534 foi modelado e implementado como um cenário do simulador de população de mosquitos. Nesta modelagem foram considerados os agentes: mosquitos *Aedes aegypti* (macho e fêmea), cachorros, gatos e seres humanos. Foram inseridos no cenário de simulação os objetos: vegetação, recipientes de água, ovitrampa, muros, paredes, janelas, portas e portões.

Para realizar as simulações computacionais foi escolhido o período virtual entre o dia 23 de setembro de 2004 e o dia 23 de março de 2005. Neste período diferentes experimentos

computacionais foram realizados. A escolha deste período é relacionada com o grau de detalhes dos dados de controle de ovitrampas disponibilizados pelo centro de controle de zoonoses. Durante este período é maior o número de ovos capturados pelas ovitrampas, pois o período é o mais chuvoso do ano.

O quarteirão simulado foi projetado com as dimensões: 198 *m* de comprimento x 66 *m* de largura. O quarteirão foi dividido em 36 lotes, sendo que cada lote mede 11 x 33 *m* e possui uma “casa”. As casas possuem portas, janelas, dimensões variadas e localizações variadas dentro do lote.

Foram distribuídos 42 reservatórios de água no cenário simulador. Inicialmente foram atribuídos para cada reservatório 10 ovos de mosquito *Aedes aegypti* e volume de água que varia entre 60 e 100% da capacidade dos reservatórios. A profundidade de cada reservatório varia de 0,2 *m* a 1 *m*. Também foram distribuídas plantas nos lotes do cenário. Alguns lotes receberam mais arbustos que outros.

Em cada residência foram inseridos um ou dois seres humanos. Alguns seres humanos deslocam-se aleatoriamente na sua região residencial. Outros saem de casa pela manhã e retornam somente à noite. A proporção de seres humanos que ficam nas residências durante o dia foi definida a partir de estimativas realizadas sobre os dados fornecidos pelo Centro de Controle de Zoonoses (CENSO). Foram inseridos também 26 cachorros e 1 gato na região modelada.

Uma ovitrampa foi inserida no centro do quarteirão. Na próxima seção o número de mosquitos capturados por esta ovitrampa é comparado com dados do centro de controle de zoonoses.

A figura 5.18 ilustra a configuração do quarteirão simulado.

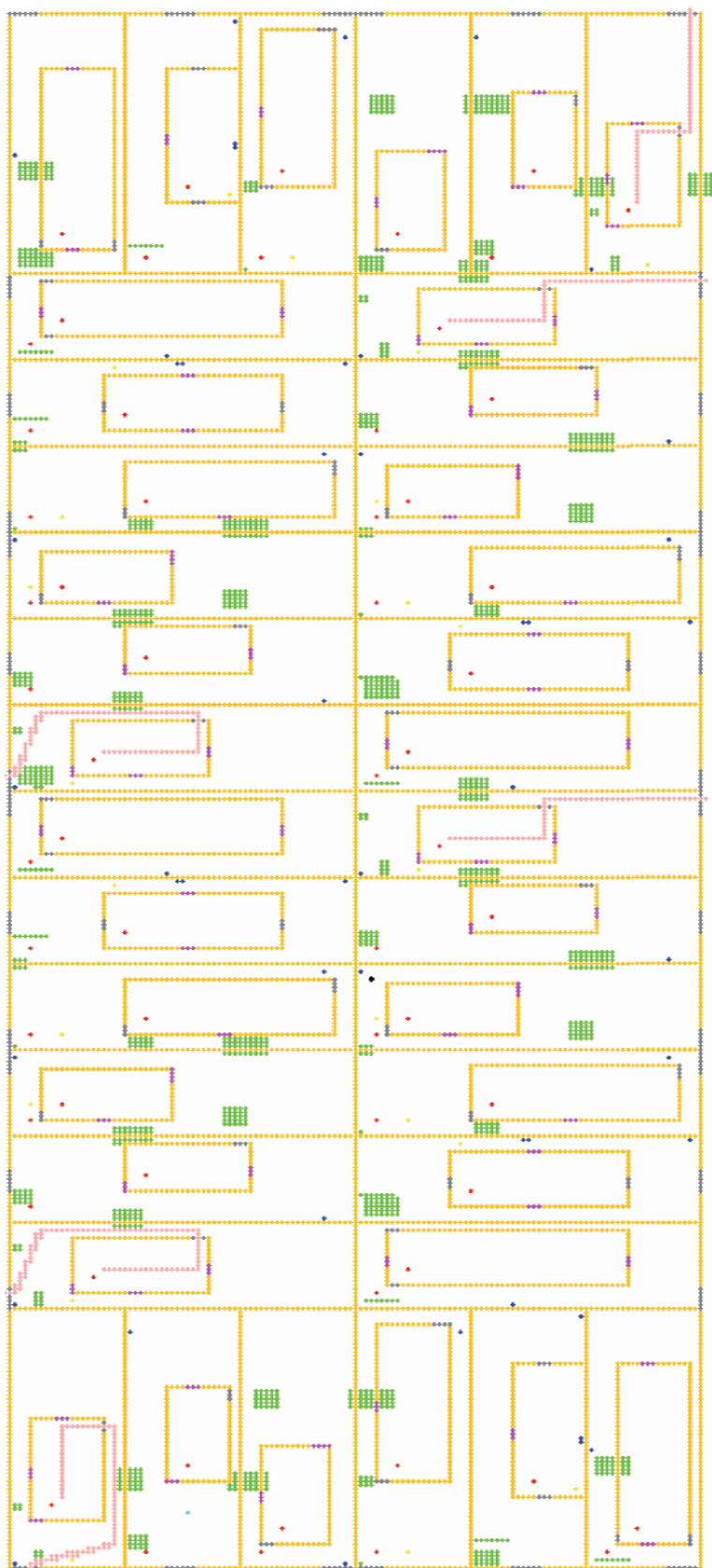


Figura 5.18: Quarteirão 3545 simulado

Na figura 5.18 os pontos de cor amarela representam as paredes e muros do cenário. Os pontos verdes representam as vegetações e a água é representada pelos pontos de cor azul. Pontos amarelos ilustram cachorros e gatos, enquanto os vermelhos representam os seres humanos. As janelas são identificadas pelos pontos de cor rosa e as portas e portões possuem cor cinza. Em alguns lotes existe um caminho de cor rosa por onde alguns seres humanos devem passar. Os mosquitos *Aedes aegypti* foram inicializados dentro dos reservatórios de água, por isto não possuem representação gráfica neste instante inicial. O único ponto de cor preta representa uma ovitrampa posicionada no quarteirão.

5.3.3 Resultados

Diferentes análises foram realizadas considerando a simulação do quarteirão 3545. A figura 5.19 apresenta o número de ovos de mosquitos *Aedes aegypti* capturados na armadilha do experimento do quarteirão 3545.

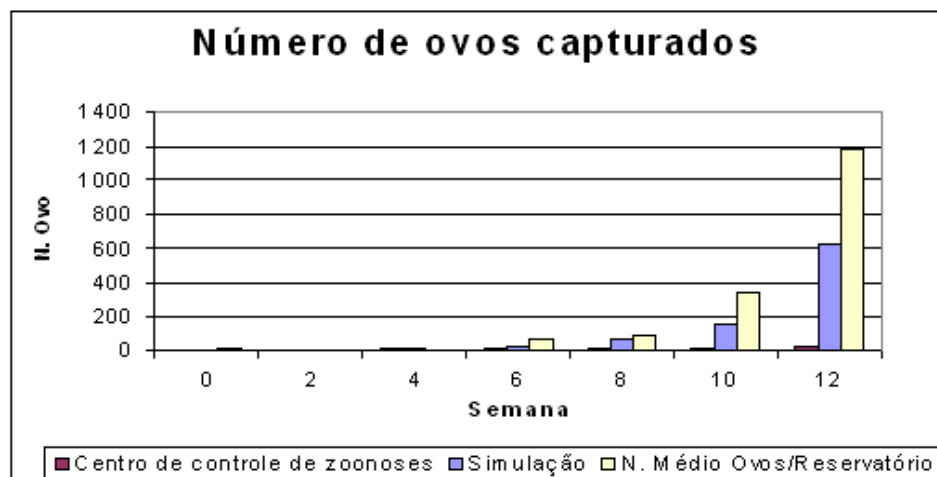


Figura 5.19: Número de ovos capturados

No gráfico da figura 5.19 observa-se que nas quatro primeiras semanas foram capturados poucos ovos na armadilha. Após este período existe uma tendência de crescimento do número de ovos capturados. Os resultados da simulação são relativamente próximos aos resultados do centro de controle de zoonoses até a 6^a semana. Após este período é crescente o número de ovos capturados na armadilha da simulação (barra de cor azul) e o número médio de ovos nos reservatórios da simulação. Esta diferença aumenta até a 12^a semana de simulação.

Esta aparente divergência entre os resultados do centro de controle de zoonoses e das simulações computacionais é explicada a partir da investigação realizada a cerca da dinâmica de distribuição dos ovos no cenário simulado, resumida na figura 5.20.

A figura 5.20 apresenta o número médio de ovos de mosquitos *Aedes aegypti* nos 42 recipientes presentes no cenário simulado. Esta figura apresenta também o desvio padrão relacionado a este número médio de ovos por recipiente.

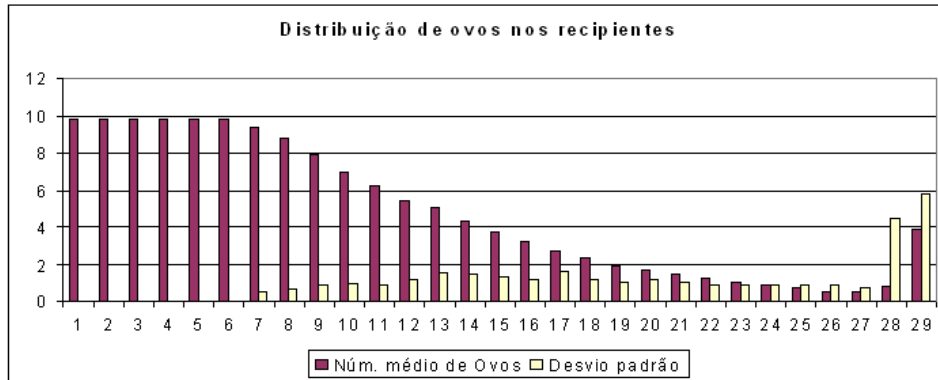


Figura 5.20: Dinâmica da distribuição de ovos na simulação

A figura 5.20 ilustra que com a evolução da simulação o desvio padrão, relacionado ao número médio de ovos por reservatório aumenta. Isto significa que a distribuição dos ovos, que inicialmente era uniforme entre os reservatórios, fica mais heterogênea com o passar do tempo. No 29º dia de simulação o reservatório com maior número de ovos possui 36 ovos, enquanto existem reservatórios sem ovos. O número médio de ovos no 29º dia simulação é de aproximadamente 3,88 ovos.

Este resultado indica que a distribuição de ovos nos reservatórios é heterogênea. Assim, pode-se afirmar que a ovitrampa pode sofrer grande variação no número de ovos contidos em seu interior dependendo de seu posicionamento no ambiente investigado.

Outros resultados obtidos corroboram com esta afirmação. Em algumas simulações foram adicionadas armadilhas em diferentes pontos do cenário simulado. Algumas armadilhas não capturaram ovos de mosquitos *Aedes aegypti*, pois dependendo de sua localização competiram com criadouros naturais (reservatórios de água). Esta afirmação explica a aparente divergência entre os resultados do centro de controle de zoonoses e das simulações computacionais (figura 5.19).

A figura 5.21 relaciona dados normalizados de captura de ovos e chuva no período simulado. Nesta figura observa-se um aumento do volume de chuvas nas quatro primeiras semanas. O número de ovos do mosquito aumenta no período posterior a esta chuva. Existe uma relação entre o aumento do pluvio e o aumento do número de ovos, considerando o período analisado.

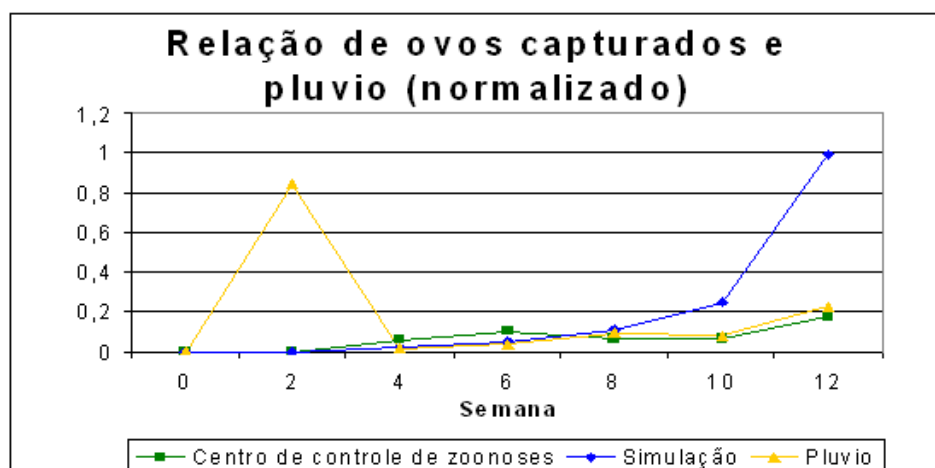


Figura 5.21: Número de ovos capturados na armadilha relacionado com o pluvio

A figura 5.22 ilustra a dinâmica da evolução da população de mosquitos *Aedes aegypti* em diferentes fases. Nesta figura observa-se um número de 420 ovos inicialmente inseridos no cenário de simulação. Após os cinco primeiros dias, o número de ovos diminui, e consequentemente o número larvas aumenta. A partir do 19º dia simulado, as larvas começam a se transformarem em pupas. Após o 22º dia de simulação os primeiros mosquitos eclodem, e em seguida começam a dar origem a novos ovos.

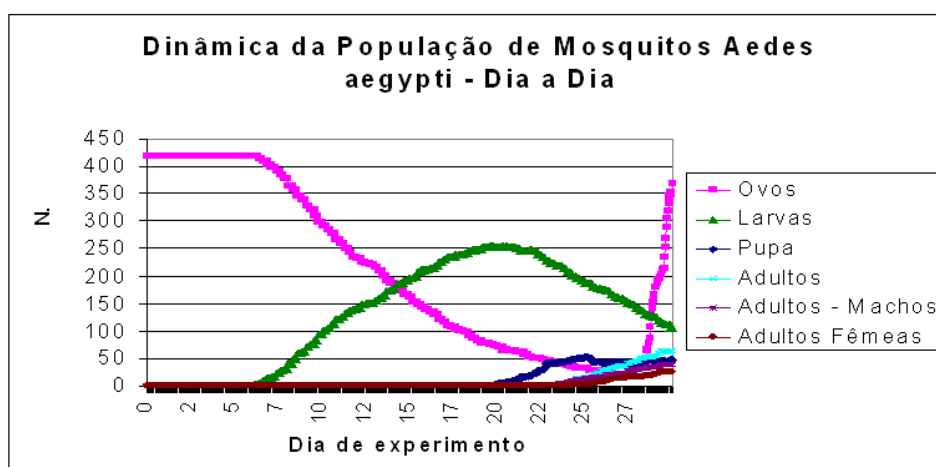


Figura 5.22: Dinâmica da população de mosquito *Aedes aegypti* em diferentes fases

Capítulo 6

Comentários e Conclusões

Considerando o problema de estimar uma população de mosquitos e as limitações dos atuais modelos e ferramentas, o presente trabalho apresenta um simulador computacional que fornece bons resultados. O simulador é composto por um modelo de simulação de população de mosquitos *Aedes aegypti*, denominado *SimPopMosq*, e um *framework* de simulação de sistemas ecológicos, denominado *SIMES*. Este simulador foi descrito conceitualmente e em termos de implementação. Resultados de simulações com o simulador foram analisados e comparados com dados experimentais reais.

O modelo de simulação e o simulador desenvolvido permitem realizar simulações de micro-regiões como, por exemplo, uma casa com seus moradores, mosquitos voando nos limites da residência e objetos da casa. Foi possível reproduzir através de simulações um experimento científico. Nestas simulações permitiu-se observar a dinâmica de grupo de mosquitos diante de restrições estabelecidas, como a falta de água ou sangue para repasto.

Na modelagem conceitual do simulador de população de mosquitos foram definidos e adotados modelos de agentes e objetos e ambiente que fazem parte do simulador. Uma das contribuições do trabalho foi a definição estruturada de um modelo de tomada de decisão para o mosquito *Aedes aegypti* (macho e fêmea). Até a presente data, nenhum modelo de tomada de decisão de mosquitos *Aedes aegypti* similar foi encontrado na literatura.

Foi adotado o protocolo *ODD* (*Overview, Design concepts and Details*) para descrever conceitualmente o modelo de simulação *SimPopMosq*. O protocolo *ODD* foi proposto para especificar modelos baseados em indivíduos. A descrição do modelo *SimPopMosq* poderia ser realizada de diferentes maneiras, como por exemplo através da utilização dos modelos de análise estruturada, essencial ou orientada a objetos. O uso do protocolo *ODD* contribuiu tornando diversos aspectos relacionados com a modelagem dos agentes claros e consistentes. Modelos clássicos, como *UML*, ignoram detalhes intrínsecos dos modelos baseados em indivíduos.

Durante a construção deste trabalho foi desenvolvido o *framework SIMES*. Este *framework* é uma extensão do *framework* de simulações multiagentes *Repast* - versão 3.0. A utilização deste *framework* como base para o desenvolvimento de modelos de simulação induz os desenvolvedores a criarem o código da sua aplicação mais padronizado, e como consequência reutilizável. Outros modelos podem ser desenvolvidos usando o *framework SIMES*.

O modelo de simulação de população de mosquitos *Aedes aegypti SimPopMosq* é elemento essencial para o funcionamento do simulador desenvolvido. Este modelo consiste em um conjunto de classes implementadas na linguagem de programação *Java* sobre os padrões dos *frameworks SIMES e Repast*. O principal benefício obtido a partir do uso do *framework SIMES* é permitir que os modelos de simulação sejam construídos com foco maior na lógica do modelo de simulação e menos foco em requisitos não-funcionais. O código-fonte do *SimPopMosq* possui implementações relacionadas essencialmente com os aspectos de simulação de população de mosquitos *Aedes aegypti* em cenários urbanos. Foram adotados diagramas de classe da *UML* para ilustrar o relacionamento de classes do modelo.

Um repositório com diferentes cenários de simulação foi desenvolvido e incorporado ao modelo de simulação *SimPopMosq*. O usuário do modelo pode alterar condições experimentais e/ou adicionar novos cenários ao repositório. Embora seja possível modelar e simular diversos cenários com os agentes e objetos disponíveis na ferramenta desenvolvida, a modelagem e a simulação de fenômenos relacionados aos mosquitos *Aedes aegypti* foram o foco principal do ambiente proposto.

Os cenários desenvolvidos permitiram a realização de diferentes simulações. Essas simulações serviram para calibrar os parâmetros, validar e aplicar o modelo de simulação desenvolvido. Os seguintes experimentos foram realizados:

i) Experimento qualitativo - Este grupo de experimentos foi feito para ilustrar o funcionamento dos modelos de racionalidade dos agentes *Aedes aegypti* macho e fêmea. Foram simulados alguns mosquitos *Aedes aegypti* em um ambiente fechado, com disponibilidade de alimento, criadouro (água) e parceiro para acasalamento. A partir deste experimento verificou-se de forma qualitativa, a coerência entre o modelo computacional proposto (machos e fêmeas *Aedes aegypti*) e os dados informados sobre o comportamento e características do mosquito na literatura.

ii) Experimento da *Green-House* - Foram realizadas simulações cujo cenário e condições iniciais são similares às de um experimento real, denominado *Green-House*. As simulações foram realizadas para calibrar e validar o modelo de simulação *SimPopMosq*. Os resultados obtidos das simulações foram comparados aos resultados do experimento da

Green-House. A correlação entre estes resultados se mostrou satisfatória, atingindo um percentual superior a 90% de similaridade nas principais variáveis analisadas.

O estudo de caso que tratava do período com-armadilha evidenciou que uma armadilha que capture fêmeas grávidas é eficiente no controle da população de mosquitos. O estudo de caso também indicou que caso seja dificultado o consumo de sangue pelas fêmeas o resultado é o controle da produção de ovos e conseqüentemente o controle da população.

iii) Experimento do *Quarteirão* - Foram realizadas simulações de um quarteirão, localizado na região oeste de Belo Horizonte - MG. Os resultados obtidos a partir desta simulação indicam que o modelo de simulação é coerente com dados fornecidos pelo Centro de Controle de Zoonoses da prefeitura de Belo Horizonte. Considerando o cenário simulado, mostrou-se que a distribuição de ovos do mosquito *Aedes aegypti* em reservatórios de um ambiente é heterogênea e que o número de ovos capturados em armadilhas nas simulações possui relação com o volume de chuva.

Uma das maiores dificuldades encontradas no desenvolvimento deste trabalho é relacionada ao ajuste de parâmetros. Muitos dos parâmetros do modelo de simulação não foram encontrados em pesquisas bibliográficas. A partir desta limitação, alguns parâmetros foram estimados, enquanto outros foram determinados após um conjunto de experimentos direcionados pelo método de otimização denominado coordenadas descendentes.

6.1 Limitações do trabalho

As principais limitações do trabalho desenvolvido são relacionadas com os seguintes itens:

- O desempenho computacional do simulador é vinculado ao número de elementos a serem simulados. Quanto maior o número de elementos, mais lenta é a simulação.
- Os cenários de simulação são desenvolvidos diretamente na linguagem de programação *Java*. Para criar, alterar ou excluir cenários de simulação, o usuário do simulador deverá desenvolver o cenário desejado, adicionando os objetos relacionados às classes dos elementos do modelo *SimPopMosq*. Um conhecimento básico do paradigma de programação orientada a objetos e da linguagem de programação *Java* é necessário.
- Os resultados do modelo de simulação proposto não foram comparados com os resultados de outros modelos descritos na literatura, como por exemplo, modelos baseados em equações diferenciais e modelos baseados em eventos discretos [Focks et al., 1993a]. Por outro lado, os modelos da literatura não foram projetados para tratar os problemas abordados neste trabalho.

6.2 Trabalhos futuros

Considerando as limitações deste trabalho e as sugestões recebidas, os seguintes trabalhos podem ser desenvolvidos futuramente:

- Permitir a configuração de cenários de simulação de forma gráfica e em tempo de execução. O usuário poderia criar, salvar, alterar e excluir cenários de simulação sem se preocupar em interagir com o código-fonte da aplicação.
- Aprimorar o mecanismo de difusão de rastro, considerando aspectos particulares de modelos de difusão de som e modelos visuais. O atual mecanismo de rastro é baseado em um modelo de difusão de substâncias.
- Incorporar ao simulador um modelo de propagação da dengue, através dos mosquitos *Aedes aegypti* e mamíferos desenvolvido no modelo *SimPopMosq*. Com esse recurso, simulações poderiam ser realizadas e dados referentes à dinâmica de contaminação da população de pessoas poderiam ser estudados. Neste caso, o simulador poderia servir como um “laboratório virtual” para responder perguntas relacionadas à dengue.
- Tornar público o projeto do simulador através da comunidade virtual *SourceForge.net*.
- Aprimorar o desempenho computacional estudando questões relacionadas à paralelização e distribuição de processamento.
- Incorporar ao *framework SIMES* um modelo de calibração de parâmetros baseado em algoritmos de otimização. O usuário do *framework* poderia especificar um conjunto de parâmetros a serem calibrados e o simulador automaticamente poderia realizar as simulações, avaliar as saídas e alterar os parâmetros conforme necessário. Este procedimento poderia ser realizado iterativamente, até que uma solução desejada seja alcançada.
- Comparar os resultados obtidos em simulações com os resultados de modelos baseado em equações diferenciais propostos na literatura.
- Ampliar os experimentos do quarteirão urbano para áreas mais extensas, e com isto, estudar diferentes estratégias de distribuição de armadilhas na região simulada. Os resultados destes experimentos podem auxiliar o centro de controle de zoonoses da Prefeitura de Belo Horizonte - MG a posicionarem suas armadilhas em locais mais apropriados. Poder-se-ia estudar também o efeito de outros tipos de armadilhas nestes cenários complexos, como o uso da armadilha de captura de insetos adultos.

Bibliografia

- [Alves et al., 2003] Alves, D., Haas, V. J., and Caliri, A. (2003). The Predictive Power of R_0 in an Epidemic Probabilistic System. *Journal of Biological Physics*, 29(1):63–75.
- [Amor et al., 2005] Amor, M., Fuentes, L., and Vallecillo, A. (2005). *Agent-Oriented Software Engineering V*, volume 3382, chapter Bridging the Gap Between Agent-Oriented Design and Implementation Using MDA, pages 93–10. Springer Berlin / Heidelberg.
- [Bailey, 1975] Bailey, N. T. J. (1975). *The Mathematical Theory of Infectious Diseases and its Applications*. Hafner Press, 2 edition.
- [Bar-Zeev, 1957] Bar-Zeev, M. (1957). The effect of density on the larvae of a mosquito and its influence on fecundity. *Bull. Res. Council Israel*, 6(B):220–228.
- [Barrett et al., 2005] Barrett, C. L., Eubank, S. G., and Smith, J. P. (2005). IF SMALL-POX STRIKES PORTLAND... EPISIMS unleashes virtual plagues in real cities to see how social networks spread disease. that knowledge might help stop epidemics. *Scientific American*.
- [Bauer et al., 2001] Bauer, B., Müller, J. P., and Odell, J. (2001). Agent uml: a formalism for specifying multiagent software systems. In *First international workshop, AOSE 2000 on Agent-oriented software engineering*, pages 91–103, Secaucus, NJ, USA. Springer-Verlag New York, Inc.
- [Bellifemine et al., 1999] Bellifemine, F., Poggi, A., and Rimassa, G. (1999). JADE — A FIPA-compliant agent framework. In *Proceedings of the 4th International Conference on the Practical Applications of Agents and Multi-Agent Systems (PAAM-99)*, pages 97–108, London, UK. The Practical Application Company Ltd.
- [Bertsekas, 1988] Bertsekas, D. P. (1988). The auction algorithm: A distributed relaxation method for the assignment problem. *Annals of Operations Research*, 14(“LIDS Report P-1653”):105–123.

- [Bertsekas, 1991] Bertsekas, D. P. (1991). An auction algorithm for shortest paths. *j-SIAM-J-OPT*, 1(4):425–447.
- [Bertsekas, 1992] Bertsekas, D. P. (1992). Auction algorithms. *Annals of Operations Research*, 1:23–31.
- [Bertsekas and Catañon, 1989] Bertsekas, D. P. and Catañon, D. A. (1989). The auction algorithm for transportation problems. *Annals of Operations Research*, 20:67–96.
- [Bertsekas and Catañon, 1991] Bertsekas, D. P. and Catañon, D. A. (1991). A generic auction algorithm for the minimum cost network flow problem. *Alphatech Report*, 1:7–66.
- [Bonabeau, 2002] Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99:7280–7287.
- [Bresenham, 1965] Bresenham, J. E. (1965). Algorithm for computer control of a digital plotter. *IBMSJ*, 4:25–30.
- [Change Vision Inc, 2007] Change Vision Inc (2007). UML Modeling Tool - JUDE. <http://jude.change-vision.com> . Acesso em: 06 jul. 2007.
- [Chung, 2003] Chung, C. A. (2003). *Simulation Modeling Handbook: A Practical Approach*. CRC Press.
- [Collier, 2000] Collier, N. (2000). Repast: An extensible framework for agent simulation. Technical report, Social Science Research Computing, University of Chicago, Chicago, Illinois.
- [Consoli and Oliveira, 1994] Consoli, R. A. G. B. and Oliveira, R. L. (1994). *Principais Mosquitos de Importância Sanitária no Brasil*. Editora Fiocruz, Rio de Janeiro.
- [CPTEC-INPE, 2007] CPTEC-INPE (2007). Centro de previsão de tempo e estudos climáticos. <http://www.cptec.inpe.br/> . Acesso em: 23 ago. 2007.
- [da Silva and de Lucena, 2003] da Silva, V. T. and de Lucena, C. J. P. (2003). Mas-ml: a multi-agent system modeling language. In *OOPSLA '03: Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 126–127, New York, NY, USA. ACM Press.

- [Depinay et al., 2004] Depinay, J.-M. O., Mbogo, C. M., Killeen, G., Knols, B., Beier, J., Carlson, J., Dushoff, J., Billingsley, P., Mwambi, H., Githure, J., Toure, A. M., and McKenzie, F. E. (2004). A simulation model of african anopheles ecology and population dynamics for the analysis of malaria transmission. *Malaria Journal*, 3(29).
- [Derouich et al., 2003] Derouich, M., Boutayeb, A., and Twizell, E. (2003). A model of dengue fever. *BioMedical Engineering OnLine*.
- [Eiras, 2000] Eiras, A. E. (2000). *Parasitologia Humana*, chapter Culicídeos, pages 355–367. Atheneu.
- [Faraco, 1998] Faraco, R. A. (1998). Uma arquitetura de agentes para negociação no domínio do comércio eletrônico. Master’s thesis, Universidade Federal de Santa Catarina.
- [Fernandes, 2005] Fernandes, A. M. R. (2005). *Inteligência artificial: noções gerais*. Visual Books, Florianópolis - SC, 1 edition.
- [Ferreira and Yang, 2003] Ferreira, C. P. and Yang, H. M. (2003). Estudo dinâmico da população de mosquitos *Aedes aegypti*. *CNMAC - Congresso Nacional de Matemática Aplicada e Computacional*, 4(2):187–196.
- [FIPA, 1996] FIPA (1996). Foundation for intelligent physical agents. <http://fipa.org>. Acesso em: 15 ago. 2007.
- [Flake, 2000] Flake, G. W. (2000). *The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems and Adaptation*. The MIT Press, Cambridge, MA.
- [Focks et al., 1995] Focks, D. A., Daniels, E., Haile, D. G., and Keesling, J. (1995). A simulation model of the epidemiology of urban dengue fever: Literature analysis, model development, preliminary validation, and samples of simulation results. *American Journal of Tropical Medicine and Hygiene*, 53(6):489–506.
- [Focks et al., 1993a] Focks, D. A., Haile, D. G., Daniels, E., and Mount, G. (1993a). Dynamic Life Table for *Aedes aegypti* (Diptera: Culicidae): Simulation Results and Validation. *Journal of Medical Entomology*, 30(6):1018–1028.
- [Focks et al., 1993b] Focks, D. A., Haile, D. G., Daniels, E., and Mount, G. A. (1993b). Dynamic Life Table for *Aedes aegypti* (Diptera: Culicidae): Analysis of the Literature and Model Development. *Journal of Medical Entomology*, 30(6):1003–1017.

- [Franklin and Graesser, 1996] Franklin, S. and Graesser, A. (1996). Is this an agent or just a program? a taxonomy for autonomous agents. In *Proceedings of the third international workshop on agents, theories, architectures, and languages*. Springer-Verlag, New York.
- [Gomes, 1998] Gomes, A. C. (1998). Medidas dos níveis de infestação urbana para *Aedes (Stegomyia) aegypti* e *Aedes (Stegomyia) albopictus* em Programa de Vigilância Entomológica. *IESUS - Informe Epidemiológico do SUS*, 7(3):49–57.
- [Gomes et al., 2006] Gomes, A. S., Sciavico, C. J. S., and Eiras, A. E. (2006). Periodicidade de oviposição de fêmeas de *Aedes aegypti* (Linnaeus, 1762) (Diptera: Culicidae) em laboratório e campo. *Revista da Sociedade Brasileira de Medicina Tropical*, 39(4):327–332.
- [Google Inc., 2007] Google Inc. (2007). Google earth. <http://earth.google.com/intl/pt/> . Acesso em: 23 ago. 2007.
- [Grimm, 1999] Grimm, V. (1999). Ten years of individual-based modelling in ecology: what have we learned, and what could we learn in the future? *Ecological Modelling*, 115(2–3):129–148.
- [Grimm et al., 2006] Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., Huth, A., Jepsen, J., Jørgensen, C., Mooij, W. M., Mueller, B., Peer, G., Piou, C., Railsback, S. F., Robbins, A. M., Robbins, M. M., Rossmanith, E., Rüger, N., Strand, E., Souissi, S., Stillman, R. A., Vab, R., Visser, U., and DeAngelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198(1):115–126.
- [Grimm and Railsback, 2005] Grimm, V. and Railsback, S. F. (2005). *Individual-based Modeling and Ecology*. Princeton University Press, New Jersey.
- [Guo et al., 2005] Guo, Z., Han, H. K., and Tay, J. C. (2005). Sufficiency verification of hiv-1 pathogenesis based on multi-agent simulation. In Beyer, H.-G. and O’Reilly, U.-M., editors, *GECCO*, pages 305–312. ACM.
- [Houghton et al., 1997] Houghton, J. T., Filho, L. G. M., Griggs, D. J., and Maskell, K. (1997). Stabilization of atmospheric greenhouse gases: Physical, biological and socio-economic implications - ipcc technical paper iii. *IPCC*, 3:51.
- [IPCC, 2007] IPCC (2007). Climate change 2007: The physical science basis - working group i of the ipcc. ”<http://www.ipcc.ch/SPM2feb07.pdf> . Acesso em: 15 ago. 2007”.

- [Jennings and Wooldridge, 1999] Jennings, N. R. and Wooldridge, M. (1999). Agent-Oriented Software Engineering. In Garijo, F. J. and Boman, M., editors, *Proceedings of the 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World : Multi-Agent System Engineering (MAAMAW-99)*, volume 1647, pages 1–7. Springer-Verlag: Heidelberg, Germany.
- [Johnson, 1997] Johnson, R. E. (1997). Components, frameworks, patterns. In *ACM SIGSOFT Symposium on Software Reusability*, pages 10–17.
- [Krasner and Pope, 1988] Krasner, G. E. and Pope, S. T. (1988). A cookbook for using the model-view-controller user interface paradigm in smalltalk-80. *Journal of Object Oriented Programming (JOOP)*, 1(3):26–49.
- [Leonel and Yoneyama, 2000] Leonel, M. A. and Yoneyama, T. (2000). Optimal and sub-optimal control in dengue epidemics. *Optimal Control Applications and Methods*.
- [Luenberger, 1989] Luenberger, D. G. (1989). *Linear and Nonlinear Programming*. Addison-Wesley Publishing Co., second edition. "Reprinted with corrections in May, 1989".
- [Luke et al., 2005] Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., and Balan, G. (2005). Mason: A multiagent simulation environment. *Simulation*, 81(7):517–527.
- [Maguire et al., 1999] Maguire, M., Skelly, C., Weinstein, P., and Moloney, J. (1999). Simulation modelling of *Aedes aegypti* prevalence, an environmental hazard surveillance tool for the control of dengue epidemics. *International Journal of Environmental Health Research*, 9(4):253–259.
- [Massive software, 2007] Massive software (2007). Massive software. <http://www.massivesoftware.com> . Acesso em: 21 ago. 2007.
- [McNeil, 1976] McNeil, W. (1976). Plagues and peoples.
- [Microsoft Corporation, 2007] Microsoft Corporation (2007). Visio Homepage - Microsoft Office Online. <http://office.microsoft.com/visio> . Acesso em: 06 jul. 2007.
- [Minar et al., 1996] Minar, N., Burkhart, R., Langton, C., and Askenazi, M. (1996). The swarm simulation system, a toolkit for building multi-agent simulations.
- [Ministério da Saúde, 2005] Ministério da Saúde (2005). Dia D de combate a dengue. <http://www.saude.gov.br/dengue> . Acesso em: 20 mai. 2006.

- [Ministério da Saúde, 2007] Ministério da Saúde (2007). Portal da Saúde. <http://portal.saude.gov.br/portal> . Acesso em: 23 set. 2007.
- [NetLogo, 2007] NetLogo (2007). NetLogo Home Page. <http://ccl.northwestern.edu/netlogo> . Acesso em: 15 ago. 2007.
- [Newton and Reiter, 1992] Newton, E. and Reiter, P. (1992). A model of the transmission of dengue fever with an evaluation of the impact of ultra-low volume (ulv) insecticide applications on dengue epidemics. *American Journal of Tropical Medicine and Hygiene*, 47:709–720.
- [North et al., 2006] North, M. J., Collier, N. T., and Vos, J. R. (2006). Experiences creating three implementations of the repast agent modeling toolkit. *ACM Trans. Model. Comput. Simul.*, 16(1):1–25.
- [North and Macal, 2005] North, M. J. and Macal, C. M. (2005). *Artificial Life Models in Software*, chapter Escaping the accidents of history: an overview of artificial life modeling with Repast. Springer London, Dordrecht, Netherlands.
- [openCollabNet, 2007] openCollabNet (2007). subversion.tigris.org - Open Source Software Engineering Tools. <http://subversion.tigris.org> . Acesso em: 04 jul. 2007.
- [Otero et al., 2006] Otero, M., Solari, H. G., and Schweigmann, N. (2006). A stochastic population dynamics model for *Aedes aegypti*: formulation and application to a city with temperate climate. *Bulletin of Mathematical Biology*, 68:1945–1974.
- [Parunak et al., 1998] Parunak, H. V. D., Savit, R., and Riolo, R. L. (1998). Agent-based modeling vs. equation-based modeling: A case study and users’ guide. In Sichman, C. and Gilbert, editors, *Multi-agent systems and Agent-based Simulation (MABS’98)*, volume 1534 of *LNAI series*. Springer-Verlag.
- [Peck, 2004] Peck, S. L. (2004). Simulation as experiment: a philosophical reassessment for biological modeling. *Trends in Ecology and Evolution*.
- [Pessoa et al., 2004] Pessoa, T. R. M., Gagliardi, H. F., and Alves, D. (2004). Desenvolvimento de autômato celular probabilístico para estudar a transmissão e espalhamento da dengue. *XXXI Seminário Integrado de Software e Hardware*.
- [Peterson, 1998] Peterson, J. (1998). Global warming would foster spread of dengue fever into some temperate regions. <http://www.niehs.nih.gov/oc/news/global.htm> . Acesso em: 10 mai. 2007.

- [Railsback et al., 2006] Railsback, S. F., Lytinen, S. L., and Jackson, S. K. (2006). Agent-based simulation platforms: Review and development recommendations. *Simulation*, 82(9):609–623.
- [Reynolds, 2007a] Reynolds, C. (2007a). Boids (Flocks, Herds, and Schools: a Distributed Behavioral Model). <http://www.red3d.com/cwr/boids/> . Acesso em: 31 jul. 2007.
- [Reynolds, 2007b] Reynolds, C. (2007b). Game research and technology. <http://www.red3d.com/cwr/games/> . Acesso em: 31 jul. 2007.
- [Rich and Knight, 1991] Rich, E. and Knight, S. (1991). *Artificial Intelligence*. McGraw Hill, 2 edition.
- [Russell and Norvig, 2003] Russell, S. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Prentice Hall, Englewood Cliffs, New Jersey 07632, second edition edition.
- [Sandholm, 1999] Sandholm, T. (1999). *Multi-agent Systems: A Modern Introduction to Distributed Artificial Intelligence*, chapter Distributed Rational Decision Making, pages 201 – 258. MIT Press.
- [Sauvage, 2004] Sauvage, S. (2004). Agent oriented design patterns: A case study. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1496–1497, Washington, DC, USA. IEEE Computer Society.
- [Sichman, 2003] Sichman, J. S. (2003). *Raciocínio social e organizacional em sistemas multiagentes: avanços e perspectivas*. Tese para obtenção do Título de Professor Livre Docente. PhD thesis, Universidade de São Paulo.
- [Silva and Lucena, 2004] Silva, V. T. D. and Lucena, C. J. P. D. (2004). From a conceptual framework for agents and objects to a multi-agent system modeling language. *Autonomous Agents and Multi-Agent Systems*, 9(1-2):145–189.
- [Situngkir, 2005] Situngkir, H. (2005). Epidemiology through cellular automata - case of study: Avian influenza in indonesia. *Computational Sociology Research - Working Paper - Bandung Fe Institute*.
- [Sun Microsystems, 2007] Sun Microsystems (2007). Java technology. <http://java.sun.com/> . Acesso em: 03 jul. 2007.

- [Swan, 1984] Swan, G. W. (1984). *Applications of Optimal Control Theory in Biomedicine*. Marcel Dekker.
- [Swarm Development Group, 2004] Swarm Development Group (2004). Swarm Development Group:Swarm 2.2. <http://wiki.swarm.org> . Acesso em: 24 jul. 2007.
- [Takahashi et al., 2005] Takahashi, L. T., Maidana, N. A., Ferreira, W. C. J., Pulino, P., and Yang, H. M. (2005). Mathematical models for the *Aedes aegypti* dispersal dynamics: travelling waves by wing and wind. *Bulletin of Mathematical Biology*, 67(3):509–528.
- [Templin, 2000] Templin, R. J. (2000). The spectrum of animal flight: insects to pterosaurs. *Progress in Aerospace Sciences*, 36(5):393–436.
- [The Eclipse Foundation, 2007] The Eclipse Foundation (2007). Eclipse.org home. <http://www.eclipse.org/> . Acesso em: 05 jul. 2007.
- [Vlassis, 2003] Vlassis, N. (2003). A concise introduction to multiagent systems and distributed AI. Informatics Institute, University of Amsterdam.
- [Wooldridge, 2002] Wooldridge, M. (2002). *An Introduction to MultiAgent Systems*. Wiley, Chichester.
- [Wooldridge and Jennings, 1994] Wooldridge, M. and Jennings, N. R. (1994). Intelligent agents: Theory and practice. Available by FTP. Submitted to The Knowledge Engineering Review, 1995.
- [World Health Organization, 2002] World Health Organization (2002). Dengue and dengue haemorrhagic fever. <http://www.who.int/mediacentre/factsheets/fs117/en/> . Acesso em: 13 mar. 2006.
- [Yang, 2003] Yang, H. M. (2003). Epidemiologia da transmissão da dengue. *CNMAC - Congresso Nacional de Matemática Aplicada e Computacional*, 4(3):387–396.
- [Yang et al., 2002] Yang, H. M., Ternes, S., and Ferreira, C. P. (2002). Dinâmica populacional do vetor transmissor da dengue. *CNMAC - Congresso Nacional de Matemática Aplicada e Computacional*, 1(2):478–478.
- [Zaki and Sobh, 2005] Zaki, M. and Sobh, T. S. (2005). Attack abstraction using a multiagent system for intrusion detection. *J. Intell. Fuzzy Syst.*, 16(2):141–150.

- [Zambonelli et al., 2003] Zambonelli, F., Jennings, N. R., and Wooldridge, M. (2003). Developing multiagent systems: The gaia methodology. *ACM Trans. Softw. Eng. Methodol.*, 12(3):317–370.

Apêndice A

Dados Meteorológicos

O simulador de população de mosquitos proposto é capaz de realizar simulações de diferentes tipos de cenários. Nestes cenários são disponíveis informações meteorológicas. As principais variáveis meteorológicas disponíveis no simulador são:

1. *Direção do vento na velocidade máxima (° NV)*
2. *Direção do vento (° NV)*
3. *Precipitação acumulada (mm)*
4. *Radiação solar acumulada (MJ/m²)*
5. *Radiação solar global (MJ/m²)*
6. *Temperatura do ar (°C)*
7. *Temperatura máxima do ar nas últimas 24 horas (°C)*
8. *Temperatura mínima do ar nas últimas 24 horas (°C)*
9. *Umidade relativa do ar (%)*
10. *Velocidade do vento 10 m (m/s)*
11. *Velocidade do vento 3 m (m/s)*
12. *Velocidade máxima do vento (Rajada) (m/s)*

Na definição do cenário de simulação é necessário informar a base de dados que irá fornecer dados para estas variáveis. Então, quando a simulação do cenário for executada,

automaticamente mecanismos internos do simulador irão consultar a base de dados definida e atribuir às variáveis meteorológicas do modelo de simulação os valores recuperados da base de dados.

Os registros armazenados nas bases de dados são identificados pelo campo “chave” data e hora. Desta forma, as consultas às bases de dados devem ser realizadas considerando a data e hora atual do “mundo simulado”.

Na configuração do cenário é necessário definir o intervalo de atualização dos dados. Assim, o sistema irá buscar na base de dados os valores das variáveis periodicamente. A definição da periodicidade dependerá do propósito de cada cenário e dos dados meteorológicos disponíveis.

Os registros com os dados meteorológicos são armazenados em arquivos no formato *XML*. O padrão interno destes arquivos é baseado no padrão assumido pelos dados disponibilizados pelo *Centro de Previsão de Tempo e Estudos Climáticos do Instituto Nacional de Pesquisas Espaciais - CPTEC/INPE* [CPTEC-INPE, 2007]. Este padrão consiste em um conjunto de marcadores *XML*. Um exemplo de um arquivo *XML* com um registro referente a dados meteorológicos coletados em determinado instante é apresentado a seguir:

```
<?xml version="1.0"?>
<table>
  <row>
    <column>2005-12-16 00:00:00.0</column>
    <column>100</column>
    <column>130</column>
    <column>261.25</column>
    <column>0</column>
    <column></column>
    <column>25</column>
    <column>29</column>
    <column>19</column>
    <column>60</column>
    <column>0.1</column>
    <column>0.1</column>
    <column>0.9</column>
  </row>
</table>
```


Observa-se neste exemplo que existem 13 marcadores do tipo “column”. Cada marcador “column” precede um dado meteorológico, com exceção do primeiro dado, que indica a data e hora que foram coletados os dados meteorológicos. Os demais dados disponíveis neste registro referem-se às variáveis meteorológicas listadas anteriormente (itens 1 a 12), e seguem a mesma ordem da lista.

Apêndice B

Distribuição de Ovos do Mosquito *Aedes aegypti*

Gomes et al. determinam um padrão de distribuição temporal de ovos por fêmeas do mosquito *Aedes aegypti* alimentadas e copuladas. Este padrão foi extraído a partir da observação de distribuições de ovos em experimentos sob condições de laboratório e campo. O padrão proposto mostra a distribuição de ovos pelas fêmeas ao longo dos dias que sucedem a última alimentação sangüínea. Além disto o padrão inclui a distribuição de ovos em diversos horários destes dias [Gomes et al., 2006].

Neste contexto o modelo de simulação proposto incorpora um padrão de distribuição de ovos similar ao apresentado em [Gomes et al., 2006]. Na figura B.1 é possível visualizar a distribuição dos ovos após o repasto sangüíneo e cópula das fêmeas.

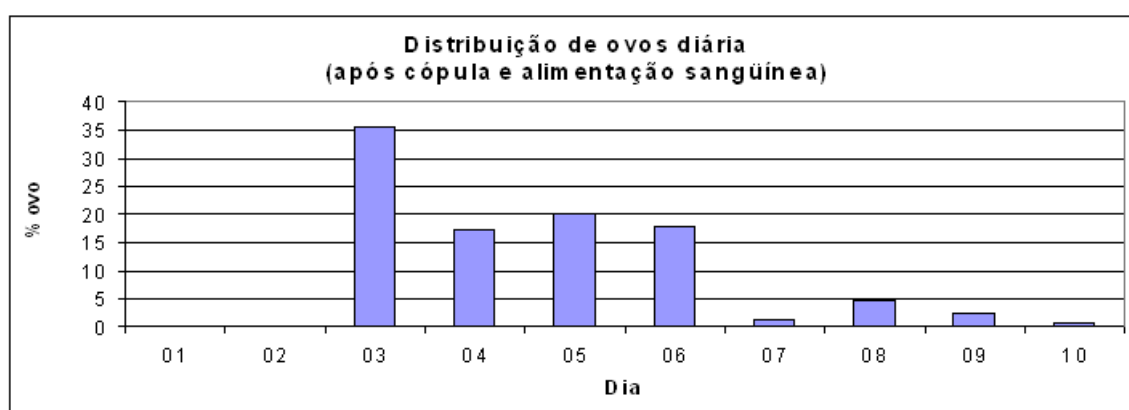


Figura B.1: Distribuição diária de ovos de *Aedes aegypti*

O gráfico da figura B.1 foi construído sob os dados apresentados na tabela B.1. Considerando este gráfico e esta tabela observa-se que após o segundo dia a fêmea começa a

colocar os ovos. O número de ovos colocado pelas fêmeas nos primeiros dias de oviposição é maior, e gradativamente vai diminuindo até o 10º dia.

Dia	% Ovos
01	0
02	0
03	35,7
04	17,4
05	20
06	17,8
07	1,2
08	4,7
09	2,5
10	0,7
Total	100

Tabela B.1: Percentuais de distribuição diária de ovos de *Aedes aegypti*

A distribuição de ovos pelas fêmeas do *Aedes aegypti* durante todo o dia obedece ao padrão de distribuição mostrado na figura B.2. A tabela B.2 complementa este gráfico, pois numericamente é possível observar os percentuais de distribuição de ovos.

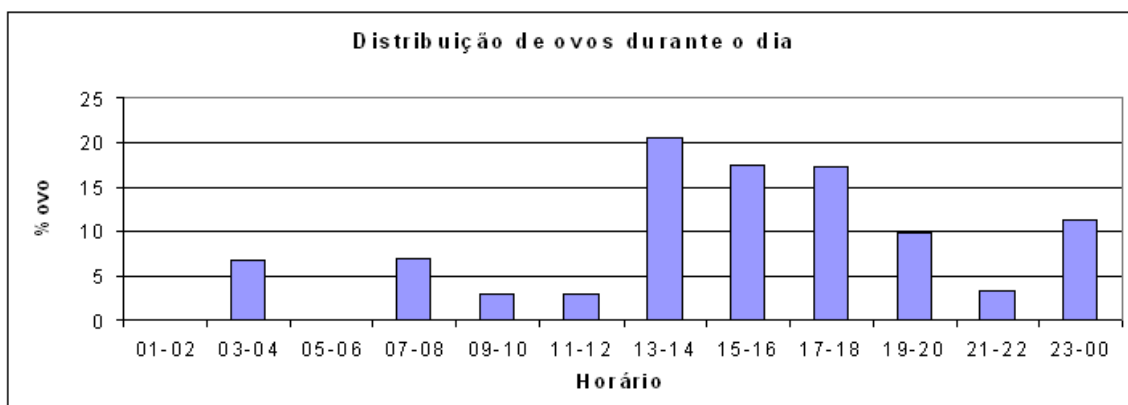


Figura B.2: Distribuição de ovos de *Aedes aegypti* durante o dia

Observa-se que das 13 horas a 1 hora do dia seguinte a fêmea coloca um número maior de ovos.

Horário	% Ovos
01:00 hs às 02:59 hs	0
03:00 hs às 04:59 hs	6,93
05:00 hs às 06:59 hs	0
07:00 hs às 08:59 hs	7
09:00 hs às 10:59 hs	3,11
11:00 hs às 12:59 hs	3,1
13:00 hs às 14:59 hs	20,52
15:00 hs às 16:59 hs	17,4
17:00 hs às 18:59 hs	17,3
19:00 hs às 20:59 hs	9,9
21:00 hs às 22:59 hs	3,42
23:00 hs às 00:59 hs	11,32
Total	100

Tabela B.2: Percentuais de distribuição de ovos de *Aedes aegypti* durante o dia

Apêndice C

Green-House - Resultados Complementares

Neste apêndice alguns gráficos ilustrativos dos resultados de simulações computacionais são apresentados. Estes resultados se referem as simulações do experimento da *Green-House* no período *pré-armadilha*.

C.1 Evolução da produção de ovos

O gráfico apresentado na figura C.1 ilustra duas seqüências de dados obtidos em experimentos. A primeira (verde) indica o número de ovos coletados em determinados dias no experimento da *Green-house*. A segunda seqüência (azul) indica o número médio de ovos coletados diariamente nas simulações computacionais. As curvas apresentam comportamentos similares.

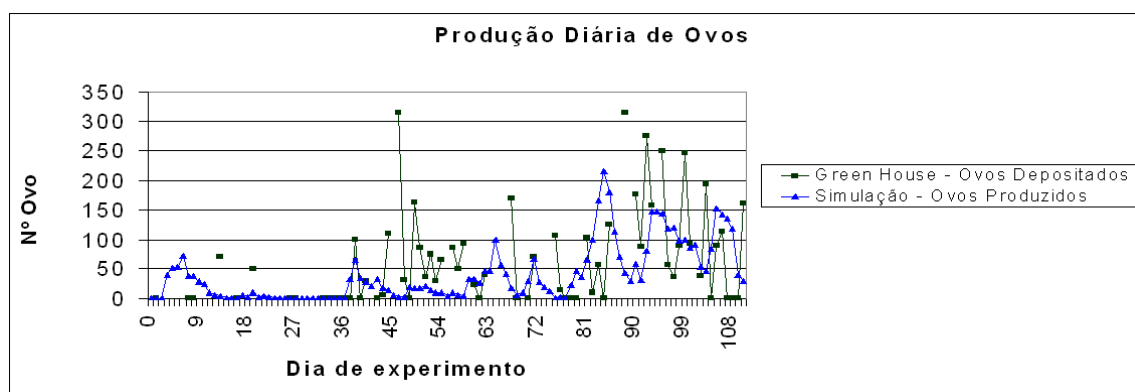


Figura C.1: Produção diária de ovos no período pré-armadilha

Na figura C.2 são apresentados gráficos com totalizações parciais dos dados.

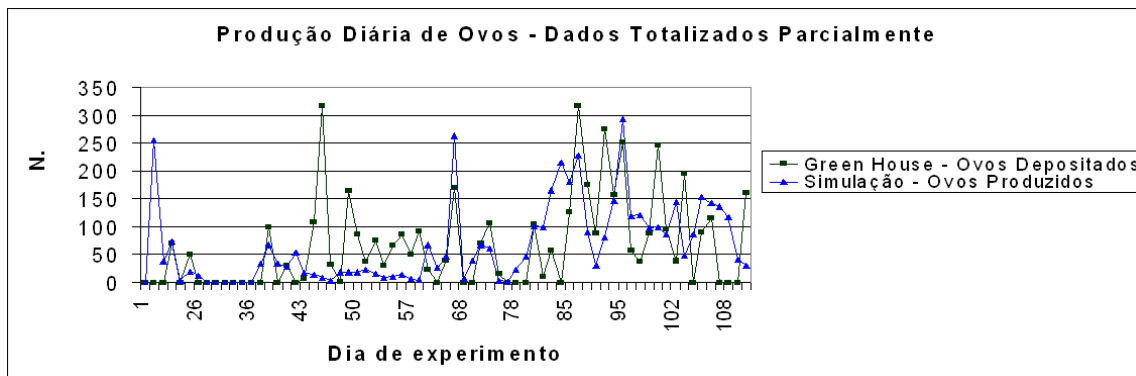


Figura C.2: Produção diária de ovos no período pré-armadilha - Dados totalizados parcialmente

Nota-se que as curvas apresentadas na figura C.2 são mais próximas do que as curvas da figura C.1. Desta maneira, este critério de totalização é adotado nas curvas dos próximos gráficos, ou seja, nos próximos gráficos as curvas possuem pontos de totalização em função dos dias em que foram coletados ovos e exúvias no experimento real.

Outra forma de visualizar a dinâmica da produção de ovos nos experimentos realizados é através da utilização de gráficos com totalizações semanais e quinzenais das produções de ovos. Nas figuras C.3 e C.4 os gráficos *semanal* e *quinzenal* podem ser observados.

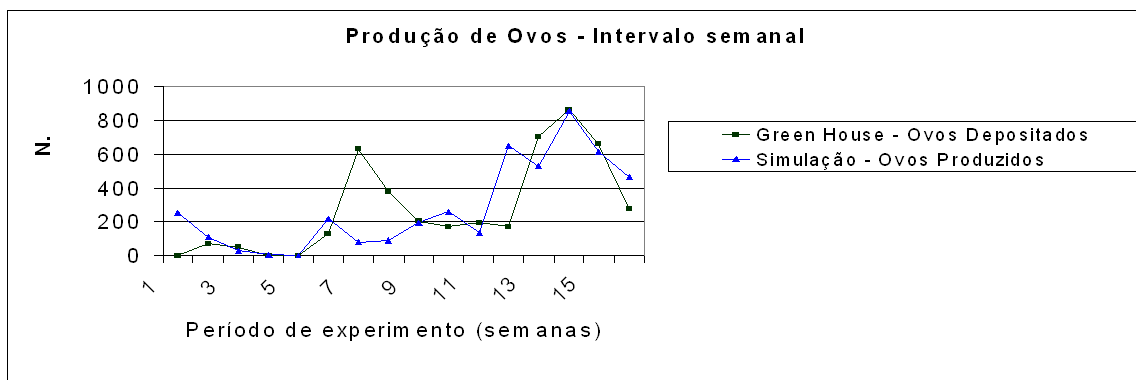


Figura C.3: Produção semanal de ovos no período pré-armadilha - Janela fixa

Nestes gráficos é possível observar um fenômeno interessante com mais nitidez do que nos outros gráficos apresentados. Uma diferença considerável entre a curva do experimento real (verde) e a curva da simulação computacional (azul), no período da 7ª até a 10ª semana (ou na 4ª quinzena), ocorre. Acredita-se que esta diferença ocorra por motivos relacionados a detalhes do experimento real que não são considerados no modelo de simulação proposto. A principal hipótese está relacionada ao modelo de distribuição temporal de ovos adotado, ou seja, o modelo adotado talvez seja mais simples e deter-

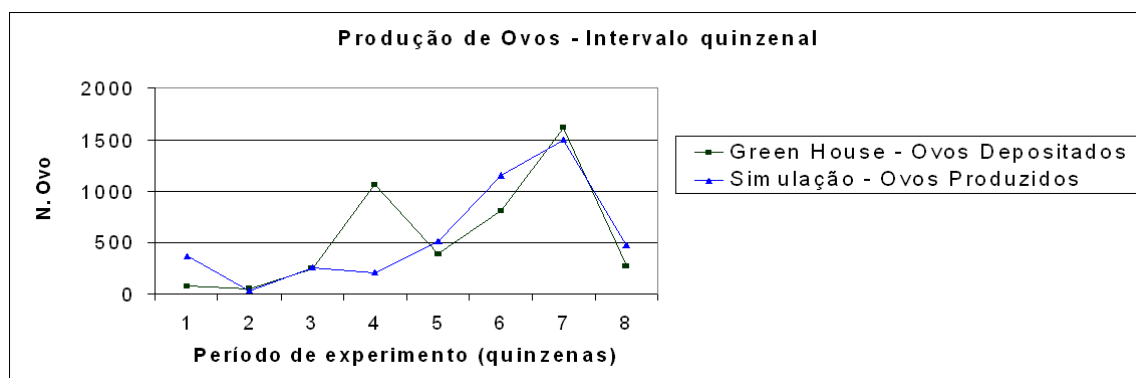


Figura C.4: Produção quinzenal de ovos no período pré-armadilha - Janela fixa

minístico do que o modelo real de distribuição de ovos dos mosquitos *Aedes aegypti*. Contudo, observa-se que mesmo existindo esta variação entre as curvas, de forma geral, estas apresentam comportamentos similares.

C.2 Evolução da produção de adultos

O gráfico existente na figura C.5 ilustra uma similaridade entre os resultados coletados no experimento real da *Green-House* e nas simulações computacionais deste experimento.

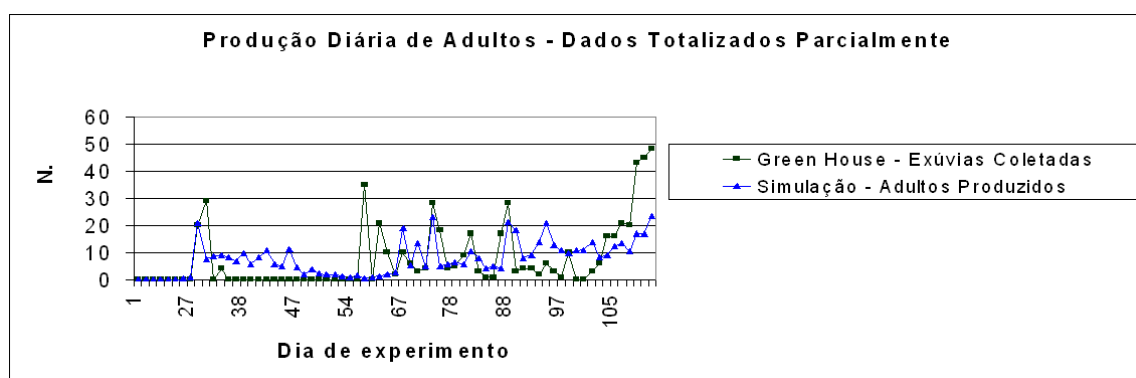


Figura C.5: Produção diária de adultos no período pré-armadilha

Na figura C.5 a curva de produção de exúvias (verde) e a curva de produção de ovos (azul) se aproximam em diversos intervalos. Entre os dias 67 e 105, a tendência das curvas foi praticamente a mesma, variando apenas a altura de algumas cristas e vales das curvas.

A fim de complementar esta análise, as produções de adultos/exúvias são apresentados nas figuras C.6 e C.7 dados consolidados semanalmente e quinzenalmente da produção de adultos/exúvias.

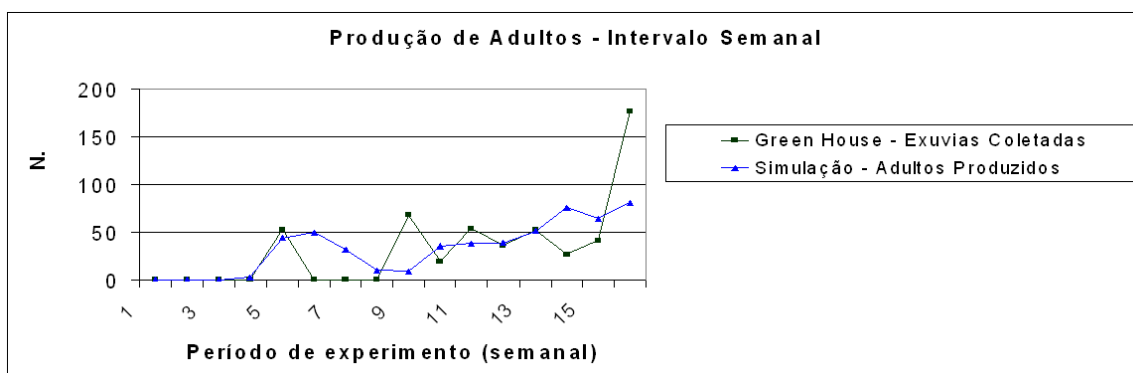


Figura C.6: Produção semanal de adultos no período pré-armadilha - Janela fixa

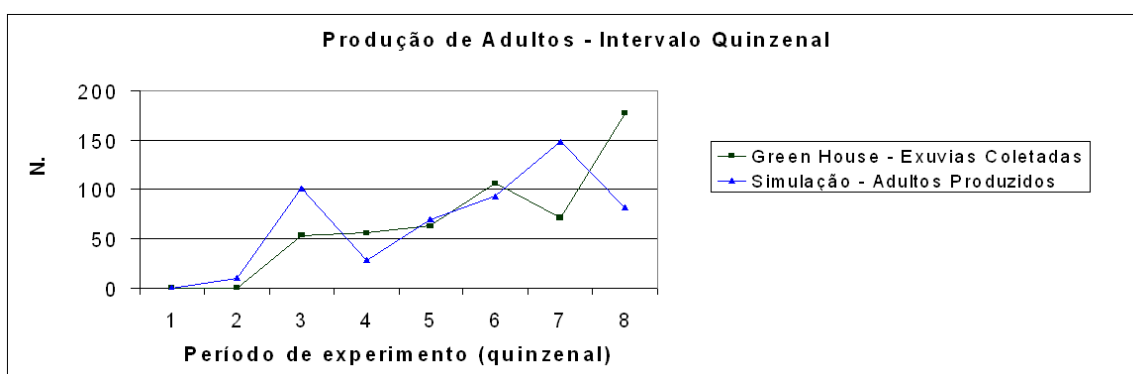


Figura C.7: Produção quinzenal de adultos no período pré-armadilha - Janela fixa

Estes gráficos que envolvem dados semanais e quinzenais oferecem uma visão melhor da dinâmica da produção de adultos/exúvias no decorrer de intervalos de tempo mais espaçados. Observa-se que nestes gráficos apresentados, os valores obtidos sempre permanecem entre 0 e 200 adultos/exúvias produzidos por intervalo de tempo.

C.3 Comportamento da taxa de picadas

Uma informação que foi analisada nos experimentos da *Green-House* é a taxa de picadas dos mosquitos *Aedes aegypti*. Como descrito anteriormente, esta taxa indica quantos mosquitos picam algum ser humano durante 1 minuto. Na figura C.8 é possível visualizar o comportamento da taxa de picadas no experimento real da *Green-House* (curva verde) e nas simulações computacionais (curva azul).

Observa-se que estas curvas se mantêm próximas. De fato, isto já é esperado, uma vez que o número de picadas em ambos os experimentos é o mesmo. No entanto, a duração das picadas tende a modificar o comportamento das curvas. Este dado não foi coletado

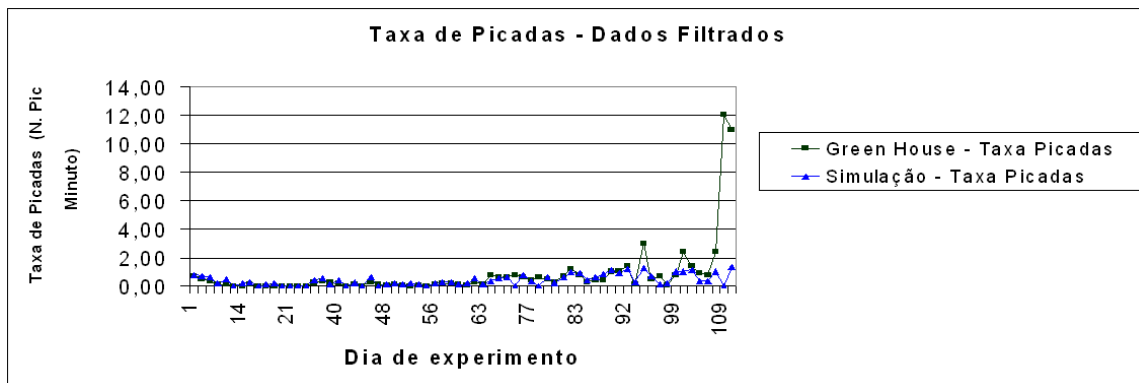


Figura C.8: Taxa de Picadas

no experimento real da *Green-House*. Nos dias finais do período *pré-armadilha*, a taxa de picadas se elevou no experimento da *Green-House*, justificando desta maneira a instalação de uma armadilha protótipo para controle da população de mosquitos.

Apêndice D

Parâmetros

Os parâmetros do modelo de simulação *SimPopMosq* e os parâmetros relacionados à infra-estrutura do simulador de população de mosquitos são armazenados em arquivos no formato *XML*. O conteúdo destes arquivos é apresentado neste apêndice.

D.1 Parâmetros da simulação

Os parâmetros do modelo de simulação são armazenados no arquivo *ParametrosEntrada.XML*. O conteúdo deste arquivo é apresentado a seguir:

```
<?xml version="1.0" encoding="UTF-8"?>
<Repast:Params xmlns:Repast="http://www.src.uchicago.edu">
<Repast:ParamBlock runs="1">
  <Repast:Param name="Instancia" type="set" value="2" io="input"/>

  <Repast:Param name="TaxaEvaporacaoRastroUmidade" type="set"
    value="0.99" io="input"/>
  <Repast:Param name="ConstanteDifusaoRastroUmidade" type="set"
    value="1.0" io="input"/>
  <Repast:Param name="MaxIntensidadeRastroAgua" type="set"
    value="10000" io="input"/>
  <Repast:Param name="MinIntensidadeRastroAgua" type="set"
    value="3000" io="input"/>

  <Repast:Param name="TaxaEvaporacaoRastroVegetacao" type="set"
    value="0.99" io="input"/>
```

```
<Repast:Param name="ConstanteDifusaoRastroVegetacao" type="set"
  value="1.0" io="input"/>
<Repast:Param name="MaxIntensidadeRastroVegetacao" type="set"
  value="10000" io="input"/>
<Repast:Param name="MinIntensidadeRastroVegetacao" type="set"
  value="3000" io="input"/>

<Repast:Param name="NumMaxTentativasMoverMamifero" type="set"
  value="4" io="input"/>
<Repast:Param name="IntervaloDifusaoRastroMamifero" type="set"
  value="5" io="input"/>
<Repast:Param name="IntervaloTomadaDecisaoMamifero" type="set"
  value="10" io="input"/>
<Repast:Param name="IntervaloExecucaoComportamentosMamifero" type="set"
  value="1" io="input"/>
<Repast:Param name="TaxaEvaporacaoRastroMamifero" type="set"
  value="0.99" io="input"/>
<Repast:Param name="ConstanteDifusaoRastroMamifero" type="set"
  value="1.0" io="input"/>

<Repast:Param name="MaxIntensidadeRastroHumano" type="set"
  value="10000" io="input"/>
<Repast:Param name="MinIntensidadeRastroHumano" type="set"
  value="3000" io="input"/>
<Repast:Param name="VelocidadeDeslocandoAleatoriamenteHumano" type="set"
  value="1" io="input"/>
<Repast:Param name="VelocidadeDeslocandoAoDestinoHumano" type="set"
  value="2" io="input"/>

<Repast:Param name="MaxIntensidadeRastroCachorro" type="set"
  value="2000" io="input"/>
<Repast:Param name="MinIntensidadeRastroCachorro" type="set"
  value="1000" io="input"/>
<Repast:Param name="VelocidadeDeslocandoAleatoriamenteCachorro" type="set"
  value="1" io="input"/>
```

```
<Repast:Param name="MaxIntensidadeRastroGato" type="set"
  value="1000" io="input"/>
<Repast:Param name="MinIntensidadeRastroGato" type="set"
  value="500" io="input"/>
<Repast:Param name="VelocidadeDeslocandoAleatoriamenteGato" type="set"
  value="1" io="input"/>

<Repast:Param name="IntervaloDifusaoRastroAedes" type="set"
  value="5" io="input"/>
<Repast:Param name="IntervaloTomadaDecisaoAdultoAedes" type="set"
  value="10" io="input"/>
<Repast:Param name="IntervaloExecucaoComportamentosAdultoAedes" type="set"
  value="1" io="input"/>
<Repast:Param name="IntervaloVerificacaoMetamorfoseAedes" type="set"
  value="21150" io="input"/>
<Repast:Param name="DuracaoFaseOvoAedes" type="set"
  value="259200" io="input"/>
<Repast:Param name="DuracaoFaseLarvaAedes" type="set"
  value="1296000" io="input"/>
<Repast:Param name="DuracaoFasePupaAedes" type="set"
  value="259200" io="input"/>
<Repast:Param name="DuracaoFaseAdultoAedes" type="set"
  value="1296000" io="input"/>
<Repast:Param name="mediaDistribuicaoNormalFaseOvo" type="set"
  value="0" io="input"/>
<Repast:Param name="desvioPadraoDistribuicaoNormalFaseOvo" type="set"
  value="10" io="input"/>
<Repast:Param name="limiteVariacaoInferiorFaseLarva" type="set"
  value="-0.2" io="input"/>
<Repast:Param name="limiteVariacaoSuperiorFaseLarva" type="set"
  value="0.2" io="input"/>
<Repast:Param name="limiteVariacaoInferiorFasePupa" type="set"
  value="-0.1" io="input"/>
<Repast:Param name="limiteVariacaoSuperiorFasePupa" type="set"
  value="0.1" io="input"/>
<Repast:Param name="RiscoMorteMudancaFaseOvoAedes" type="set"
```

```
value="30" io="input"/>
<Repast:Param name="RiscoMorteMudancaFaseLarvaAedes" type="set"
value="30" io="input"/>
<Repast:Param name="RiscoMorteMudancaFasePupaAedes" type="set"
value="30" io="input"/>
<Repast:Param name="RiscoMorteMudancaFaseAdultoAedes" type="set"
value="90" io="input"/>
<Repast:Param name="numMaxTentativasMoverAedes" type="set"
value="4" io="input"/>
<Repast:Param name="alturaVooPadraoAedes" type="set"
value="1" io="input"/>
<Repast:Param name="velocidadeVerticalEmRelacaoAHorizontalAedes" type="set"
value="0.8" io="input"/>
<Repast:Param name="duracaoAlimentacaoSeivaAedes" type="set"
value="1200" io="input"/>
<Repast:Param name="intensidadeAltaRastroVegetacaoAedes" type="set"
value="4000" io="input"/>
<Repast:Param name="intensidadeMediaRastroVegetacaoAedes" type="set"
value="1000" io="input"/>

<Repast:Param name="TaxaEvaporacaoRastroAedesMacho" type="set"
value="0.99" io="input"/>
<Repast:Param name="ConstanteDifusaoRastroAedesMacho" type="set"
value="1.0" io="input"/>
<Repast:Param name="MaxIntensidadeRastroAedesMacho" type="set"
value="10000" io="input"/>
<Repast:Param name="MinIntensidadeRastroAedesMacho" type="set"
value="3000" io="input"/>
<Repast:Param name="velocidadeVoandoAleatoriamenteAedesMacho" type="set"
value="0.5" io="input"/>
<Repast:Param name="velocidadeProcurandoSossegoAedesMacho" type="set"
value="0.5" io="input"/>
<Repast:Param name="velocidadeProcurandoAlimentoAedesMacho" type="set"
value="0.5" io="input"/>
<Repast:Param name="velocidadeTentandoAlimentarAedesMacho" type="set"
value="0.7" io="input"/>
```

```
<Repast:Param name="velocidadeProcurandoParceiroAedesMacho" type="set"
  value="0.5" io="input"/>
<Repast:Param name="velocidadeTentandoAcasalarAedesMacho" type="set"
  value="0.7" io="input"/>
<Repast:Param name="nivelBaixoEnergiaAedesMacho" type="set"
  value="0" io="input"/>
<Repast:Param name="nivelAltoEnergiaAedesMacho" type="set"
  value="155520" io="input"/>
<Repast:Param name="nivelMedioEnergiaAedesMacho" type="set"
  value="77760" io="input"/>
<Repast:Param name="consumoEnergiaVoandoAleatoriamenteAedesMacho" type="set"
  value="0.6" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoSossegoAedesMacho" type="set"
  value="0.6" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoAlimentoAedesMacho" type="set"
  value="0.7" io="input"/>
<Repast:Param name="consumoEnergiaTentandoAlimentarAedesMacho" type="set"
  value="1.0" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoParceiroAedesMacho" type="set"
  value="0.7" io="input"/>
<Repast:Param name="consumoEnergiaTentandoAcasalarAedesMacho" type="set"
  value="1.0" io="input"/>
<Repast:Param name="intensidadeAltaRastroFemeaParaAedesMacho" type="set"
  value="4000" io="input"/>
<Repast:Param name="intensidadeMediaRastroFemeaParaAedesMacho" type="set"
  value="1000" io="input"/>
<Repast:Param name="TaxaEvaporacaoRastroAedesFemea" type="set"
  value="0.99" io="input"/>
<Repast:Param name="ConstanteDifusaoRastroAedesFemea" type="set"
  value="1.0" io="input"/>
<Repast:Param name="MaxIntensidadeRastroAedesFemea" type="set"
  value="10000" io="input"/>
<Repast:Param name="MinIntensidadeRastroAedesFemea" type="set"
  value="3000" io="input"/>
<Repast:Param name="velocidadeVoandoAleatoriamenteAedesFemea" type="set"
  value="0.5" io="input"/>
```

```
<Repast:Param name="velocidadeProcurandoSossegoAedesFemea" type="set"
  value="0.5" io="input"/>
<Repast:Param name="velocidadeProcurandoAlimentoAnimalAedesFemea" type="set"
  value="0.5" io="input"/>
<Repast:Param name="velocidadeTentantoAlimentarAnimalNormalAedesFemea" type="set"
  value="0.6" io="input"/>
<Repast:Param name="velocidadeTentantoAlimentarAnimalVelozAedesFemea" type="set"
  value="0.7" io="input"/>
<Repast:Param name="velocidadeProcurandoAlimentoSeivaAedesFemea" type="set"
  value="0.45" io="input"/>
<Repast:Param name="velocidadeProcurandoCriadouroDesovaAedesFemea" type="set"
  value="0.5" io="input"/>
<Repast:Param name="nivelBaixoEnergiaAedesFemea" type="set"
  value="0" io="input"/>
<Repast:Param name="nivelAltoEnergiaAedesFemea" type="set"
  value="155520" io="input"/>
<Repast:Param name="nivelMedioEnergiaAedesFemea" type="set"
  value="77760" io="input"/>
<Repast:Param name="duracaoAlimentacaoAnimalFemea" type="set"
  value="900" io="input"/>
<Repast:Param name="maxSangueIngeridoPorPicadaFemea" type="set"
  value="90" io="input"/>
<Repast:Param name="consumoEnergiaVoandoAleatoriamenteAedesFemea" type="set"
  value="0.6" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoSossegoAedesFemea" type="set"
  value="0.6" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoAlimentoAnimalAedesFemea" type="set"
  value="0.8" io="input"/>
<Repast:Param name="consumoEnergiaTentantoAlimentarAnimalNormalAedesFemea" type="set"
  value="0.9" io="input"/>
<Repast:Param name="consumoEnergiaTentantoAlimentarAnimalVelozAedesFemea" type="set"
  value="1.0" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoAlimentoSeivaAedesFemea" type="set"
  value="0.9" io="input"/>
<Repast:Param name="consumoEnergiaProcurandoCriadouroDesovaAedesFemea" type="set"
  value="0.8" io="input"/>
```

```
<Repast:Param name="sucessoPicadaAedesFemea" type="set"
  value="80" io="input"/>
<Repast:Param name="riscoMortePicadaAedesFemea" type="set"
  value="0" io="input"/>
<Repast:Param name="intensidadeAltaRastroHumanoParaAedesFemea" type="set"
  value="4000" io="input"/>
<Repast:Param name="intensidadeMediaRastroHumanoParaAedesFemea" type="set"
  value="1000" io="input"/>
<Repast:Param name="intensidadeAltaRastroMachoParaAedesFemea" type="set"
  value="4000" io="input"/>
<Repast:Param name="intensidadeMediaRastroMachoParaAedesFemea " type="set"
  value="1000" io="input"/>
<Repast:Param name="minDuracaoReproducaoAedesFemea" type="set"
  value="300" io="input"/>
<Repast:Param name="maxDuracaoReproducaoAedesFemea" type="set"
  value="600" io="input"/>
<Repast:Param name="minNumOvosPosturaAedesFemea" type="set"
  value="80" io="input"/>
<Repast:Param name="maxNumOvosPosturaAedesFemea" type="set"
  value="120" io="input"/>
<Repast:Param name="velocidadeOvoposicaoAedesFemea" type="set"
  value="0.02" io="input"/>
<Repast:Param name="numNascimentoMachoParaCadaAedesFemea" type="set"
  value="1" io="input"/>
<Repast:Param name="distribuicaoOvoposicaoDia01" type="set"
  value="0" io="input"/>
<Repast:Param name="distribuicaoOvoposicaoDia02" type="set"
  value="0" io="input"/>
<Repast:Param name="distribuicaoOvoposicaoDia03" type="set"
  value="35.7" io="input"/>
<Repast:Param name="distribuicaoOvoposicaoDia04" type="set"
  value="17.4" io="input"/>
<Repast:Param name="distribuicaoOvoposicaoDia05" type="set"
  value="20" io="input"/>
<Repast:Param name="distribuicaoOvoposicaoDia06" type="set"
  value="17.8" io="input"/>
```



```
<Repast:Param name="distribuicao0voposicaoDia07" type="set"
  value="1.2" io="input"/>
<Repast:Param name="distribuicao0voposicaoDia08" type="set"
  value="4.7" io="input"/>
<Repast:Param name="distribuicao0voposicaoDia09" type="set"
  value="2.5" io="input"/>
<Repast:Param name="distribuicao0voposicaoDia10" type="set"
  value="0.7" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_01_02" type="set"
  value="0" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_03_04" type="set"
  value="6.93" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_05_06" type="set"
  value="0" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_07_08" type="set"
  value="7" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_09_10" type="set"
  value="3.11" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_11_12" type="set"
  value="3.10" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_13_14" type="set"
  value="20.52" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_15_16" type="set"
  value="17.4" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_17_18" type="set"
  value="17.3" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_19_20" type="set"
  value="9.9" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_21_22" type="set"
  value="3.42" io="input"/>
<Repast:Param name="distribuicao0voposicaoHora_23_00" type="set"
  value="11.32" io="input"/>

<Repast:Param name="IntervaloDifusaoRastroNatureza" type="set"
  value="5" io="input"/>
<Repast:Param name="IntervaloAtualizacaoNivelAgua" type="set"
```

```

        value="10800" io="input"/>

</Repast:ParamBlock>
</Repast:Params>

```

D.2 Parâmetros de infra-estrutura

Os parâmetros do modelo de infra-estrutura do simulador de população de mosquitos são armazenados no arquivo *parametrosInfraestrutura.xml*. O conteúdo deste arquivo é apresentado a seguir:

```

<?xml version="1.0" encoding="UTF-8"?>
<Repast:Params xmlns:Repast="http://www.src.uchicago.edu">
<Repast:ParamBlock runs="1">
  <Repast:Param name="TipoSimulacao" type="set"
    value="0" io="input"/>
  <Repast:Param name="NumDimensoes" type="set"
    value="2" io="input"/>
  <Repast:Param name="AtualizacaoGrafica" type="set"
    value="100" io="input"/>
  <Repast:Param name="ColetorLixo" type="set"
    value="100" io="input"/>
  <Repast:Param name="NumElementosCelulaInicial" type="set"
    value="3" io="input"/>
  <Repast:Param name="FinalizarProcessoAoFinal" type="set"
    value="1" io="input"/>
  <Repast:Param name="SegundosPorTick" type="set"
    value="1" io="input"/>
  <Repast:Param name="MetrosPorCelula" type="set"
    value="0.5" io="input"/>
  <Repast:Param name="NumSegundosPorDia" type="set"
    value="86400" io="input"/>
  <Repast:Param name="NumCelulasRastroPorCelulaGrid" type="set"
    value="1" io="input"/>
</Repast:ParamBlock>
</Repast:Params>

```

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)