

UNIVERSIDADE DE PASSO FUNDO
FACULDADE DE AGRONOMIA E MEDICINA VETERINÁRIA
PROGRAMA DE PÓS-GRADUAÇÃO EM AGRONOMIA

**TÉCNICAS DE ENGENHARIA DE
SOFTWARE APLICADAS À MODELAGEM E
SIMULAÇÃO DE DOENÇAS DE PLANTAS**

WILLINGTHON PAVAN

Tese apresentada ao Programa de Pós-graduação em Agronomia da Faculdade de Agronomia e Medicina Veterinária da Universidade de Passo Fundo, para obtenção do título de Doutor em Agronomia - Área de Concentração em Fitopatologia.

Passo Fundo, julho de 2007.

UNIVERSIDADE DE PASSO FUNDO
FACULDADE DE AGRONOMIA E MEDICINA VETERINÁRIA
PROGRAMA DE PÓS-GRADUAÇÃO EM AGRONOMIA

**TÉCNICAS DE ENGENHARIA DE
SOFTWARE APLICADAS À MODELAGEM E
SIMULAÇÃO DE DOENÇAS DE PLANTAS**

WILLINGTHON PAVAN

Orientador: Prof. PhD. José Maurício Cunha Fernandes

Tese apresentada ao Programa de Pós-graduação em Agronomia da Faculdade de Agronomia e Medicina Veterinária da Universidade de Passo Fundo, para obtenção do título de Doutor em Agronomia - Área de Concentração em Fitopatologia.

Passo Fundo, julho de 2007.



UNIVERSIDADE DE PASSO FUNDO
FACULDADE DE AGRONOMIA E MEDICINA VETERINÁRIA
PROGRAMA DE PÓS-GRADUAÇÃO EM AGRONOMIA
ÁREA DE CONCENTRAÇÃO EM FITOPATOLOGIA



A Comissão Examinadora, abaixo assinada, aprova a tese

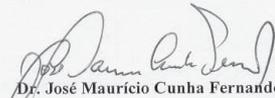
“Técnicas de engenharia de software aplicadas à modelagem
e simulação de doenças de plantas”

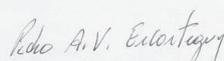
Elaborada por

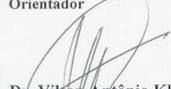
WILLINGTHON PAVAN

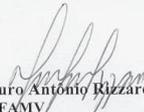
Como requisito parcial para a obtenção do grau de
Doutor em Agronomia – Área de Fitopatologia

Aprovada em: 20/07/2007
Pela Comissão Examinadora

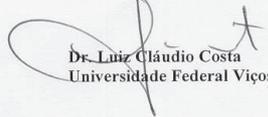

Dr. José Mauricio Cunha Fernandes
Presidente da Comissão Examinadora
Orientador


Dr. Pedro Alexandre Varella Escosteguy
Coord. Prog. Pós-Graduação em Agronomia


Dr. Wilson Antônio Klein
Universidade de Passo Fundo


Dr. Mauro Antônio Rizzardi
Diretor FAMV


Dr. Carlos Alberto Forcelini
Universidade de Passo Fundo


Dr. Luiz Cláudio Costa
Universidade Federal Viçosa


Dr. Clyde Fraisse
University Florida - USA

CIP – Catalogação na Publicação

- P337t Pavan, Willingthon
Técnicas de engenharia de software aplicadas à modelagem e simulação de doenças de plantas / Willingthon Pavan. -- 2007.
182 f. : il. color. ; 29 cm.
- Tese (Doutorado em Agronomia) – Universidade de Passo Fundo, 2007.
Orientador: Dr. José Maurício Cunha Fernandes
1. Trigo. 2. Fitopatologia – Simulação por computador. 3. Modelagem – Software. I. Fernades, José Maurício Cunha, (orient.). II. Título.

CDU: 632.1:004.41

Bibliotecária Ana Paula Benetti Machado CRB 10/1641

DEDICATÓRIA

À minha lindinha, Williane, por ter cedido três anos e meio da sua infância, me dando forças com suas “cartinhas”, carinhos e doces beijinhos. Te amo muito filha.

À minha esposa Fabiane, pelo amor, força, coragem e incentivo. Amo você.

Aos meus pais, Zilio e Maria Elena (Tica), pela vida, carinho, alegria, exemplos de vida e apoio sempre presente nestes longos anos da minha vida.

AGRADECIMENTOS

Ao meu orientador José Maurício Cunha Fernandes, grande amigo, por ter acreditado, incentivado e dedicado longos períodos do seu tempo na orientação desse trabalho.

Ao Clyde William Fraisse, pelo profissionalismo, amizade e acolhimento na fantástica experiência de intercâmbio realizado na Universidade da Flórida.

Aos meus irmãos Kamilla e Rafael, pela alegria, amizade e oportunidade de estarmos sempre juntos.

Aos meus avós, pelo carinho e dedicação.

À Zara e ao Laércio, pelo apoio e carinho.

Aos familiares, por terem acreditado e apoiado nos momentos difíceis.

Ao amigo Tony Hunt da Universidade de Guelph, pela alegria, disponibilidade e gentileza de ter cedido o código fonte do simulador CropSim-Wheat.

Aos amigos Emerson Del Ponte e Paulo Pires pelas informações compartilhadas e ajuda em diversos momentos desta jornada.

À equipe do projeto de pesquisa Mosaico/Simuplan da Universidade de Passo Fundo, pela amizade, auxílio e companheirismo.

Aos professores do programa de Pós-Graduação em Agronomia da Universidade de Passo Fundo, pelo exemplo profissional, excelentes aulas e exemplo de vida.

À equipe do Dr. Jones, Universidade da Flórida, pela amizade e auxílio na condução dos trabalhos. Ao Dr. Jones, meu sinceros agradecimentos.

Aos funcionários e todos os colegas do programa, pelas amizades, auxílios, realizações e momentos de lazer.

Aos pesquisadores e funcionários da Embrapa Trigo, pela oportunidade de ter convivido neste maravilhoso centro de conhecimento.

Aos grandes amigos, que souberam entender a ausência e dar apoio nos momentos difíceis.

Aos meus alunos, por terem compartilhado do entusiasmo provocado por este trabalho.

À Deus, por estar sempre presente.

SUMÁRIO

LISTA DE FIGURAS	xiii
LISTA DE TABELAS	xix
RESUMO	xxi
ABSTRACT	xxiii
1 INTRODUÇÃO	1
2 REVISÃO DE LITERATURA	7
2.1 Trigo	7
2.2 Doenças	8
2.3 Progresso de doenças de plantas	11
2.3.1 A proposta de Vanderplank	12
2.3.2 Análise do progresso de doenças	15
2.4 Modelos de Simulação	16
2.4.1 APSIM	19
2.4.2 Daisy	20
2.4.3 Century	22
2.4.4 DSSAT	25

2.5	Modelos de Simulação de Epidemias	31
2.6	Ciclo de Vida de um Modelo de Simulação	40

CAPÍTULO I

PADRÕES DE PROJETO APLICADOS A MODELOS DE SIMULAÇÃO DO CRESCIMENTO E DESENVOLVIMENTO DAS CULTURAS	43
Introdução	45
Material e Métodos	50
DSSAT	50
R	53
Java	54
Rserve	56
Banco de dados	57
Infra-estrutura	57
MVC	58
Resultados	60
SGDC	63
SBD	65
SMPD	66
SW	67
MSC	68

Discussão	71
---------------------	----

CAPÍTULO II

DESENVOLVIMENTO DE MODELOS DE SIMULAÇÃO DE DOENÇAS UTILIZANDO UML E ORIENTAÇÃO A OBJETOS 73

Introdução	75
Teoria e Metodologia	79
Características do Ciclo de Vida do Patógeno	79
Modelo de crescimento e desenvolvimento	84
O modelo	91
Resultados	105
Discussão	118

CAPÍTULO III

SISALERT - UMA PROPOSTA METODOLÓGICA PARA O AUXÍLIO À TOMADA DE DECISÃO NO MANEJO DE DOENÇAS 125

Introdução	128
Material e Métodos	131
Plataforma	131
A Estrutura	132
A Implementação	134

Resultados	140
Discussão	152
3 CONSIDERAÇÕES FINAIS	159
REFERÊNCIAS BIBLIOGRÁFICAS	161

LISTA DE FIGURAS

1	Componentes de uma coluna do modelo Daisy (Adaptado de Abrahamsen e Hansen (2000))	23
2	Visão geral do modelo Century (Adaptado de Century (2007))	24
3	Estrutura do Modelo DSSAT (Adaptado de Jones et al. (2003))	27
4	Auto e allo-deposição (Traduzido de Willocquet e Savary (2004))	38
5	Ciclo de vida de um modelo de simulação (Adaptado de Chwif (1999))	42
6	Estrutura do Modelo DSSAT (Adaptado de Jones et al. (2003))	52
7	Modelo de simulação de doenças de plantas (MSP)	61
8	Interface implementada pelas classes do sistema	63
9	Servidor de Gerenciamento de Dados Climáticos (SGDC)	65
10	Integração R x CropSim-Wheat	69
11	Integração Modelo x RServe	70
12	Executando código R na linguagem Java.	70
13	Estrutura do Modelo DSSAT (Adaptado de Jones et al. (2003))	86
14	Estrutura do Modelo Genérico de Doenças.	92

15	Conexão entre o modelo genérico de doença com o CropSim-Wheat.	95
16	Controle da simulação pela interface com o usuário.	96
17	Interface Weather e suas fontes de dados.	97
18	Auto e allo deposição (Adaptado de Willocquet e Savary (2004))	99
19	Alimentação e contribuição entre as nuvens	100
20	Fluxograma de funcionamento do modelo genérico	107
21	Tela da Interface visual do simulador genérico de doenças	108
22	Acúmulo de favorabilidade para liberação do inóculo inicial	108
23	Funções para cálculo da favorabilidade da temperatura e molhamento	111
24	Cálculo de novas lesões	112
25	Cálculo da expansão das lesões (área visível e invisível) no período latente	113
26	Cálculo da expansão das lesões (área visível e invisível) no período infeccioso	113
27	Produção de novos esporos pelo <i>cohort</i>	114
28	Cálculo do fator de interferência com relação a área doente	114
29	Interferência da idade do <i>cohort</i> na produção de esporos	114

- 30 Resultado da modificação dos parâmetros de transferência de inóculo para/entre as nuvens, onde "FB" significa folha-bandeira; "pto", a proporção de de esporos transferidos para a nuvem da folha (*Organ*); "ptp", a proporção de esporos transferidos para a nuvem da planta (*Plant*) e "ptf", a proporção de esporos transferidos para a nuvem do campo (*field*). 116
- 31 Comparação entre valores observados e simulados para três folhas de trigo (*Organ 9*: Folha-bandeira, *Organ 8*: Folha-bandeira -1 e *Organ 7*: Folha-bandeira -2) . . 117
- 32 Severidade simulada para os anos de 1979 a 2006 . . . 117
- 33 Estrutura de implementação do sistema SISALERT . . 132
- 34 Tabela que armazena os dados das estações meteorológicas automáticas com suas coordenadas geográficas e município onde estão instaladas. 136
- 35 Tabela que armazena dados dos municípios abrangidos pelo sistema. 136
- 36 Tabela que armazena prognósticos climáticos previamente processados. 136
- 37 Tabela que armazena informações climáticas coletados pelas estações meteorológicas. 137
- 38 Tabela que armazena resultados do modelos de simulações. 137
- 39 Topologia de servidores utilizado 138
- 40 Estações meteorológicas e áreas de abrangencia. . . . 139

41	Fluxograma de funcionamento do SISALERT	141
42	Fluxograma de funcionamento do SISALERT	142
43	Obtenção de prognósticos climáticos do INPE/CPTEC	143
44	Obtenção de dados das estações meteorológicas auto- máticas	144
45	Sistema de auxílio à tomada de decisões: seleção de data de espigamento e município	145
46	Sistema de auxílio à tomada de decisões: visualização dos dados simulados e climáticos em forma de gráficos	145
47	Método <i>rate</i> (taxas): verificação da ligação com o CropSim-Wheat e cálculo de novas espigas	147
48	Método <i>integration</i> (integração) de Plant: invoca a in- tegração dos demais módulos e cria novo cohort de es- pigas	148
49	Método <i>rate</i> (taxas): cálculo de novas anteras	149
50	Método para remoção de um <i>cohort</i> de anteras	149
51	Métodos <i>rate</i> (taxas) e <i>integration</i> (integração) da classe Inóculo: cálculo da nuvem de esporos, atuali- zação e integração	150
52	Cálculo da velocidade da infecção baseada na tempe- ratura (<i>inf</i>)	150
53	Método <i>integration</i> (integração): cálculo da giberela diária (<i>dailyGib</i>)	151
54	Cálculo da proporção de anteras (<i>st</i>)	151

55 Cálculo da severidade a cada rodada (*severity*) 152

LISTA DE TABELAS

1	Parâmetros utilizados no simulador da ferrugem da folha do trigo (parte I)	109
2	Parâmetros utilizados no simulador da ferrugem da folha do trigo (parte II)	110
3	Parâmetros de configuração da planta e do simulador .	111
4	Parâmetros de configuração do SISALERT	146

RESUMO

TÉCNICAS DE ENGENHARIA DE SOFTWARE APLICADAS À MODELAGEM E SIMULAÇÃO DE DOENÇAS DE PLANTAS

WILLINGTHON PAVAN¹

RESUMO - Diante do aumento na demanda por alimentos, ameaças ao ambiente, aumento da pressão sobre os recursos terra/água e a globalização dos mercados, muitos sistemas de produção agrícola ao redor do mundo estão rapidamente sendo modificados. Novas metodologias surgem para avaliar as opções de desenvolvimento, os impactos das mudanças nos sistemas de produção e a estabilidade e sustentabilidade, de modo que decisões possam ser tomadas a fim de alcançar os objetivos. Para isso faz-se necessária a integração do conhecimento de diferentes disciplinas. Nesse sentido, o desenvolvimento de modelos de simulação apresenta-se como alternativa viável para pesquisa dos problemas que envolvem o setor agrícola. O desenvolvimento desses modelos requer uma análise profunda do sistema em estudo, sendo crucial o emprego de métodos e ferramentas de engenharia de software. Nas últimas décadas, a modelagem matemática de doenças de plantas, por exemplo, tem se expandido muito rapidamente, contribuindo para o manejo integrado. Os recentes avanços na ciência da computação têm auxiliado nesse desenvolvimento, trazendo vantagens e facilidades operacionais para a construção de modelos que representem os processos complexos. A modularidade e a forma genérica são os termos que descrevem a nova e amplamente aceita metodologia para superar as complexidades no desenvolvimento

¹Doutor em Agronomia, Universidade de Passo Fundo - pavan@upf.br

e reúso de modelos. Este trabalho teve como objetivo aplicar padrões de projeto a modelos de simulação do crescimento e desenvolvimento das culturas, utilizando-se principalmente do padrão MVC (Model-View-Controller), desenvolver um modelo de simulação de doenças de plantas, genérico e parametrizável, utilizando-se da UML e linguagens de programação orientadas a objetos, e aplicar a estrutura projetada para o desenvolvimento de uma proposta metodológica para o auxílio à tomada de decisão no manejo de doenças de plantas (SISALERT). A aplicação das técnicas de engenharia de software mostrou-se apropriada e robusta para orientar no desenvolvimento de modelos de simulação. A união das tecnologias estudadas comprovou a possibilidade do reaproveitamento do conhecimento legado, visto que a linguagem de programação R permitiu realizar a comunicação entre Fortran e Java. Com o modelo de simulação do patossistema *Puccinia*-Trigo, pôde-se acompanhar a evolução do crescimento da área foliar do trigo, juntamente com o progresso da ferrugem da folha, mostrando-se como uma excelente alternativa para o estudo das relações patógeno/hospedeiro e para a quantificação de danos. Com o desenvolvimento da plataforma SISALERT, demonstrou-se a viabilidade da aplicação prática em larga escala para o auxílio à tomada de decisões, podendo ser expandida para outros usos/culturas.

ABSTRACT**SOFTWARE ENGINEERING TECHNIQUES
APPLIED TO MODELING AND SIMULATION OF
PLANT DISEASES****WILLINGTHON PAVAN¹**

ABSTRACT - Faced with increased demands for food by a growing population, threats to the environment due to increasing pressures on land and water resources, and globalization of markets and trade, many agricultural production systems around the world are rapidly changing. Methods are needed to evaluate options for development, impacts of changes in production systems, and stability and sustainability of proposed changes in practices so that decisions can be made to achieve societal goals. A systems approach, integrating knowledge from different disciplines, is necessary. In this direction, simulation models are presented as a research tool for addressing problems involving the agricultural sector. The development of these models requires a deep analysis of the system in study, being crucial the choice of methods and tools of software engineering. In the last few decades, the mathematical modeling of plant diseases, for example, has expanded very quickly, contributing for disease integrated management. The recent advances in the computer science have assisted in this development, bringing operational advantages and usefulness for the construction of models representing complex processes. The modularity and the generic form are terms that describe the new and widely accepted methodology to surpass the complexities in the development and re-use of models. This work had as objective to

¹Doutor em Agronomia, Universidade de Passo Fundo - pavan@upf.br

apply project patterns towards the development of crop simulation models, using MVC (Model-View-Controller), to develop a generic plant disease model, using UML and object oriented programming languages, and to apply the proposed architecture for the development of a novel approach for delivering decision supporting to plant disease management (SISALERT). The application of software engineering techniques revealed to be appropriate and robust to guide in the development of plant disease simulation models. Combining technologies proved to be possible to use existing knowledge legacy. The programming language R, allowed to carry through the communication between Fortran and Java. The simulation model of the pathosystem *Puccinia*-Wheat reasonable well, representing leaf rust progress accordingly to the wheat leaf area growth. It can be used as a tool for studying pathogen/host relationship and for damage quantification. The platform SISALERT, showed to be feasible the application of decision support systems on a large scale. Moreover, it was demonstrated that it can be expanded for other uses/crops.

1 INTRODUÇÃO

Diante do aumento na demanda por alimentos, de ameaças ao ambiente, do aumento da pressão sobre os recursos terra/água e a globalização dos mercados, muitos sistemas de produção agrícola no mundo estão rapidamente sendo modificados.

Produtores, indústrias e autoridades devem responder rapidamente a esses eventos globais, cada vez mais complexos, assim como às condições agroecológicas e socioeconômicas locais. A pesquisa agrícola é uma fonte de informação muito importante para os tomadores de decisões, entretanto é muito difícil para as técnicas tradicionais de pesquisa fornecer as informações necessárias aos tomadores de decisão nas situações complexas encontradas atualmente.

Metodologias são necessárias para avaliar as opções de desenvolvimento, os impactos das mudanças nos sistemas de produção e a estabilidade e sustentabilidade das mudanças propostas, de modo que decisões possam ser tomadas a fim de alcançar os objetivos sociais. Para isso, faz-se necessária a integração do conhecimento de diferentes disciplinas. Estes métodos focam numa cuidadosa definição do problema ou questão a ser respondida, integração do conhecimento e informação, utilizando modelos e outras ferramentas, com o intuito de analisar os diferentes cenários e opções, a fim de ajudar os responsáveis pelas tomadas de decisões a conhecerem as conseqüências e os riscos prováveis de suas decisões.

Entre as diversas culturas existentes, o trigo (*Triticum aestivum*

L.) destaca-se como um dos alimentos mais importantes da humanidade, sendo cultivado nos quatro cantos do mundo e possuindo uma grande relevância econômica e social. A produtividade do trigo é muito variada, sendo afetada por muitos fatores, como as condições do solo, do clima, do manejo, entre outras. Em alguns locais, como o Brasil, a ocorrência de elevado número de doenças é um dos fatores mais importantes que contribuem significativamente para a redução do rendimento e da qualidade dos grãos. No Brasil, a maior parte do cultivo do trigo localiza-se na região Sul, abrangendo os estados do Paraná, Rio Grande do Sul e Santa Catarina. Também é nesses estados que a cultura está sujeita às maiores adversidades climáticas, associadas à influência de fenômenos de grande escala, como o ENOS (El Niño Oscilação Sul).

A presença das fases do ENSO (EL Niño e La Niña) altera significativamente a variabilidade interanual dos elementos meteorológicos. Em anos de ocorrência de El Niño, observam-se precipitações pluviométricas acima da média histórica para o sul do Brasil, principalmente na primavera e no início do verão (período de maturação e colheita do trigo), e em anos de La Niña a tendência é que ocorram estiagens no mesmo período para a região (BERGAMIN FILHO et al., 1995a; BERLATO; FONTANA, 2003; BERLATO et al., 2005).

Além dos fatores climáticos, a alegada falta de competitividade na cultura nos sistemas de produção tem sido uma das maiores dificuldades para que o Brasil se firme, de fato, como um país produtor de trigo. Os problemas são atinentes a aspectos da produção nacional, mas também têm relação com a importação de trigo de países que oferecem subsídios aos seus produtores e à cadeia de comercialização de trigo.

A melhoria dos sistemas de produção de trigo no Brasil não é uma tarefa fácil. Trata-se de um sistema de produção muito complexo. Interações entre fatores do ambiente físico e de ordem socioeconômica acabam determinando o resultado final e condicionando o desempenho das explorações com esse cereal em terras brasileiras. Em essência, os problemas são de natureza sistêmica, não disciplinar, e, como tal, sua solução requer também o uso de um enfoque de pesquisa transdisciplinar. Por essa característica, o mais adequado parece ser o uso do enfoque sistêmico, via técnicas de modelagem e simulação e de teorias de decisão, integrando conhecimentos de diversas áreas e analisando possibilidades que possam otimizar a produção de trigo no Brasil, quer seja no planejamento agrícola, quer seja no manejo da cultura.

Conforme destacado, o sistema de produção de trigo no Brasil é muito complexo, incluindo interações entre fatores bióticos e abióticos. Alguns desses podem ser modificados via manejo da cultura, entretanto o clima ainda exerce a maior influência sobre o rendimento final de trigo. Um dos grandes desafios da ciência agrônoma é a análise objetiva de um conjunto de fatores que atuam simultaneamente.

Os modelos de simulação de culturas podem auxiliar no entendimento desses fatores, assim como indicar alternativas no manejo da cultura perante diferentes cenários para o sistema de produção. Define-se que um sistema é uma parte de um todo, constituído por uma série de componentes e suas inter-relações. Um modelo de sistema agrícola deve ter uma estrutura hierárquica composta de vários subsistemas (módulos) de definição autônoma e cujo exercício funcional deve fornecer uma descrição compreensiva e quantitativa do sistema de produção (FERNANDES et al., 1994).

Os modelos de simulação há mais de sessenta anos têm auxiliado a

abordagem globalizante dos problemas que envolvem o setor agrícola. Tal modo de observação da realidade, integrado e multidisciplinar, constitui o atributo fundamental do que, usualmente, se designa por investigação de sistemas agrícolas (BRAGA et al., 2005).

Apesar da existência de um grande número de modelos para a cultura do trigo, poucos reúnem, ao mesmo tempo, os efeitos das condições climáticas, das ocorrências de secas, pragas e doenças, das práticas de manejo da cultura e da fertilidade do solo, sobre o comportamento agrônômico e fisiológico da cultura. Isso ocorre em razão da complexidade envolvida e da falta de padronização nos desenvolvimentos dos modelos pelos cientistas. Modelos complexos, essencialmente mecanísticos, do tipo de simulação do crescimento e desenvolvimento de culturas ligados à ocorrência e evolução de pragas, doenças e plantas invasoras, podem servir para orientar decisões no manejo desses fatores, incluindo aspectos como a probabilidade de ocorrência, o curso e o risco de perdas na produção.

O acúmulo do conhecimento sobre o complexo sistema de produção, aliado às informações meteorológicas cada vez mais presentes, aos prognósticos climáticos e ao entendimento do efeito das mudanças climáticas, tem oferecido a oportunidade de aplicá-los em modelos de simulação a fim de obter as respostas de forma mais realística para as perguntas dos tomadores de decisões, como a amplitude de rendimento de uma determinada cultura e a severidade do ataque de uma doença diante de prognósticos climáticos.

Com os revolucionários avanços tecnológicos na informática e o surgimento da internet, com a conectividade global e a integração com linguagens modernas de programação, têm-se produzido novos conceitos e caminhos a serem ainda explorados na produção e trans-

ferência do conhecimento. Com a revolução provocada pelas tecnologias baseadas na Web, grandes mudanças no desenvolvimento e no emprego de sistemas de suporte à tomada de decisão para produtores e especialistas no manejo de doenças estão se tornando realidade (FERNANDES et al., 2007, 2007b).

Com a evolução ocorrida nas áreas agronômicas e na informática, este trabalho teve como objetivo aliar metodologias já consagradas, como os modelos de simulação, com os recentes avanços tecnológicos nas linguagens de programação, projetos de software, dispositivos móveis, internet e a pronta disponibilização de informações climatológicas em prol dos avanços tão necessários para manter o crescimento da humanidade (produção agrícola).

Na presente tese, os estudos desenvolvidos foram divididos numa revisão da literatura e três artigos. Na revisão da literatura são apresentados os conceitos sobre o patossistema escolhido e os avanços da modelagem ao longo do tempo. No primeiro artigo são apresentados padrões de projeto orientados a objetos que podem ser aplicados ao desenvolvimento de modelos de simulação de doenças de plantas e o crescimento e desenvolvimento das culturas. No segundo artigo é apresentada a descrição de um modelo genérico capaz de representar epidemias, utilizando-se do patossistema *Puccinia*-Trigo como um caso especial do modelo genérico, acoplado a um modelo de crescimento do trigo, com a avaliação da dinâmica do modelo de epidemia da ferrugem da folha. Por fim, no terceiro artigo é apresentada uma metodologia para o desenvolvimento de sistemas de auxílio à tomada de decisões, capazes de reaproveitar o conhecimento legado, aliando-se a modernas tecnologias computacionais, visando a um adequado manejo de doenças.

2 REVISÃO DE LITERATURA

2.1 Trigo

O trigo (*Triticum aestivum* L.) é um cereal originário das regiões semiáridas e temperadas da Mesopotâmia e Europa, cujo cultivo teve início nas lavouras primitivas do sudoeste da Ásia de 7000 a 9000 a.C. (BELL, 1987). É um cereal de grande importância, sendo fonte de 20% do total de calorias consumidas pela humanidade (ANGRA, 1995). Possui um destaque mundial, sendo a segunda cultura em produção de grãos, superada apenas pelo milho (TRIGO, 2004).

O cultivo do trigo, no Brasil, teve início no ano de 1534, com a vinda dos colonizadores europeus, na Capitania de São Vicente. Contudo, é limitado por uma série de fatores, entre eles, as condições de clima, solo, pragas e doenças, formando um complexo que muitas vezes impossibilita o rendimento econômico em determinadas áreas tritícolas (PRATES; FERNANDES, 2001).

A produção comercial do trigo no estado do Rio Grande do Sul iniciou a partir de 1950, com o processo de modernização da agricultura, com a incorporação de máquinas e equipamentos agrícolas, em razão dos fortes subsídios governamentais naquela época. Após, o processo de produção avançou para os demais estados do Brasil, iniciando pelo oeste de Santa Catarina, seguido pelo Paraná, algumas regiões de São Paulo e, mais recentemente, para o Centro-Oeste brasileiro (BRUM; HECK, 2005).

Graças à atual tecnologia e à grande variedade de cultivares existentes, o trigo pode ser cultivado nas mais variadas condições climáticas, podendo, no Brasil, ser cultivado com sucesso desde a região Sul até o Cerrado. Na região do Brasil Central, dependendo da época do ano, duas opções de cultivo estão disponíveis: o cultivo de sequeiro e o cultivo irrigado (ALVES, 2005).

Apesar dos esforços das pesquisas, dos reais avanços conquistados e da produção de apenas 50% das necessidades do consumo nacional, dificuldades ainda são encontradas pelos produtores de trigo no que diz respeito ao aumento da produção, dentre as quais estão o elevado custo de produção, a comercialização das safras e a incidência de um grande número de doenças (BRUM; HECK, 2005).

2.2 Doenças

O trigo é uma cultura afetada por um grande número de doenças, as quais estão entre os principais fatores limitantes da produção, podendo causar prejuízos consideráveis, principalmente quando as condições climáticas são favoráveis ao desenvolvimento de epidemias. Os prejuízos podem ser agravados quando várias doenças incidem simultaneamente na lavoura. Diversos são os fatores que interferem no estabelecimento e no desenvolvimento de epidemias, dentre os quais estão as condições ambientais, a suscetibilidade dos cultivares, a agressividade dos patógenos e a época do início da infecção (BARROS et al., 2006).

Entre as doenças que ocorrem no trigo destacam-se aquelas causadas por fungos, como o oídio, causado por *Blumeria* (sin. *Erysiphe*) *graminis* (DC) Speer f.sp. *tritici* Em. Marshall (anamorfo: *Oidium*

monilioides (Nees.) Link.); a ferrugem da folha, causada por *Puccinia recondita* Rob.ex. Desm. f.sp. *tritici* Heriks. & Henn; a ferrugem do colmo, causada por *Puccinia graminis* Pers f.sp. *tritici* Heriks. & Henn; a mancha da gluma, causada por *Phaeosphaeria nodorum* (anamorfo: *Stagonospora nodorum* (Berk.) Cast & Germ.); a mancha marrom, causada por *Cochliobolus sativus* (Ito & Kurib.), Drechs. ex. Dastur. (anamorfo: *Bipolaris sorokiniana* Sacc. in. Sorok.); a mancha amarela, causada por *Pyrenophora tritici-repentis* (Died) Drechs. (anamorfo: *Drechslera tritici-repentis* (Died.) Schoem.); a giberela, causada por *Gibberella zeae* (Schw.) Petch (anamorfo: *Fusarium graminearum* (DC. ex. Mérat)); a bruzone, causada por *Magnoportha grisea* (Herbert) Yaegashi & Udagawa (*Pyricularia grisea* (Cooke) Sacc.); o carvão, causado por *Ustilago tritici* (Pers.), e o mal-do-pé, causado por *Gaeümannomyces graminis* var. *tritici* (Sacc.) Arx. & Oliv (REIS et al., 2001; BARROS et al., 2006).

As doenças de origem bacteriana que infectam o trigo são a mancha estriada causada por *Xanthomonas campestris* pv. *undulosa* e a queima da folha, induzida por *Pseudomonas syringae* pv. *Syringae*. As principais viroses são *Barley Yellow Dwarf Virus* (BYDV) e *Wheat Soil Borne Mosaic Virus* (WSBMV) (PICININI, 1990).

Para um bom manejo das doenças do trigo, evitando perdas na cultura, devem-se adotar, simultaneamente, várias práticas integradas, como a utilização de cultivares resistentes e/ou precoces, genes de resistência ou tolerância, o plantio em época adequada, práticas culturais e controle químico, sendo recomendado, quando se têm culturas bem conduzidas, alto potencial produtivo, ou se necessite garantir a produtividade (BARROS et al., 2006).

As doenças fúngicas têm sido uma das principais causas de frus-

tração na safra do trigo no Brasil, induzindo a elevadas perdas no rendimento e na qualidade de grãos (PICININI et al., 1996). O controle químico das moléstias aumenta os custos de produção no que se refere à compra dos produtos, à maquinaria e à mão-de-obra para aplicação na lavoura, além de envolver riscos para a saúde humana e animal e para o equilíbrio ecológico.

O elevado número e frequência de ocorrência de doenças causadas por fungos levam a que os fungicidas assumam um papel importante na cultura do trigo no sul do Brasil. O uso de fungicidas é justificado na combinação de rendimentos elevados e alta pressão de infecção, ambos resultantes de um ambiente com umidade adequada, adubação nitrogenada e sistema de produção. Os fungicidas aplicados na parte aérea do trigo, na maioria das vezes, aumentam o peso do grão e o rendimento e também estendem a duração do dossel. Grãos bem formados resultam em pesos específicos elevados (BAYLES, 1977), um componente importante da qualidade. Grãos enrugados, por sua vez, podem contribuir para o maior teor de impurezas, menor taxa de extração de farinha e redução dos índices da energia metabolizável (GOODING et al., 2000).

O volume de perdas é variável de ano para ano, dependendo das condições climáticas. O fungicida propiconazole, por exemplo, utilizado em duas aplicações, no estágio de emborrachamento e na floração plena, num período de 12 anos de experimentação em Passo Fundo, RS, mostrou dano médio de 44,61%, equivalente a 1.152 kg de trigo por hectare, atribuída ao inadequado controle das doenças fúngicas (PICININI et al., 1996). O desenvolvimento das doenças foliares interfere diretamente no processo de fotossíntese, além de alterar o metabolismo das plantas, influenciando, assim, o rendimento e a qualidade dos grãos de trigo (GOODING et al., 2000).

O desenvolvimento e a implementação de medidas de controle na cultura do trigo requerem informações precisas e acuradas sobre o dano causado pelos patógenos. Waggoner e Berger (1987) afirmam que a lógica da relação entre severidade de doença e dano é incerta, pois severidade de doença, considerada isoladamente, tem efeito diferente caso ocorra precoce ou tardiamente numa planta, seja porque a desfolha não é considerada, seja porque a área foliar das plantas é ignorada, ou, ainda, porque a fisiologia nos ensina que a produtividade depende de área sadia, verde, fotossintetizante das folhas (CHARLES-EDWARDS, 1982), não da área doente, necrosada, depauperada, ocupada pelo patógeno.

Estimar danos por meio de observações de doenças é extremamente difícil. A severidade dos sintomas visuais pode ser resultante de diferentes alterações na fisiologia da planta; o efeito depende do que aconteceu com a planta no passado, do que está acontecendo no presente e do que vai acontecer no futuro. Portanto, uma estratégia confiável de prever danos é o emprego da simulação que usa modelos matemáticos de crescimento da planta em relação ao ambiente em que ela se encontra e, também, simula as alterações produzidas pelo patógeno no crescimento, na fisiologia, e as conseqüências no rendimento (SHAW, 1997).

2.3 Progresso de doenças de plantas

Tendo em vista que o progresso de doenças é freqüentemente a manifestação mais facilmente visualizada, a dinâmica temporal das doenças de plantas tem sido mais enfatizada no estudo de epidemias (AZEVEDO et al., 2002).

No estudo de epidemias, a coleta de dados é a fase mais importante para se entender o seu funcionamento. A seqüência de dados obtidos em intervalos regulares de tempo durante um período específico é conhecida por série temporal, também denominada de série histórica. Segundo Latorre e Cardoso (2001), uma série histórica pode ser composta por três componentes não observáveis: tendência (T), sazonalidade (S) e a variação aleatória denominada de ruído branco (a). Na elaboração de um modelo, a primeira escolha seria um relacionamento aditivo desses componentes: $Z_t = T_t + S_t + a_t$ (LATORRE; CARDOSO, 2001).

2.3.1 A proposta de Vanderplank

Segundo Campbell e Madden (1990), a análise de Vanderplank foi baseada largamente nas taxas de infecção e no relacionamento entre a quantidade de inóculo e a quantidade de doença produzida por ela. A base do sistema foi formada por dois modelos, o logístico e o monomolecular, embora algumas referências tenham sido feitas ao modelo exponencial. Os modelos foram aplicados como modelos biológicos, ou seja, as suposições biológicas específicas foram feitas sobre os patossistemas a que os modelos foram aplicados (CAMPBELL; MADDEN, 1990).

Com o propósito de ilustrar, o aumento da doença foi considerado análogo à acumulação de juros monetários (simples e composto). Em juros simples, o juro é calculado apenas sobre o capital, sem ganhos sobre o juro obtido, ao passo que em juros compostos é calculado sobre o valor do capital mais o valor dos juros. Dessa forma, as doenças poderiam ser classificadas como: doenças de juros simples (doenças monocíclicas) e doenças de juros compostos (doenças policíclicas).

Nas doenças monocíclicas, as plantas infectadas durante o ciclo da cultura não servirão de fonte de inóculo para novas infecções durante o mesmo ciclo; nas doenças policíclicas, as plantas infectadas servirão de fonte de inóculo para novas infecções durante o mesmo ciclo.

No caso de doenças de juro simples (monocíclicas), não há uma relação entre velocidade de aumento da doença e quantidade de doença em cada instante, sendo o aumento gradativo do número de plantas doentes durante o ciclo da cultura função do inóculo original previamente existente (MICHEREFF, 2006).

Para o caso das doenças de juro compostos (policíclicas), considera-se que há uma relação entre velocidade de aumento da doença e quantidade de doença em cada instante, ou seja, plantas doentes (ou lesões) dão origem a novas plantas doentes (ou novas lesões) no mesmo ciclo da cultura. Segundo Michereff (2006), a velocidade de aumento da doença é proporcional à própria quantidade de doença em cada instante. Assim, se uma lesão der origem a 10 lesões, 10 lesões darão origem a 100, 100 a 1000, 1000 a 10.000, e assim por diante.

Segundo Campbell e Madden (1990), a analogia entre juro e doenças de plantas é quebrada em pelo menos quatro formas:

- infecções ocorrem intermitentemente, não continuamente;
- há um limite para a quantidade de tecido hospedeiro disponível e portanto para a quantidade de doença;
- novos tecidos infectados não se tornam imediatamente infectivos;
- doenças freqüentemente ocorrem em focos.

Com isso, a analogia do dinheiro serve como um intuitivo ponto inicial, embora não inteiramente realístico, ajudando a estabelecer uma base para o uso de modelos específicos a fim de descrever o progresso da doença descrita.

A distinção entre doenças monocíclicas e policíclicas é de uma importância primordial na análise de Vanderplank. Com as doenças policíclicas, o patógeno pode multiplicar-se através de sucessivas gerações no curso da epidemia. Como exemplo desta doença pode-se citar a ferrugem do trigo, agente causal *Puccinia graminis* f. sp. *tritici*, na qual a epidemia começa com apenas poucas pústulas, que a cada 10 dias podem produzir uma nova geração. Para este tipo de doença Vanderplank propôs a forma linearizada do modelo logístico $\ln[y/(1-y)]$, sendo y a proporção da doença (incidência ou severidade) (CAMPBELL; MADDEN, 1990).

Para as doenças monocíclicas, há um aumento da doença sem a multiplicação do patógeno, o qual não se move de planta para planta, e a produção de inóculo durante a estação de crescimento não se move para infectar tecidos e causar infecções, ou seja, as plantas infectadas durante o ciclo da cultura não servirão de fonte de inóculo para novas infecções durante o mesmo ciclo. Esse é um caso típico da murcha do algodoeiro, na qual o agente causal, *Fusarium axysporum* f. sp. *vasinfectum*, coloniza principalmente o interior do xilema das plantas infectadas. Para este tipo de doença, Vanderplank sugeriu $\ln[1/(1-y)]$ contra o tempo e propôs o uso de modelos monomoleculares para o aumento da doença, o que implica que o progresso da doença com o tempo não é logarítmico, permitindo a diminuição proporcional, $1-y$, do tecido disponível para a infecção (CAMPBELL; MADDEN, 1990).

A descrição do progresso da doença foi dividida em duas categorias

biológicas: doenças com efetiva multiplicação do inóculo durante a estação e doenças sem a efetiva multiplicação do inóculo.

2.3.2 Análise do progresso de doenças

Processos dinâmicos, incluindo a mudança na quantidade da intensidade da doença numa população de plantas sobre o tempo, são definidas pelas taxas da mudança sobre o tempo (CAMPBELL; MADDEN, 1990). Para comparar e prever o desenvolvimento de doenças é necessário quantificar ou modelar matematicamente mudanças no desenvolvimento através do tempo (BOWEN, 1997). Muitos modelos de crescimento, voltados para a biologia populacional, têm sido utilizados para modelar o desenvolvimento de doenças em populações de plantas.

Considerando que y é a quantidade de doença medida como severidade ou incidência, uma epidemia pode ser descrita em termos de dy/dt , o qual representa a taxa absoluta do aumento da doença ou a taxa de crescimento absoluto. Para epidemias com uma fase regressiva, dy/dt pode também representar a taxa absoluta da diminuição da doença (CAMPBELL; MADDEN, 1990).

A curva do progresso da doença, em relação à descrição e análise, tem sido vista e discutida extensivamente nas últimas décadas. A análise dos dados do progresso da doença consiste largamente no uso de métodos e modelos adaptados do vasto campo da análise da curva de crescimento e dinâmica de populações, além da adaptação de modelos e métodos especialmente para epidemias de doenças de plantas.

Entre os modelos mais utilizados na análise do progresso da do-

ença estão os modelos Exponencial, Monomolecular, Logístico, Gompertz, Richards, Weibull e os polinomiais. Esses modelos, por alguma necessidade, podem ser modificados para se ajustar à curva da doença estudada.

2.4 Modelos de Simulação

A idéia de representar os fenômenos da natureza por meio de simuladores é um dos antigos sonhos do homem, que começou a se tornar realidade há mais de cinco décadas, quando do desenvolvimento das linguagens de programação. Na década de 1960, por exemplo, muitos esforços foram feitos para o desenvolvimento de uma linguagem chamada Simula (*Simulation language*), a qual, com características de orientação a objetos, tinha como objetivo principal o desenvolvimento de sistemas de simulação, modelados pela iteração de um grande número de objetos distintos (NYGAARD; DAHL, 1981). A partir de então, muitos esforços foram realizados e muitos resultados de sucesso obtidos.

Modelos de simulação de culturas capazes de prever o rendimento final têm sido estudados intensivamente em várias partes do mundo (HOOGENBOOM, 2000) e representam a simulação da dinâmica do crescimento das culturas através da integração numérica, com a ajuda de computadores. Em essência, esses modelos de simulação são programas de computadores que representam, matematicamente, o crescimento das plantas em relação ao ambiente (GRAVES et al., 2002). Essas aplicações variam desde a predição do rendimento final em nível de propriedade até escalas regionais e nacionais, maior parte dessas predições envolve prognósticos climáticos, sendo feitas antes do momento da semeadura; outras são obtidas durante o desenvolvi-

mento da cultura (BANNAYAN et al., 2003).

Os primeiros modelos de simulação de culturas foram desenvolvidos nos anos 60 e tinham como finalidade simular a interceptação da luz e a fotossíntese nas plantas (WIT, 1965). Com o tempo, os modelos de simulação tornaram-se mais complexos, envolvendo grande quantidade de dados para iniciar a simulação. Com a complexidade dos modelos também vieram as limitações, mas, mesmo assim, há forte evidência de que os modelos de simulação de culturas podem ter participação importante na pesquisa científica, na tomada de decisão e na educação (GRAVES et al., 2002).

Esses modelos de simulação variam desde equações empíricas para a previsão de rendimento até modelos complexos de desenvolvimento, crescimento e rendimento. Os modelos complexos consideram os princípios que regem os processos de desenvolvimento e crescimento e oferecem ampla aplicação.

Em virtude da enorme complexidade dos sistemas envolvidos, quando se desenvolvem modelos matemáticos que descrevem o desenvolvimento de culturas, procura-se fracioná-los a fim de facilitar o entendimento e melhor representar, com uma maior precisão, as suas várias partes (CHWIF, 1999). Para exemplificar, pode-se imaginar um sistema composto por diversos modelos de simulação, no qual cada um é responsável por determinadas tarefas, como:

- prognóstico de clima, movimentação global e mudanças e impactos climáticos;
- desenvolvimento de culturas (planta);
- aparecimento/ataque de pragas e doenças;

- determinação de rendimentos;
- dinâmica da água no solo, desenvolvimento de sistemas radiculares e absorção de nutrientes.

Quando se desenvolvem modelos simples e, conseqüentemente, menores, é mais fácil, no caso da detecção de problemas, descartá-los e começar novamente. Em sistemas complexos isso se dá inversamente, tendo-se muita resistência a abandonar todo o trabalho realizado, mesmo que se perceba o fracasso. Isso demonstra a importância da divisão do sistema em partes menores.

Atualmente, encontram-se na literatura diversos modelos de simulação desenvolvidos para a área agrícola, dentre os quais se verifica uma sobreposição de focos (modelos que fazem o mesmo que outros). Porém, em muitos casos isso é necessário em virtude das particularidades de cada cultura e região agrícola (VERHAGEN et al., 2001).

Modelos de simulação, compostos pelo sistema solo-planta-atmosfera, podem ser considerados como os métodos mais adequados e de mais baixo custo para pesquisas sobre grande variabilidade ambiental. Estes permitem a determinação dos riscos envolvidos na produção agrícola em função dos principais componentes de produção, utilizando séries climatológicas históricas ou sintéticas.

Graças à formação de grupos multidisciplinares e interinstitucionais no mundo inteiro, muitos modelos de simulação do crescimento e desenvolvimento de culturas foram desenvolvidos para os mais variados propósitos. Dentre os modelos desenvolvidos destacam-se o APSIM (Agricultural Production Systems sIMulator), o Daisy, o Century e o DSSAT (Decision Support System for Agrotechnology Transfer). Deste último foi adotado neste trabalho como modelo de crescimento

e desenvolvimento da planta o submodelo CropSim-Wheat.

2.4.1 APSIM

O APSIM, desenvolvido na Austrália, é um sistema que tem por objetivo auxiliar os agentes de assistência técnica e agricultores nas recomendações de cultivos (HAYMAN; EASDOWN, 2002). Segundo Keating et al. (2003), o APSIM é um *framework modelling* que permite que módulos individuais de componentes-chaves de sistemas agrícolas sejam plugados ao sistema. Todos os módulos são estruturados de forma que fiquem dispostos ao redor do núcleo do simulador. O APSIM inclui um grande número de módulos agrossilvopastoris, processos do solo (balanço de água, pH, erosão etc.) e de gerenciamento (controle).

A estrutura definida neste sistema possibilita que os módulos possam ser desenvolvidos e acoplados com uma grande facilidade, permitindo também uma boa manutenção no sistema como um todo, assim como a retirada e inclusão de módulos conforme a necessidade. O APSIM é resultado da necessidade de ferramentas que forneçam previsões exatas da produção da cultura em relação ao clima, genótipo, solo e fatores de gerenciamento direcionados ao gerenciamento de recursos a longo termo. É composto por módulos biofísicos, que simulam processos biológicos e físicos, módulos de gerenciamento, que permitem ao usuário especificar as regras gerenciais pretendidas que caracterizam o cenário simulado, e módulos para facilitar a entrada e saída de dados das simulações. No centro está o núcleo do sistema, o qual direciona o processo da simulação, controlando todas as mensagens que são trocadas pelos módulos (KEATING et al., 2003).

Muitas interfaces de usuário para a construção, teste e aplicação

de modelos, assim como interfaces e ferramentas associadas a bancos de dados para a visualização e futuras análises das saídas das simulações estão disponíveis para o APSIM. Ferramentas para o desenvolvimento, teste e documentação também podem ser encontradas no repositório disponível na Web, no endereço <http://www.apsim.info/apsim>.

Quanto à entrada de dados, são fornecidas as características do solo (módulos de solo), as informações climáticas diárias (módulos meteorológicos), as características da superfície do solo e definição de resíduos, as características do cultivar e informações sobre a gerência agrônômica. Como resultados do modelo para diferentes cenários (clima) podem-se citar as mudanças no rendimentos da colheita e pastagem, componentes de rendimento e perdas provocados pela erosão.

2.4.2 Daisy

O Daisy é um modelo agroecológico monodimensional que simula o crescimento de uma cultura, o balanço de água e de calor, o balanço da matéria orgânica, a dinâmica do amônio e do nitrato, baseado em informações sobre práticas gerenciais e dados climáticos. Permite a construção de complexos cenários para a simulação de diferentes estratégias de gerenciamento e rotação de culturas.

O modelo Daisy (solo-planta-atmosfera) foi desenvolvido na Royal Veterinary and Agricultural University, Dinamarca, durante a segunda metade da década de 1980. Desde então vem sendo melhorado e expandido pelo conhecimento obtido de resultados de pesquisas realizadas. Originalmente, o modelo havia sido escrito na linguagem de programação Fortran, sendo posteriormente reescrito na linguagem de programação C++, tendo sido projetado para facilitar a interação entre modelos (ABRAHAMSEN; HANSEN, 2000; HANSEN, 2002).

Além das características elencadas, encontra-se inclusive no modelo a simulação do destino de pesticidas, entre outros. Para a execução da simulação, são necessários dados climáticos diários, como radiação, temperatura do ar e precipitação. Além desses dados climáticos, podem ser informados também dados sobre umidade relativa e velocidade do vento.

Dentre diversos estudos e aplicação do modelo Daisy, exemplos nos quais o uso do modelo pode contribuir são (HANSEN, 2002):

- mudanças no rendimento em função da disponibilidade da água e do nitrogênio sob várias práticas de gerência agrícolas;
- otimizações da irrigação;
- lixiviação de nitrato de áreas agrícolas;
- lixiviação de pesticidas de áreas agrícolas.

A inicialização e parametrização do modelo são feitas por meio de arquivos de configuração. A entrada para o sistema é muito flexível e permite o armazenamento completo ou parcial da parametrização dos processos do modelo selecionados através de biblioteca de arquivos. Para incluir a biblioteca de configurações, basta uma simples referência ao arquivo correto.

O sistema de saída é também muito flexível e permite que o usuário defina a própria saída, que pode ser detalhada ou apenas conter os principais resultados da simulação. Os arquivos de saída podem ser carregados facilmente por software de planilhas eletrônicas.

O modelo é dividido em componentes como balanço de água, ciclo do carbono e do nitrogênio e temperatura do solo. O componente

balanço de água é dividido em acúmulo de neve e seu derretimento, interceptação pelo dossel, infiltração e percolação, evaporação do solo, dinâmica da água no solo e transpiração. O componente “ciclo do carbono e nitrogênio” é dividido em crescimento da cultura, retorno de matéria orgânica, adsorção de nitrogênio e balanço do nitrogênio mineral.

O componente de balanço de água do modelo é responsável pelo balanço de água da superfície e do solo; portanto, a água disponível na atmosfera e na terra constitui os limites do sistema considerado. Os fluxos considerados como de superfície são precipitação e irrigação (ganhos) e evapotranspiração e *runoff* de superfície (perdas). Os fluxos considerados no limite inferior são a percolação profunda (perda) ou a ascensão capilar (ganho) e, se o sistema considerado contiver dreno artificial, também fluxo de dreno (perda) (HANSEN, 2002).

O modelo de cultura (Crop) é o mais complexo, porém as iterações com os demais modelos são menos complexos. Tendo como base a Figura 1, o modelo Crop informa a distribuição vertical do dossel ao modelo “Bioclima”, que, por sua vez, retorna a transpiração potencial e a radiação. O modelo pode utilizar essas informações para determinar quanto de água e nitrogênio deve extrair do solo (ABRAHAMSEN; HANSEN, 2000).

2.4.3 Century

Modelo originalmente desenvolvido para simular a dinâmica de C, nutrientes (N, P e S) e MO do solo em pastagens naturais nas planícies norte americanas (PARTON et al., 1987), é aplicado com sucesso a diversos ecossistemas e tipos de solo, especialmente sob condições de clima temperado. É aplicável a diferentes sistemas agrícolas, que

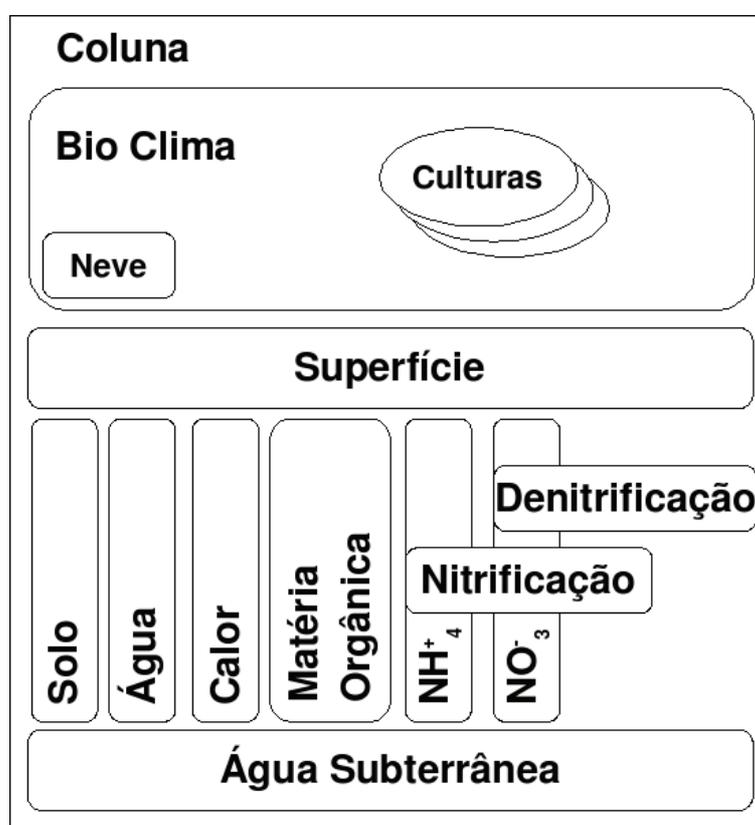


Figura 1: Componentes de uma coluna do modelo Daisy (Adaptado de Abrahamsen e Hansen (2000))

se diferenciam em práticas agrônômicas, como rotação de culturas, preparo do solo, fertilização e irrigação (LEITE; MENDONÇA, 2003; LEITE et al., 2004; CENTURY, 2007).

É composto de vários submodelos, como o submodelo de água, o submodelo de produção vegetal e o submodelo de dinâmica da matéria orgânica do solo, sendo baseado em compartimentos com diferentes características de decomposição ou taxas de transformação (ativo, lento e passivo) (Figura 2). A configuração mínima dos elementos é o C e o N para todos os compartimentos do modelo (LEITE et al., 2004; CENTURY, 2007).

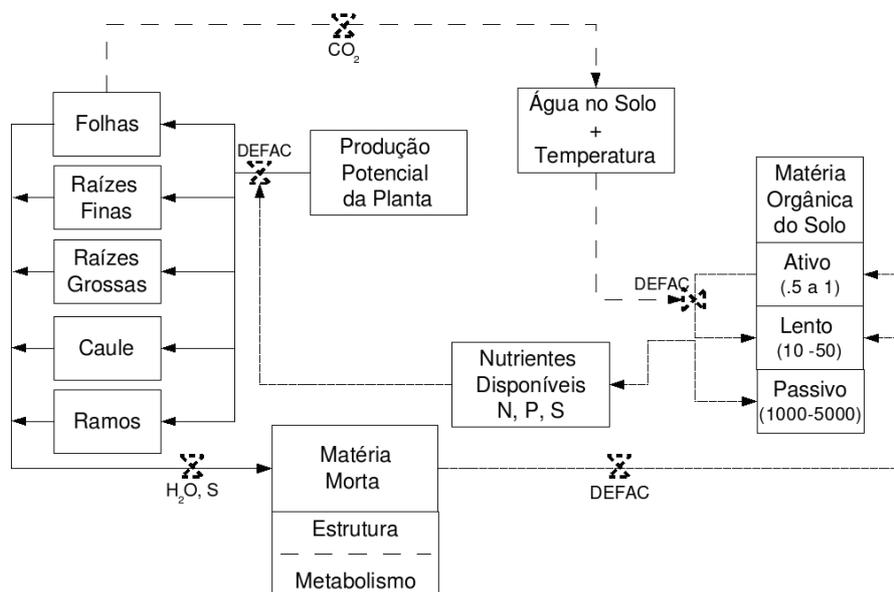


Figura 2: Visão geral do modelo Century (Adaptado de Century (2007))

O passo do modelo é mensal e requer a alimentação por meio de variáveis como: temperatura do ar, precipitação mensal, conteúdo de lignina do material vegetal, teores de N, P e S do material vegetal, textura do solo, aporte de N do solo e da atmosfera e teor inicial de C, N, P e S nos diferentes compartimentos do solo (CENTURY, 2007).

Com o sucesso obtido pelo modelo na simulação da dinâmica da MO em solos de região temperada, estimulou-se que mais pesquisadores aplicassem-no em ecossistemas tropicais. Com o Century, é possível relacionar a MO aos níveis de fertilidade e tipos de solo e ao manejo atual e futuro, otimizar o entendimento das transformações dos nutrientes em solos tropicais e auxiliar no desenvolvimento de sistemas de manejo sustentáveis (LEITE et al., 2004).

Por causa da acurácia e do sucesso do uso deste modelo, o módulo de simulação de resíduos do Century foi incorporado à suite do

DSSAT, tornando os modelos desta mais apropriados para simular e analisar a sustentabilidade a longo prazo (ICASA, 2007).

2.4.4 DSSAT

O DSSAT é uma das mais conhecidas e difundidas aplicações dos modelos de simulação do crescimento e desenvolvimento das culturas, sendo desenvolvido pelo projeto IBSNAT (International Benchmark Sites Network for Agrotechnology Transfer) junto à Universidade do Havaí a partir de 1974. O projeto IBSNAT teve duração de cerca de dez anos, resultando no aplicativo DSSAT e numa rede internacional de pesquisadores envolvidos no desenvolvimento e teste de modelos de simulação. Atualmente, é mantido pelo consórcio ICASA (*International Consortium for Agricultural Systems Applications* - <http://icasa.net>), que reúne um grande número de cientistas de diversas universidades e centros de pesquisa, os quais trabalham de forma colaborativa para a construção dos modelos e suas aplicações (JONES et al., 2001, 2003; ICASA, 2007).

O DSSAT é um sistema composto por diversos modelos de simulação, estes orientados a processos, projetado para aplicações globais e que trabalha independentemente de local, estação e cultivar (VERHAGEN et al., 2001). É apropriado para que os estudos a longo prazo avaliem as estratégias eficientes da gerência da cultura e otimizem a produção (FARIA; BOWEN, 2003). O DSSAT combina dados de solo e de clima com modelos de culturas e aplicativos, a fim de simular os resultados de vários anos de estratégias no manejo de culturas (ICASA, 2007); é o resultado do trabalho colaborativo de muitos cientistas de diversas universidades e centros de pesquisa que fazem parte do consórcio .

Como um pacote de software, que integra os efeitos do solo, características da espécie, coeficientes genéticos, dados climáticos e opções de manejo, o DSSAT permite que os usuários possam solicitar resposta a perguntas como “O que ... se ...?” e simular os resultados de experimentos em minutos num computador, o que poderia consumir uma parte significativa da vida profissional de um agrônomo (ICASA, 2007). A comparação das saídas dos modelos com os resultados observados faz parte da validação de um modelo.

O DSSAT é uma coleção de programas independentes que interagem entre si, na qual os modelos de simulação de culturas ficam no centro (Figura 3), ao passo que as bases de dados descrevem o clima, solo, observações e condições experimentais, além de informações sobre genótipos para aplicar nos modelos em diferentes situações (JONES et al., 2003). Os componentes de software são escritos em:

- Fortran (modelos);
- C (shell);
- Pascal (gráficos);
- Dbase (base de dados) e
- Basic (programas de gerenciamento de estratégia e risco).

O DSSAT fornece também a possibilidade de validação das saídas dos modelos, permitindo, assim, que os usuários possam comparar os resultados da simulação com os resultados observados. Esta validação se dá pela entrada de dados por parte do usuário (dados observados), executando o modelo e comparando as saídas obtidas. Simulando resultados prováveis de estratégias da gerência da cultura, oferece

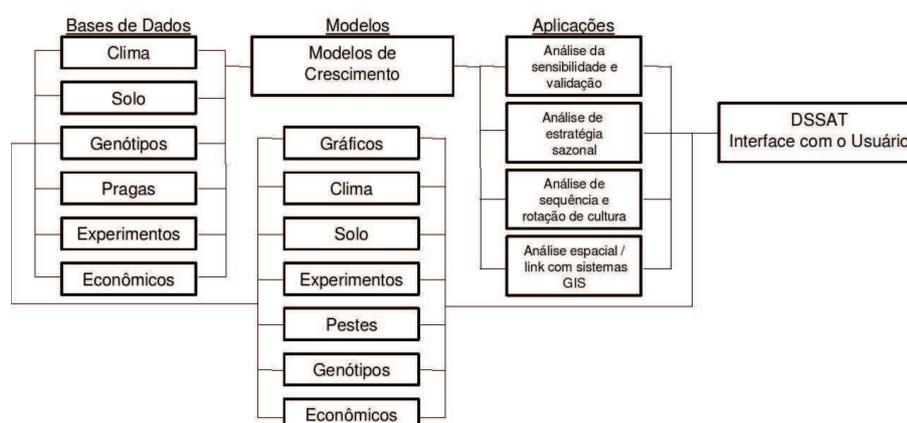


Figura 3: Estrutura do Modelo DSSAT (Adaptado de Jones et al. (2003))

ao usuário as informações necessárias para apreciar novas culturas, produtos e práticas a fim de adotá-las (ICASA, 2007).

Atualmente, a suíte DSSAT é composta por 27 diferentes modelos de culturas, formada por uma coleção de programas independentes que operam em conjunto, estando inseridos numa ferramenta que facilita a criação e o gerenciamento de arquivos de experimentos, de solo e de clima. Inclui aplicações que auxiliam na análise sazonal e sequencial dos riscos econômicos e impactos ambientais associados com irrigação, uso de fertilizantes e nutrientes, mudanças climáticas, sequestro de carbono do solo, variabilidade climática e agricultura de precisão (JONES et al., 2003; ICASA, 2007).

O DSSAT simula o crescimento, desenvolvimento e produção de uma cultura numa área uniforme de terreno, com as informações de manejo informadas ou simuladas, assim como mudanças no conteúdo da água do solo, carbono e nitrogênio que ocorrem em relação ao tempo. Possui uma estrutura composta por um programa principal, uma unidade módulo solo e os módulos para os componentes primá-

rios. A unidade módulo solo é responsável pela interface entre a aplicação e os componentes que interagem numa área uniforme de solo, ao passo que os componentes descrevem as mudanças temporais no solo e nas plantas, em resposta ao clima e ao manejo (JONES et al., 2003). Cada módulo possui seis passos operacionais responsáveis pela estruturação e organização, os quais são: inicialização da execução, instalação da cultura, cálculo das taxas, integração, saídas diárias e sumário das saídas.

Dentre os componentes primários encontrados no DSSAT, como clima, solo-planta-atmosfera, CROPGRO (modelo para a cultura), planta, manejo e praga, o módulo solo é representado como um perfil unidimensional, sendo homogêneo horizontalmente e consistindo de um número de camadas verticais de solo. O módulo solo é composto por quatro submódulos: água no solo, temperatura do solo, carbono e nitrogênio do solo.

O submódulo água no solo computa as mudanças diárias no conteúdo de água do solo pela infiltração da chuva e irrigação, drenagem vertical, fluxo insaturado, evaporação e processo de absorção de água pelas raízes. A infiltração de água é computada pela subtração do *runoff* da precipitação que ocorre durante o dia, sendo o processo baseado no número da curva (*curve number*), que leva em conta a textura, o declive e a cobertura. A drenagem através do perfil é calculada com base num parâmetro assumido para que seja constante com a profundidade. Se a condutividade hidráulica saturada de qualquer camada for menor que a drenagem vertical computada, a água é acumulada na camada acima. Quando há a aplicação de irrigação, a quantidade de água é adicionada à precipitação para cálculo da infiltração e *runoff* diário (FARIA; BOWEN, 2003).

O submódulo solo-planta-atmosfera é responsável por computar diariamente a evaporação do solo e transpiração das plantas. Ele traz junto ao solo as entradas da planta e atmosfera e computa a interceptação da luz pelo dossel, a evapotranspiração potencial e a transpiração da planta.

Com o módulo Praga, o usuário pode entrar com suas observações sobre populações de insetos ou danos em diferentes partes da planta, severidade de doenças em diferentes tecidos da plantas e danos físicos a plantas ou seus componentes. Os efeitos da praga e/ou doença no crescimento e produção são determinados de forma empírica, por meio de uma função de dano.

Os dados mínimos exigidos pelo modelo para sua operação abrangem o local onde o modelo será operado (latitude, longitude, elevação, médias anuais de temperatura e sua amplitude etc.), dados climáticos diários do período do ciclo de crescimento (radiação solar, temperatura do ar máxima e mínima e precipitação), características do solo (características básicas das camadas do solo), condições iniciais (cultura anterior, raiz etc.) e gerenciamento da cultura (tipo e nome do cultivar, data de plantio, profundidade, espaçamento etc.).

Confome Verhagen et al. (2001), dentro da estrutura do DSSAT podem-se encontrar alguns modelos de simulação para diversas culturas, como os seguintes:

- a família CERES de modelos: CERES-maize, CERES-wheat, (CROPSIM-Wheat), CERES-rice, CERES-barley, CERES-sorghum e CERES-millet;
- a série de modelos CROPGRO para leguminosas: CROPGRO-soybean, CROPGRO-peanut, CROPGRO-dry bean (*Phaseo-*

lus);

- a série de modelos CROPSIM para culturas de raiz: CROPSIM-cassava e SUBSTOR-potato;
- outras culturas: CROPGRO-Tomato, CROPGRO-Chickpea, Sugarcane e Sunflower.

Todos os modelos de culturas compartilham um formato comum de entrada e saída e são similares em nível de detalhe, operando num passo de tempo diário; são baseados num entendimento de processos biofísicos. Os modelos são orientados a processo, projetados para aplicações globais e trabalham independentemente de local, estação climática, cultivar e sistemas de gerenciamento. Os modelos simulam o efeito do clima, água no solo, genótipo e dinâmica do nitrogênio na cultura e no solo em relação ao crescimento da cultura e produção (VERHAGEN et al., 2001).

Dentre as maiores limitações existentes no DSSAT relacionadas com os modelos de simulação de culturas estão a ausência do efeito imposto às culturas por pragas/doenças, o modo de preparo do solo, o excesso de água no solo, entre outros, deixando um campo aberto para a inserção de outros modelos, mesmo que isso implique gasto de energia para a integração dos diferentes modelos.

CropSim-Wheat

O CropSim-Wheat, substituto do Ceres-Wheat, é um dos modelos de simulação pertencentes à suite do DSSAT, responsável pela simulação do crescimento e desenvolvimento do trigo (Figura 3, Modelos de Crescimento).

No Brasil, o CropSim-Wheat tem sido testado, calibrado e validado por pesquisadores, sendo utilizado para simular o processo do desenvolvimento de cultivares de trigo. Este modelo permite a sua parametrização por meio de arquivos texto. Suas saídas, resultados da simulação, podem ser utilizadas para alimentar outros modelos responsáveis pela simulação de doenças em plantas, por exemplo (FERNANDES et al., 2004; DELPONTE et al., 2005).

É composto por um módulo planta de trigo que se conecta com o módulo de clima e de solo, pertencentes à suite do DSSAT, os quais computam a energia e a água disponível para o crescimento da planta de trigo, ao passo que o módulo planta de trigo simula os eventos fenológicos, expansão foliar, acúmulo de carboidratos e a partição entre parte aérea e raízes. A alimentação do modelo consiste em características da planta do trigo (cultivares) e dados de clima (radiação solar, temperatura mínima, temperatura máxima e precipitação), assim como práticas agronômicas, como data de semeadura e espaçamento (HUNT; PARARAJASINGHAM, 1995).

2.5 Modelos de Simulação de Epidemias

Além dos modelos de simulação voltados às culturas, muitos esforços foram realizados para o desenvolvimento de simuladores de doenças de plantas. Segundo Fernandes et al. (1994), dentre os primeiros relatados encontra-se o EPIDEM, para a alternariose em tomate (WAGGONER; HORSFALL, 1969), e o EPIMAY, para a queima da folha do milho (WAGGONER et al., 1972). A partir dos trabalhos citados, muitos outros modelos foram desenvolvidos para a simulação de epidemias, alguns sendo utilizados pela academia com fins educacionais e outros, para aplicações práticas em sistemas de auxílio à

tomada de decisões.

Os primeiros modelos de simulação serviram para avaliar aspectos básicos da epidemiologia e identificar falhas existentes no conhecimento do sistema. Com o passar do tempo e com a evolução do conhecimento, outros aspectos foram sendo introduzidos nos modelos de simulação, como o manejo de pragas/doenças, além da tomada de decisão. A inclusão desses aspectos visa ao fornecimento de informações quantitativas da projeção da doença, permitindo que decisões de manejo sejam tomadas levando em consideração a relação custo/benefício (FERNANDES et al., 1994).

Modelos de simulação de epidemias têm sido vislumbrados por muitos como uma ferramenta de utilidade para agricultores e agentes de extensão, com vistas a apoiá-los na tomada de decisão com relação ao controle da doença, podendo projetar situações e estimar a resposta do sistema ao tipo de manejo ou às alternativas disponíveis (FERNANDES et al., 1994; DELPONTE et al., 2004). O principal objetivo no desenvolvimento destes modelos é fornecer, de forma clara e detalhada, as informações necessárias para o auxílio na escolha de estratégias de manejo que mantenham as doenças abaixo do nível de dano econômico estabelecido para uma determinada cultura.

Segundo Fernandes et al. (1994) "a maioria dos simuladores de epidemias traz em si a equação da taxa básica de infecção de Vanderplank", visto que os modelos mais simples se baseiam na equação original, considerando um período latente p , um período infeccioso i e uma taxa R . A epidemia inicia com um conídio depositado numa planta suscetível germinando e, após completar o período latente, estabelecendo uma lesão, iniciando a esporular continuamente durante o período infeccioso. Durante esse período de esporulação, novos es-

poros são produzidos, podendo iniciar uma nova lesão, que, ao final do período latente, produzirá novos esporos, os quais podem originar novas lesões, e assim sucessivamente.

Uma vez que o rendimento final de uma cultura pode ser fortemente influenciado pela ocorrência de epidemias, inúmeros modelos têm sido desenvolvidos, desde os mais simples aos mais complexos, com o intuito de entender o funcionamento destas e implementar sistemas de simulação capazes de auxiliar na tomada de decisões.

Nos últimos anos, muitos esforços foram realizados para o desenvolvimento de diferentes modelos de simulação de epidemias nas mais diversas culturas (MAANEN; XU, 2003a). Por exemplo, podem-se citar os modelos desenvolvidos para os patossistemas *Puccinia*-Trigo e *Phakopsora*-Soja. Na cultura da soja destacam-se os esforços para a ferrugem asiática, causada pelo fungo *Phakopsora pachyrhizi*, ao passo que na cultura do trigo destacam-se os modelos para a ferrugem da folha, causada pelo fungo *Puccinia triticina* (DELPONTE *et al.*, 2006; PIVONIA; YANG, 2006).

DelPonte *et al.* (2006), em seu artigo, relaciona um grupo de doze modelos de ferrugem, organizando-os em modelos de simulação (sete) e modelos empíricos (cinco). Modelos de simulação são baseados em conceitos derivados de um patossistema e ajudam a melhorar o entendimento da estrutura e comportamento de um sistema biológico. Por sua vez, modelos empíricos são tipicamente construídos através de relacionamentos estatísticos de variáveis explanatórias com dados experimentais.

Entre os modelos de simulação relacionados por DelPonte *et al.* (2006) estão o SOYRUST e o CLIMEX, como modelos epidemiológicos, e o HYSPLIT e o SRAPS, como modelos aéreo-biológicos. Os

modelos aéreo-biológicos mostram-se importantes quando se necessita controlar a entrada de uma nova praga numa determinada área. Como exemplo, pode-se citar a necessidade de monitoramento da ferrugem da soja nos Estados Unidos da América, o que fez com que grupos de pesquisadores desenvolvessem um grande número de modelos de simulação (MAGAREY et al., 2007).

Uma descrição breve de alguns modelos relatados por DelPonte et al. (2006) pode ser encontrada abaixo:

- SOYRUST - É um modelo de simulação para a ferrugem da soja (agente causal *Phakopsora pachyrhizi*), desenvolvido para avaliar epidemias da doença, simulando o aumento diário da severidade da doença em duas variedades; reflete o efeito ambiental do ciclo da epidemia, a fim de prever o desenvolvimento, em passos diários. As equações que descrevem os efeitos ambientais nos componentes da doença foram desenvolvidas para reanalisar dados coletados, com vistas a obter um método de avaliação do sistema. As taxas de infecção foram preditas utilizando o orvalho e a temperatura após a inoculação como variáveis independentes ($R^2=0.88$, $P<0.0001$). Os modelos exponenciais, que usaram dia fisiológico como uma variável independente, explicaram 98% das variações do período latente e senescente das lesões da doença (YANG et al., 1991; DELPONTE et al., 2006).
- CLIMEX - É um modelo computacional utilizado para avaliar o efeito das mudanças climáticas em processos biológicos; permite a estimativa da distribuição geográfica potencial e a abundância sazonal de uma espécie em relação ao clima. Os modelos do CLIMEX consistem num conjunto de parâmetros que definem as condições ideais de temperatura e umidade, os quais

permitem o crescimento populacional de uma determinada espécie durante uma estação favorável, e um conjunto de parâmetros de estresse, que definem sua habilidade para sobreviver em estações desfavoráveis (SUTHERST; MAYWALD, 1985; MCCLAY, 2005; DELPONTE et al., 2006).

- HYSPLIT (Hybrid Single-Particle Lagrangian Integrated Trajectory) - É um modelo que simula a simples trajetória de partículas, sua dispersão, concentração e deposição, originado de uma determinada localização geográfica e período do ano. É um modelo de dispersão para simular o transporte a longa distância de *P. pachyrhizi* utilizando prognósticos climáticos de um mês. É o resultado do trabalho cooperativo do NOAA (National Oceanic & Atmospheric Administration) e BOM (Australia's Bureau Of Meteorology) (DELPONTE et al., 2006; NOAA, 2007).
- SRAPS (Soybean Rust Aerobiology Prediction System) - É um modelo modular que trata da maioria dos estádios dos processos aereobiológicos, como a produção de esporos, sua liberação e saída do docel, mortalidade e deposição em regiões alvo. O modelo requer como dados de entrada informações sobre a localização geográfica do inóculo (DELPONTE et al., 2006).

Rossi et al. (1997) apresentam um modelo de simulação para o desenvolvimento de epidemias de ferrugem da folha baseando-se no efeito das condições meteorológicas sobre o ciclo de uredósporos: RUSTDEP (RUST Development of EPidemics). Este modelo simula o progresso da severidade da doença, expressando-o como uma porcentagem da área da folha com ferrugem (folhas individuais). Para novas infecções, o modelo considera a proporção de esporos capazes

de estabelecer a infecção, influenciados pela temperatura e pelo molhamento foliar. O período latente é dependente da temperatura, ao passo que o período infeccioso é governado pela temperatura e estágio de crescimento do hospedeiro. A simulação é realizada sobre o curso de uma estação de crescimento, sendo esta de passo diário.

Em Aris (2000) encontra-se a descrição do desenvolvimento de um modelo de simulação para a mancha da gluma do trigo. O modelo foi criado e adaptado para as condições climáticas prevalentes na Carolina do Norte, EUA, utilizando dados climáticos diários, com o objetivo de ser utilizado posteriormente para toda a região. A metodologia principal utilizada neste trabalho foi a adição de um modelo de doença a um modelo de crescimento de uma cultura já existente, neste caso o modelo CERES-Wheat, do qual as saídas (área foliar) serviram de entrada para o modelo de doença.

Para o modelo de doença desenvolvido por Aris (2000), foi utilizada a linguagem Matlab (*Mathwork Inc., Natick MA*), ao passo que o modelo CERES-Wheat é desenvolvido em Fortran. A validação do modelo deu-se pela comparação das saídas com os resultados obtidos em experimentos de campo.

Pivonia e Yang (2006), em seu trabalho, ajustaram um modelo de previsão de doenças para quatro tipos de ferrugens, sendo duas para milho e duas para trigo, através do efeito da temperatura nos estádios da epidemia, com o intuito de avaliar o potencial estabelecimento pela estimativa do aumento diário da infecção em hospedeiros suscetíveis durante a fase exponencial de uma epidemia.

O conceito de mudança no tempo e no espaço está inserido, implicitamente, na definição de epidemia. Uma maior atenção, por parte dos pesquisadores, é dada para a análise temporal de epidemias, em

detrimento da espacial, em razão da simplicidade e facilidade de aplicação da análise temporal. Um conhecimento amplo das características de uma epidemia, tanto temporal como espacial, possibilita uma visão mais completa da estrutura e do comportamento de patossistemas (VALE et al., 2004).

Willocquet e Savary (2004) apresentam um modelo de simulação epidemiológico integrando três escalas organizacionais de populações em plantas hospedeiras, sendo elas pequenas áreas (sítios), folhas e plantas inteiras. O modelo tem como objetivo incorporar essas escalas numa estrutura espacial hierárquica de uma epidemia, utilizando taxas de auto-deposição, deposição de um propágulo produzido numa lesão localizada na própria folha, e allo-deposição, deposição de um propágulo produzido numa lesão localizada em outra folha da mesma planta ou outra planta.

Com um passo diário, o modelo simula o ciclo de infecção na escala do sítio, da folha e da planta. Na escala do sítio, o modelo distingue três tipos de infecções: (i) infecções de áreas localizadas na folha doente, (ii) infecções de áreas pertencentes a folhas sadias de uma planta infectada e (iii) infecções de áreas localizadas em plantas sadias (Figura 4). A distinção destes três tipos de infecções permite a incorporação de parâmetros epidemiológicos para a propagação da doença em diferentes escalas, assim como as taxas de autodeposição e allo-deposição dos propágulos (WILLOCQUET; SAVARY, 2004).

Para o sítio, mais baixa escala, o modelo simula a dinâmica das áreas livres, latentes, infecciosas e removidas, considerando que não há propágulos vindos de fora, que o número de áreas, folhas e plantas não muda no tempo e que não há expansão das lesões (WILLOCQUET; SAVARY, 2004).

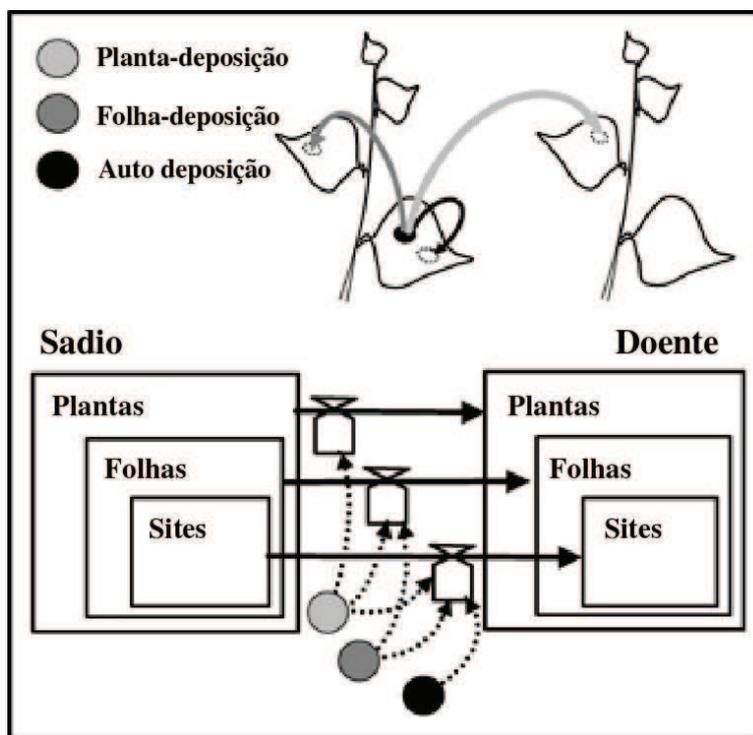


Figura 4: Auto e allo-deposição (Traduzido de Willocquet e Savary (2004))

Além de fornecer uma base para avaliar a eficiência potencial do uso de diferentes cultivares com suscetibilidade e resistência num patossistema, fornece uma maneira de investigar os efeitos da interação entre a arquitetura do hospedeiro e a disseminação do patógeno na dinâmica da epidemia (WILLOCQUET; SAVARY, 2004).

Robinson e Geils (2006), apresentam um modelo conceitual e numérico que representa a epidemiologia do nanismo do *mistletoe* (*Arceuthobium*) em árvores coníferas. Este modelo é responsável pela simulação da reprodução, dispersão e padrões espaciais do *mistletoe*, usado para o gerenciamento e planejamento silvicultural. É baseado no conhecimento da biologia do *mistletoe* e de práticas florestais adquiridas através de *workshops*, exercícios programados e pesquisa e

desenvolvimento continuado.

O modelo desenvolvido foi ligado a outros dois modelos de crescimento de árvores, auxiliando na identificação das perspectivas e desafios na compreensão da espécie invasiva, da progressão da doença e do padrão espacial dos *mistletoe* e seus hospedeiros. A dinâmica espacial da dispersão opera através de um conjunto de escalas, sendo estas dentro da própria árvore, para as árvores vizinhas e outras mais distantes. O modelo epidemiológico opera com passo anual, realizando a inicialização, a contabilização da simulação e o retorno dos resultados para os modelos de crescimento.

Verifica-se que o desenvolvimento das doenças é extremamente dependente dos elementos climáticos e da planta, dando-se os avanços em função do microclima do dossel. O monitoramento desses elementos é de fundamental importância, tendo em vista a construção de modelos de simulação.

A complexidade das inter-relações entre os diferentes fatores que contribuem para a resposta ao uso de pesticidas seguidamente leva a que o uso desses insumos seja feito de maneira inadequada. O uso incorreto desses, além de não proporcionar o resultado esperado, aumenta o custo de produção. O uso de modelos complexos, essencialmente mecanísticos, do tipo de simulação do crescimento e desenvolvimento de culturas ligados à ocorrência e evolução de pragas, doenças e plantas invasoras, pode servir para orientar decisões no manejo desses fatores, incluindo aspectos como a probabilidade de ocorrência, o curso e o risco de perdas na produção.

O caráter integracionista presente nos modelos de simulação é uma das grandes contribuições que a simulação da dinâmica de processos biológicos tem proporcionado à ciência agrícola. Atualmente, os mo-

delos de simulação são imprescindíveis, principalmente com vistas ao manejo integrado de pragas e doenças.

2.6 Ciclo de Vida de um Modelo de Simulação

No que diz respeito ao desenvolvimento de modelos de simulação, devem-se seguir certos passos a fim de que seja bem-sucedido. Segundo Chwif (1999), estes passos ou processos são conhecidos na literatura como “metodologia de simulação” ou “ciclo de vida de um modelo de simulação”.

Conforme Chwif (1999), o ciclo de vida de um modelo de simulação é composto por três etapas:

- concepção ou formulação do modelo;
- implementação do modelo;
- análise dos resultados do modelo.

Para a primeira etapa, o sistema em estudo deve ser bem entendido, de forma a se obter claramente os objetivos do mesmo. Isso pode ser feito pelo amplo estudo do sistema e seus inter-relacionamentos ou por meio de discussões com o especialista no assunto. A abrangência do modelo deve ser decidida em nível de detalhe e as hipóteses, estabelecidas. Após, deve-se representar o modelo com alguma técnica a fim de obter um modelo conceitual, para que seja facilmente entendido.

A coleta de dados também tem uma grande importância nesta etapa, sendo necessário ter em mente que a qualidade dos mesmos é fundamental, de forma a refletir a realidade no momento do ajuste e

teste do modelo. Como já mencionado, é preciso ter em mente que se forem fornecidos dados não confiáveis (lixo), ter-se-ão como resultado dados não confiáveis (lixo) (MEDEIROS, 2003).

Para a segunda etapa, deve-se converter o modelo conceitual num modelo computacional, implementando-o numa linguagem de programação que melhor reflita as necessidades e as conexões com outros modelos, devendo ser verificado, contra o modelo conceitual, se os resultados estão de acordo com o pretendido e/ou observado (CHWIF, 1999).

Após a verificação e validação do modelo computacional, tem-se a terceira etapa, na qual se realizam os experimentos, originando o modelo experimental (operacional). Nesta etapa são realizadas várias “rodadas” do modelo, analisando-se os resultados da simulação com o uso de técnicas estatísticas e armazenando-os para posteriormente gerar as conclusões e recomendações. Caso os resultados não sejam o esperado, modifica-se o modelo e recomeça-se o ciclo novamente (CHWIF, 1999).

As etapas apresentadas acima, e visualizadas na Figura 5, podem não ser seguidas linearmente num estudo prático, podendo haver iterações e retroalimentação no processo à medida que o entendimento do problema muda (CHWIF, 1999).

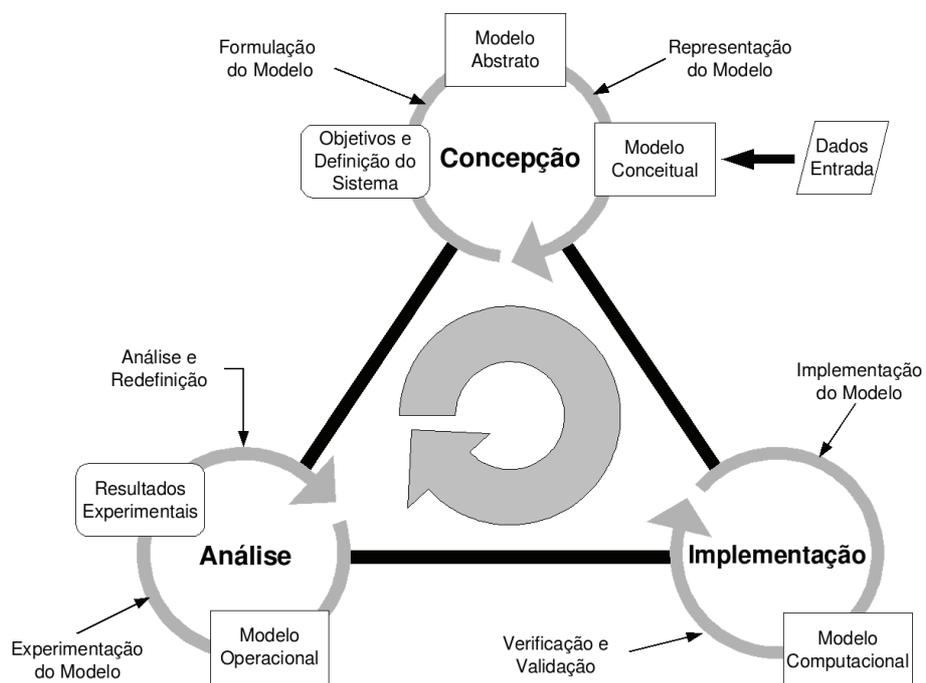


Figura 5: Ciclo de vida de um modelo de simulação (Adaptado de Chwif (1999))

CAPÍTULO I

PADRÕES DE PROJETO APLICADOS A MODELOS DE SIMULAÇÃO DO CRESCIMENTO E DESENVOLVIMENTO DAS CULTURAS

Willingthon Pavan

Faculdade de Agronomia e Medicina Veterinária

Universidade de Passo Fundo

C.P. 611, 99001-970

RESUMO

O desenvolvimento de modelos de simulação na agricultura, assim como em outras áreas, requer uma análise profunda do sistema em estudo, sendo crucial o emprego de métodos e ferramentas de engenharia de software. A modularidade e a forma genérica são os termos que descrevem a nova e amplamente aceita metodologia para superar as complexidades no desenvolvimento e reuso de modelos. A arquitetura MVC (Model-View-Controller), um padrão de projeto amplamente aceito no desenvolvimento de software, auxilia na definição das regras e responsabilidades dos objetos num sistema, possibilitando o desenvolvimento de aplicações com fluxo simplificado, robusto, flexível e de fácil manutenção. Um dos principais aspectos do MVC reside na separação entre os módulos da aplicação, de forma a possibilitar o reaproveitamento e o desenvolvimento genérico de modelos. Este trabalho teve como objetivo apresentar uma estrutura para o desenvolvimento de simuladores de doenças de plantas, acoplados a modelos de crescimento, com a utilização extensiva de padrões de projeto orientados a objetos. Com este trabalho tornou-se possível

organizar e reutilizar conhecimentos construídos durante as últimas décadas, evitando mudanças bruscas e melhorando substancialmente a performance e a forma de disponibilização dos resultados. Com os resultados, observou-se que a aplicação de técnicas e padrões no desenvolvimento de sistemas complexos, como os modelos de simulação, torna-os mais robustos, bem documentados, reusáveis, compatíveis e facilmente acopláveis a outras soluções já existentes.

ABSTRACT

DESIGN PATTERNS APPLIED TO CROP GROWTH MODELS

The development of simulation models for agriculture, as well as in other areas, requires a deep analysis of the system in study, being crucial the choice of methods and tools of software engineering. The modularity and the generic form are terms that describe the new and widely accepted methodology to surpass the complexities in the development and re-use of models. The MVC (Model-View-Controller) architecture, a project design pattern widely accepted in the software development industry, assists in the rules definition and properties of the objects in a system, allowing the development of simpler applications, robust, with flexible flow and easy maintenance. One of the main aspects of the MVC architecture resides in the separation of the application modules which permits the development of generic and reusable models. This work had as objective to present a structure for the development of plant disease simulators coupled to crop growth models with the extensive use of object oriented design pattern. In this work, it was substantially demonstrated how to organize and to reuse knowledge constructed during the last decades, preventing huge changes and improving the performance and the delivering of results. It was shown that the use of design pattern techniques for development of simulators of complex systems resulted in more robust, well documented and reusable simulation models. In addition the models were easily connected to other already existing applications.

Introdução

O desenvolvimento de modelos de simulação envolvendo sistemas de produção agrícola requer uma análise profunda do sistema, geração e acúmulo de conhecimento e uma ampla base de dados experimentais. Porém, é também crucial o emprego de métodos e ferramentas de tecnologia de software. A modularidade e a forma genérica são os termos que descrevem a nova e amplamente aceita metodologia para superar as complexidades que surgem ao construir, manter e reusar modelos em partes ou como um todo (REYNOLDS; ACOCK, 1997).

Na agricultura, como em outros modelos dinâmicos, o sistema pode ser analisado em termos de variáveis de estado, de fluxo e auxiliares (WIT, 1982). Estas variáveis, em geral, acabam fazendo parte de um conjunto de equações diferenciais que devem ser integradas numericamente, dada a complexidade do sistema. Esta estrutura comumente usada permite que seja possível desenhar e codificar um modelo com finalidades de entradas e saídas arranjadas com ferramentas de software.

Desde o surgimento dos primeiros computadores, o homem vem desenvolvendo técnicas para levar para dentro dos computadores o que encontra no mundo real, as linguagens de programação. Essas possibilitam que os computadores sigam ordens predefinidas e reportem os resultados obtidos. Nas últimas décadas, muitas linguagens foram criadas e em muitos paradigmas foram desenvolvidas, tentando fazer com que os computadores se pareçam cada vez mais com o cérebro humano. Dentre os paradigmas desenvolvidos estão o imperativo, o lógico, o funcional e o orientado a objetos, cada um tentando resolver os problemas de comunicação de forma diferente.

Modelos podem ser definidos como uma representação simplificada, por necessidade, do que se percebe ser realidade, podendo ser um objeto, uma idéia ou um sistema (SHANNON, 1975). São uma descrição matemática das diversas causas e efeitos envolvidos num sistema real; para serem perceptíveis é necessário que as relações entre causas e efeitos sejam claras e limitadas (KEEN; SPAIN, 1992). São provenientes de aproximações realizadas para se poder entender melhor um determinado fenômeno, mesmo que essas aproximações não condizam totalmente com a realidade. De qualquer forma, um modelo retrata, ainda que de maneira simplificada, os aspectos da situação pesquisada (LEAL, 1999).

Modelos de simulação do crescimento e do desenvolvimento de culturas têm sido projetados e desenvolvidos em várias partes do mundo (SWANEY et al., 1983; JONES; RITCHIE, 1990; MAVROMATIS et al., 2001; VERHAGEN et al., 2001; KEATING et al., 2003; NORWOOD et al., 2006). Dentre as culturas, pode-se citar o desenvolvimento para o trigo, soja, milho, cana-de-açúcar, mandioca etc., sendo esses para fins acadêmicos e práticos. Como um exemplo prático pode-se citar o uso do APSIM (Agricultural Production Systems SIMulator) e do DSSAT (Decision Support System for Agrotechnology Transfer).

O APSIM foi desenvolvido na Austrália para simular processos biológicos em sistemas agrícolas, relacionando os resultados econômicos e biológicos das práticas gerenciais em face do risco climático (KEATING et al., 2003). Este modelo está integrado a um sistema de suporte à tomada de decisão chamado “*Whopper Cropper*”, o qual tem como objetivo fornecer orientações no manejo de culturas, utilizando dados climatológicos observados e de previsão sazonal (COX, 2006).

O DSSAT, desenvolvido por um grupo de pesquisadores das universidades da Geórgia, Flórida, Hawaii, Guelph e Iowa State, além do centro internacional para fertilidade do solo e desenvolvimento agrícola (IFDC - International Center for Soil Fertility and Agricultural Development), simula o crescimento de culturas, produtividade, necessidades de água e nutrientes e o impacto ambiental na produção agrícola, estando atualmente na versão 4.0.

A maior parte dos programas de simulação do crescimento e desenvolvimento de culturas foi escrita em Fortran, razão por que muitos esforços têm sido feitos para que sejam reestruturados, visando aos aspectos fundamentais de modularidade. Uma vez que muitos programas foram desenvolvidos de forma independente, a incompatibilidade representa uma dificuldade para aqueles que se encontram fora do ambiente de desenvolvimento. Assim, nos dias de hoje há um consenso sobre a necessidade de se desenvolverem modelos mais compatíveis e eficientes, o que leva a que novas técnicas sejam utilizadas para minimizar esta heterogeneidade.

Na busca por técnicas de desenvolvimento que contemplem as necessidades de compatibilidade, interatividade e expansibilidade de modelos, novas tecnologias foram criadas a fim de possibilitar a aplicação da idéia de modularidade no desenvolvimento, como o paradigma de orientação a objetos (PAPAJORGJI; PARDALOS, 2006). A programação orientada a objetos tem revolucionado a maneira como os softwares são concebidos, escritos e mantidos. A orientação a objetos simplesmente significa a visualização de um problema em termos de objetos envolvidos com este, permitindo que os softwares não apenas modelem o mundo real, mas permitam que o código seja escrito de uma forma mais organizada e consistente (ROBERTS; DESSOUKY, 1998).

Características de reaproveitamento são usadas com grande frequência na programação orientada a objetos, pela facilidade e rapidez no desenvolvimento, podendo ser aproveitadas para desenvolver sistemas de simulação complexos. A orientação a objetos possibilita inserir novos elementos no ambiente sem que haja a necessidade de recodificação, permitindo que os componentes envolvidos possam trocar informações necessárias para o funcionamento do modelo; assim, possibilita-se o desenvolvimento de softwares modulares capazes de se acoplar na busca de um objetivo específico. Busca-se, dessa forma, utilizar estas tecnologias a fim de obter um módulo que agregue diversos outros submódulos, capazes de gerar os resultados esperados.

Na busca por padrões que viessem a auxiliar os desenvolvedores surgiram os *Design Patterns*, os quais são geralmente tidos como soluções já testadas e bem-sucedidas para problemas encontrados. O princípio básico é que, à medida que os problemas vão ocorrendo, é preciso que sejam documentados para que no momento em que voltem a ocorrer seja possível saber qual providência tomar. Assim, pode-se entender como *Design Patterns* a descrição de como resolver da melhor forma possível um determinado problema que se repete (CORAZZA, 2004).

A utilização de programação multicamadas permite que as aplicações se tornem independentes da arquitetura do sistema, oferecendo uma facilidade maior quando da sua manutenção. Com esta técnica, é possível realizar a separação das camadas de apresentação e das regras de negócio, além de fazer a separação destas da camada de armazenamento, seja utilizando bancos de dados, XML ou outros; assim, facilita-se o desenvolvimento de softwares capazes de serem mantidos, evitando a propagação para outros módulos. Quando o objetivo é o desenvolvimento de softwares robustos, o uso de multica-

madras torna-se indispensável, em virtude do grande impacto que as mudanças causam num software.

Desde o final da década de 1960 até os dias atuais, pesquisadores têm desenvolvido modelos computacionais para a simulação do crescimento e do desenvolvimento de culturas, incluindo o rendimento (BECK et al., 2002). Observa-se que na maior parte desses modelos de simulação, como o CropSim-Wheat, são raros ou inexistentes os que apresentam alguma forma de contabilização do impacto das pragas numa planta.

Segundo o conceito estabelecido pela FAO (Food and Agriculture Organization of the United Nations), entende-se como praga qualquer espécie, raça ou biótipo de vegetais, animais ou agentes patogênicos nocivos aos vegetais ou produtos vegetais. Dessa forma, o termo compreende animais (insetos, ácaros e nematóides), doenças (causadas por fungos, bactérias, fitoplasma, vírus e viróides) e plantas daninhas (plantas invasoras ou plantas espontâneas) (KOGAN, 1998).

Em virtude da inexistência da contabilização do impacto das pragas por grande parte dos modelos de simulação de culturas, este projeto foi originado buscando criar uma estrutura capaz de integrar os diferentes sistemas de simulação, de forma facilitada e com o uso de modernas técnicas de desenvolvimento, além da possibilidade de uso de uma base de dados centralizada.

Dessa forma, pela união das técnicas de programação orientada a objetos e uso de padrões de desenvolvimento, pode-se alcançar um alto nível na criação de sistemas de modelagem e simulação, tanto nas áreas do crescimento e desenvolvimento de plantas como no de submodelos, como os das pragas. O objetivo deste trabalho é propor uma estrutura para o desenvolvimento de simuladores acoplados a

modelos de crescimento de plantas, utilizando programação orientada a objetos numa plataforma MVC.

Material e Métodos

Para o desenvolvimento deste trabalho utilizaram-se diferentes tipos de técnicas e tecnologias, desde aquelas desenvolvidas e utilizadas nos primórdios da computação até as existentes nos dias atuais, como a modelagem orientada a objetos, o *design pattern* MVC, as linguagens de programação orientadas a objetos, a separação das tecnologias na forma de multicamadas e muitas outras.

Do conhecimento legado utilizaram-se daqueles construídos desde o final da década de 1960 até os dias de hoje, que são os modelos de simulação do crescimento e desenvolvimento de plantas escritos em linguagem Fortran.

DSSAT

O DSSAT é um sistema composto por diversos modelos de simulação, esses orientados a processos, projetados para aplicações globais e que trabalham independentemente de local, estação e cultivar (VERHAGEN et al., 2001). É apropriado para que os estudos a longo prazo avaliem as estratégias eficientes da gerência da cultura e otimizem a produção (FARIA; BOWEN, 2003). O DSSAT combina dados de solo e de clima com modelos de culturas e aplicativos a fim de simular os resultados de vários anos de estratégias no manejo de culturas (ICASA, 2007). É o resultado do trabalho colaborativo de muitos cientistas de diversas universidades e centros de pesquisa que fazem parte do consórcio ICASA (International Consortium for Agricultural

Systems Applications)(<http://icasa.net>).

Como um pacote de software, que integra os efeitos do solo, características da espécie, coeficientes genéticos, dados climáticos e opções de manejo, o DSSAT permite que os usuários possam solicitar resposta a perguntas como “O que ... se ...?” e simular os resultados de experimentos em minutos num computador, os quais poderiam consumir uma parte significativa da vida profissional de um agrônomo (ICASA, 2007). A comparação das saídas dos modelos com os resultados observados faz parte da validação de um modelo.

O DSSAT é uma coleção de programas independentes que interagem entre si, no qual os modelos de simulação de culturas ficam ao centro (Figura 6), enquanto as bases de dados descrevem o clima, solo, observações e condições experimentais, além de informações sobre genótipos para aplicar nos modelos em diferentes situações (JONES et al., 2003). Os componentes de software são escritos em:

- FORTRAN (modelos);
- C (shell);
- Pascal (gráficos);
- Dbase (base de dados) e
- Basic (programas de gerenciamento de estratégia e risco).

Cropsim-Wheat

Dentre os vários modelos de simulação que fazem parte da suíte do DSSAT, escolheu-se como modelo de simulação do crescimento e

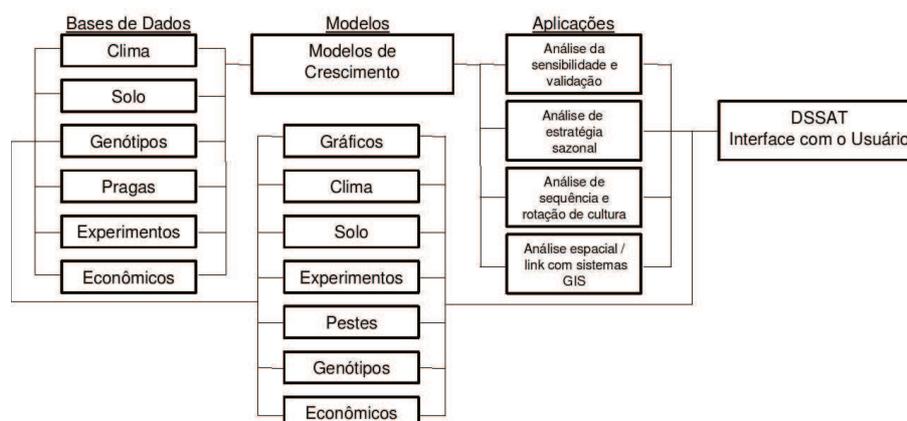


Figura 6: Estrutura do Modelo DSSAT (Adaptado de Jones et al. (2003))

desenvolvimento do trigo, para ser inserido na estrutura aqui apresentada o modelo Cropsim-Wheat, o qual é desenvolvido e mantido por Leslie A. Hunt (HUNT, 2007).

Dentre os 27 modelos de crescimento de culturas incorporados no DSSAT (ICASA, 2007), o CropSim-Wheat apresenta-se como responsável pela simulação do crescimento e desenvolvimento do trigo (Figura 6, Modelos de Crescimento). Consiste do módulo planta de trigo que se conecta com os módulos de clima e de solo, pertencentes à suite do DSSAT, os quais computam a energia e a água disponível para o crescimento da planta de trigo, ao passo que o módulo planta de trigo simula os eventos fenológicos, expansão foliar, acúmulo de carboidratos e a partição entre a parte aérea e as raízes.

A alimentação do modelo consiste em características da planta do trigo (cultivares), dados de clima (radiação solar, temperatura mínima, temperatura máxima e precipitação), assim como práticas agrônomicas, data de semeadura e espaçamento (HUNT; PARARA-JASINGHAM, 1995).

R

Para possibilitar a integração entre os diversos módulos dos sistemas, desenvolvidos em diferentes linguagens, foi utilizada a tecnologia R (<http://www.r-project.org>), a qual é uma linguagem e ambiente para computação estatística e geração de gráficos, similar à linguagem e ambiente S, desenvolvida no Bell Laboratories por John Chambers e colegas (RIBEIRO JUNIOR, 2001; R Development Core Team, 2006).

O R disponibiliza uma grande variedade de métodos estatísticos e técnicas gráficas. Um dos seus pontos fortes é a facilidade com que gráficos de alta qualidade podem ser produzidos. É uma linguagem bem desenvolvida, possuindo características herdadas de outras, como estruturas de controle, funções definidas pelo usuário (programação estruturada e modular), recursividade, facilidade de manipulação de arquivos e programas externos, orientação a objetos, entre outras (RIBEIRO JUNIOR, 2001; SILVA JUNIOR, 2005).

O R é disponibilizado sob os termos da GNU (General Public License) da Free Software Foundation (<http://www.fsf.org>) na forma de código aberto; roda num grande número de plataformas UNIX e similares (incluindo FreeBSD e Linux), podendo também ser compilado e rodar em Windows 9x/NT/2000 e MacOS (RIBEIRO JUNIOR, 2001; SILVA JUNIOR, 2005).

R é um ambiente integrado de facilidades de software para a manipulação de dados, cálculos e geração de gráficos; inclui um efetivo tratamento de dados e facilidades de armazenamento, tanto em arquivos como em bancos de dados; pode ser expandido com a adição de novas funcionalidades pela definição de novas funções. Como em muitas outras linguagens, a maior parte das bibliotecas existentes na

linguagem é desenvolvida na própria linguagem e disponibilizada na forma de pacotes, tornando-o dinâmico e muito poderoso (R Development Core Team, 2006).

Para tarefas que exigem um alto poder de processamento, linguagens como C, C++ e Fortran podem ser utilizadas para desenvolvimento de blocos de código e integrados ao R, podendo ser chamados em tempo de execução, assim como códigos podem ser escritos nestas linguagens para manipular diretamente objetos em R (R Development Core Team, 2006).

Com as potencialidades e características da linguagem, aliado à enorme quantidade de pacotes disponíveis (<http://cran.r-project.org/src/contrib/PACKAGES.html>) para os mais variados fins, torna-se possível a implementação de qualquer tipo de solução computacional, o que conduziu a sua escolha como linguagem para a realização da ponte entre os modelos de simulação escritos em Fortran e os em Java.

Java

Para o desenvolvimento dos componentes, sistemas e modelos de simulação foi utilizada a linguagem de programação Java (<http://java.sun.com>), a qual é uma linguagem de distribuição gratuita, capaz de produzir softwares robustos e multiplataforma que podem rodar em diversos tipos de microcomputadores e dispositivos de pequeno porte.

Java é uma linguagem orientada a objetos, de alto nível, influenciada de várias formas por C, C++ e Smalltalk, assim como por boas idéias implementadas em outras linguagens. Foi desenvolvida por uma

equipe de desenvolvedores da Sun Microsystems no início da década de 1990, sendo composta por uma gama de produtos baseados no poder da rede e na idéia de que um software deve ser capaz de rodar em diferentes máquinas, sistemas e dispositivos, como handhelds, PDAs (Palm), celulares, entre outros (MICROSYSTEMS, 2007; SILVEIRA, 2007).

Java é dividida em três plataformas: Standard Edition (SE), Enterprise Edition (EE) e Micro Edition (ME). A plataforma SE é voltada para o desenvolvimento de aplicações Desktop, ao passo que a EE é voltada para desenvolvimento Web; já a ME foi criada para o desenvolvimento de aplicações para equipamentos de pequeno porte (ADAMSON, 2006).

Uma plataforma pode ser definida como um ambiente de hardware ou software em que programas são executados, ou seja, uma combinação de sistema operacional e hardware. As plataformas em Java são compostas por dois componentes: a máquina virtual Java (JVM) e a API Java (Java Application Programming Interface). A JVM é a base para as plataformas Java, sendo portada para vários tipos de hardwares, ao passo que a API Java é uma grande coleção de componentes de softwares que fornecem muitas potencialidades úteis, agrupadas em bibliotecas de classes e interfaces, conhecidas como pacotes (*packages*) (MICROSYSTEMS, 2007).

Java destaca-se entre as demais linguagens pelo fato de ser dinâmica, isto é, por ser uma linguagem interpretada e poder ser carregada a qualquer momento, mesmo quando o interpretador já estiver rodando, permitindo que estas classes sejam carregadas dinamicamente e instanciadas conforme o fluxo de execução. A uniformidade dos conceitos de Java permite a integração de conceitos modernos, como

componentes, invocação remota, reflexão/introspecção, validação, conectividade a bancos de dados e muitos outros.

Com relação à conectividade a banco de dados, Java possui integração com um grande número de SGBDs relacionais e orientados a objetos. É detentora de uma poderosa API (Application Program Interface), a JDBC-ODBC (Java DataBase Connectivity and Open Database Connectivity), escrita inteiramente em Java, a qual permite o envio de comandos SQL para um driver de banco de dados, responsável por acessar o banco e devolver os MSP (Modelo de Simulação de Pragas). Permite, dessa forma, que a aplicação não fique presa a um banco de dados específico, o que pode ser substituído a qualquer instante.

Rserve

Para a realização da integração entre as tecnologias (Java, R, C e Fortran), utilizou-se o Rserve, um servidor TCP/IP desenvolvido por Simon Urbanek sob licença GPL, que permite que outros programas se utilizem das facilidades da linguagem R ou façam ligações com a biblioteca do R (URBANEK, 2003; RSERVE, 2007).

Todas as conexões realizadas através do Rserve possuem uma *workspace* (área de trabalho) e um diretório de trabalho separado. As implementações funcionam como aplicações cliente-servidor, permitindo que linguagens como C/C++ e Java possam se utilizar dos recursos da linguagem R (URBANEK, 2003). O Rserve suporta conexões remotas, autenticação e transferência de arquivos. O seu uso típico é para integrar o R (retaguarda) com sistemas para computação de modelos estatísticos, gráficos, entre outros (URBANEK, 2003; RSERVE, 2007).

Para o desenvolvimento com a linguagem Java, utilizou-se a biblioteca JRclient, um completo conjunto de ferramentas que permite que qualquer aplicação Java acesse o Rserve, sendo inteiramente desenvolvido em Java. Fornece uma transformação automática de tipos entre o Java e o R, como a transformação dos int, double, arrays, String, vetores e classes para os objetos específicos do R, como RBool, RList etc. (URBANEK, 2003; RSERVE, 2007).

Banco de dados

Os dados necessários para “alimentar” o sistema foram dispostos, em forma de tabelas, numa base de dados relacional implementada no servidor de banco de dados PostgreSQL 8.1 (<http://www.postgresql.org>), sob o servidor Linux Ubuntu Dapper (<http://www.ubuntu.com>).

A base de dados foi modelada a fim de armazenar as informações climáticas (dados observados e dados de prognósticos), os dados de entrada para os modelos (tipos de solo, espécies, cultivares, tratamentos etc.) e os dados resultantes da execução dos modelos simulação, possibilitando a análise e a validação dos modelos.

Infra-estrutura

Tendo em vista a necessidade de seguir regras bem estabelecidas e de forma clara, utilizaram-se padrões, os quais são maneiras de descrever as melhores práticas, os bons projetos e a captura de experiências, de forma que seja possível, para outros, reutilizá-las (HILLSIDE, 2007). O estabelecimento de padrões auxilia o desenvolvedor a criar sistemas, aproveitando-se de técnicas já conhecidas e bem estudadas com o intuito de modelar o comportamento das aplicações.

Atualmente, podem-se encontrar diversos tipos de “padrões de projeto”, cada um com uma finalidade específica, nomeando, abstraindo e identificando os aspectos-chave de uma estrutura. A descrição de soluções já testadas para determinados problemas de projeto forma uma base sólida de soluções, as quais atuam como blocos de software pré-fabricados para a construção de softwares complexos. A qualidade do software está intimamente relacionada com a utilização, por parte do profissional, desses padrões.

Dentre os inúmeros padrões existentes, como o Business Delegate, Composite Entity, Composite View, Data Access Object etc., destaca-se o MVC (Model-View-Controller - Modelo-Visão-Controle), sendo o *desing pattern* mais conhecido e utilizado. É responsável por separar acesso a dados, lógica de negócio e apresentação de dados e interação de usuário.

MVC

Para obter uma estrutura clara e que possibilitasse a reutilização e acoplamento de novos módulos e modelos, utilizou-se o padrão MVC, o qual é um modelo de desenvolvimento de aplicações que utiliza as características da programação em camadas, sendo divididas em três as camadas ou áreas funcionais: Modelo (Model), Visão (View) e Controle (Controller).

Modelo

O modelo é a parte que representa o estado e o comportamento do componente, gerenciando e conduzindo todas as transformações, não tendo conhecimento específico sobre o controlador nem sobre a visão; é o próprio sistema que mantém links entre o modelo e a visão,

notificando a visão quando há mudanças no seu estado. É um repositório de dados e de lógicas de negócio, ou seja, uma representação do banco de dados. Geralmente, em aplicações baseadas em banco de dados parte da função do modelo é retornar dados de consultas ao banco e persisti-los.

A implementação da camada do modelo deu-se pelo uso de JavaBeans (beans) e Enterprise JavaBeans (EJB), os quais oferecem escalabilidade, concorrência, balanceamento de cargas, gerência de recursos automática além de outros benefícios. Os *beans* fornecem o acesso rápido aos dados, ao passo que os EJBs fornecem acesso à lógica e aos dados compartilhados do negócio (MICROSYSTEMS, 2002).

Visão

A visão é a responsável por disponibilizar os dados produzidos pelo modelo, gerenciando o que pode ser visto do estado representado pelo modelo. Os componentes da visão, também conhecidos como componentes de apresentação, em aplicações Web, geralmente, são páginas que geram conteúdo dinâmico e seus MSP podem incluir conteúdo HTML, arquivos PDF, arquivos XML, imagens, gráficos etc.

Para a implementação da visão utilizaram-se as tecnologias JSP (JavaServer Pages) e Servlets (Classes Java executadas no servidor), acessando a camada de modelo e gerando, de forma dinâmica, o conteúdo estático a ser devolvido ao cliente, em formatos texto, XML, WML, gráfico e HTML. Como lógica de interação, quando o usuário final executar alguma ação dentro das páginas HTML criadas pela visão, um evento é submetido ao controlador, que terá a função de verificar o que fazer com essa ação.

Controle

O controle é responsável por receber as informações passadas pelo usuário através das visões e, baseado no conteúdo desta requisição, combinado com a programação ou o meta dado, decidir o que deve ser feito (CORAZZA, 2004). Em outras palavras, gerencia a interação do usuário/sistema com o modelo, fornecendo o mecanismo pelo qual as mudanças são feitas no estado do modelo.

Para a implementação do controle utilizou-se o Apache Struts 2.0 (<http://struts.apache.org>), o qual é um framework livre, de código aberto, destinado à criação de aplicações Web na linguagem de programação Java. É composto de diversas tecnologias, possibilitando aos desenvolvedores Web criar aplicações baseadas em padrões de fácil construção, com facilidades de extensão e de fácil manutenção. Além do MVC, o Struts adere a uma série de outros padrões e boas práticas, possibilitando o desenvolvimento de forma transparente (FOUNDATION, 2007).

Resultados

O modelo MSP (Modelo de Simulação de Pragas) foi desenvolvido utilizando as técnicas de MVC, multicamadas e orientação a objetos, além de diversas outras tecnologias mencionadas, dividindo-se em módulos de software responsáveis pela manipulação de dados, interface com o usuário e controle da “lógica de negócio”, que está por trás da interface com o usuário. A Figura 7 mostra o relacionamento entre as camadas do modelo.

Para a execução do modelo MSP foram projetados servidores com responsabilidades distintas de processar as requisições feitas pelos de-

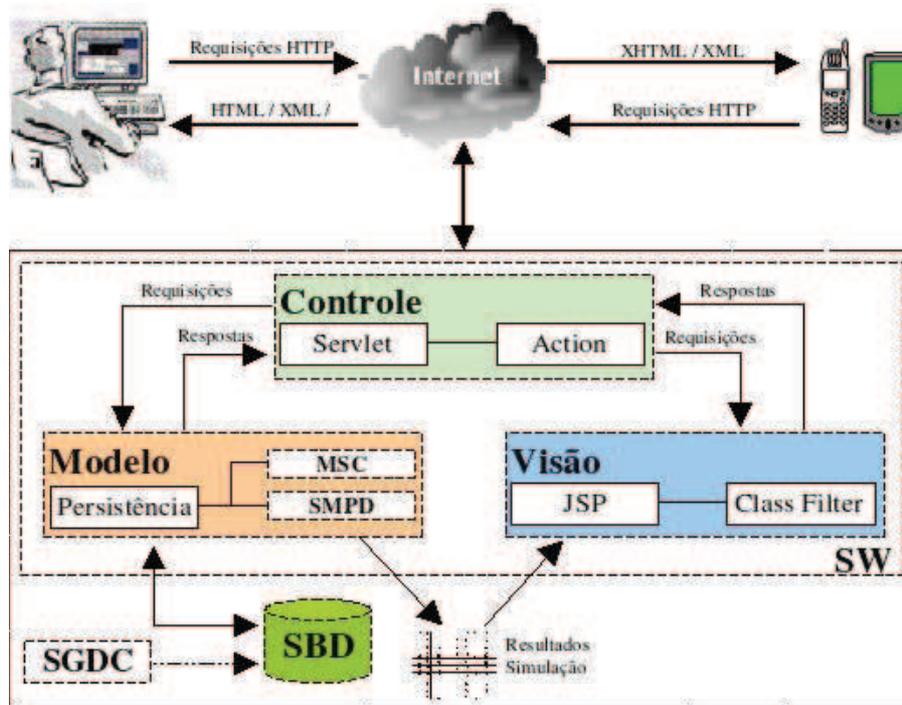


Figura 7: Modelo de simulação de doenças de plantas (MSP)

mais. Estes servidores foram divididos em: (i) servidor de gerenciamento de dados climáticos (SGDC), (ii) servidor de banco de dados (SBD), (iii) servidor de modelos de previsão de doença (SMPD), (iv) servidor Web (SW) e (v) servidor de modelo de simulação de culturas (MSC).

A maneira como são organizados os componentes dentro da estrutura MVC oferece uma grande confiabilidade, além de facilitar o desenvolvimento e capacitar a execução em cada um separadamente, possibilitando a substituição de um determinado componente a qualquer instante sem que haja a necessidade de alteração no sistema como um todo. Por exemplo, se houver a necessidade de substituição de um determinado modelo de simulação, isso pode ser feito rapidamente e sem muito esforço.

O desenvolvimento da estrutura foi baseado no framework Struts, dividindo-se a programação em três camadas: camada de controle, camada de visão e camada de modelo. Cada componente deste sistema foi programado através de classes Java, encapsulando as operações e atributos de cada um.

O funcionamento do sistema dá-se de forma paralela e dinâmica, com cada componente sendo controlado e gerenciado pelas regras definidas em cada servidor. Alguns componentes são invocados por agendamentos feitos pelos programadores, ao passo que outros obedecem aos eventos gerados por outros componentes ou pelos usuários do sistema, em tempo de execução. Com a estrutura adotada, observa-se uma certa autonomia, característica esta buscada pela tecnologia de objetos.

O fluxo da execução do sistema é mediado pelo controlador central do Struts, o qual é responsável por delegar as requisições para os tratadores apropriados, localizados no modelo, os quais representam a lógica e o estado do sistema. A requisição é respondida através do controlador e apresentada na visão de maneira adequada, sendo as respostas orientadas por meio de mapeamentos, os quais são carregados de arquivos de configuração XML, levando a que haja dependência entre a visão e o modelo, auxiliando na criação, expansão e manutenção do sistema.

Com a visão de que um sistema deva ser decomposto em partes coesas, mas fracamente acopladas (modularidade), e que permita a substituição de diferentes versões dos módulos a qualquer tempo, todas as classes desenvolvidas seguiram um padrão de modularidade, o qual foi definido e descrito por Porter et al. (1999). Cada um dos módulos foi subdividido a fim de que refletisse melhor as necessidades

do modelo, separando, dessa forma, o código em inicialização, cálculo de taxas, integração e saída (Figura 8).

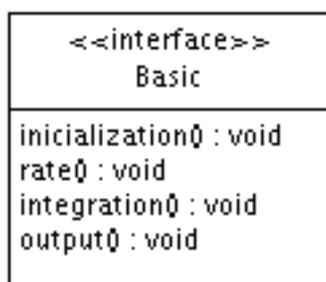


Figura 8: Interface implementada pelas classes do sistema

Tendo como princípio básico a utilização de linguagem orientada a objetos neste projeto, a persistência dos dados foi realizada utilizando-se em todos os módulos o *framework* Hibernate, o qual é responsável por armazenar e recuperar objetos do banco de dados, possibilitando uma maior transparência na camada de persistência e deixando o sistema mais flexível quanto a uma substituição do banco de dados (BAUER; KING, 2005).

SGDC

O servidor SGDC é responsável pela obtenção e armazenamento de dados climáticos, disponibilizados de diversas formas e locais, como pelas estações meteorológicas automáticas dispostas no território nacional, tanto de propriedade privada como de propriedade do governo brasileiro.

Os dados coletados são encontrados em diferentes formatos, desde XML (Extensible Markup Language), texto, HTML (HyperText Markup Language), arquivos binários, entre outros. Esses dados são processados a fim de se analisar sua veracidade e integridade; após esse

processamento, são catalogados, indexados e armazenados no banco de dados com o objetivo de serem facilmente recuperados.

Este servidor possui uma estrutura composta por uma variedade de pequenos aplicativos, cada um com uma tarefa específica a realizar, como os “robôs de busca de dados” disponíveis na Web, os coletores de dados de estações meteorológicas automáticas, os verificadores de integridade e veracidade dos dados, os sumarizadores de variáveis climáticas e os transportadores de dados para o SBD (Figura 9). Todos os aplicativos são gerenciados por um agendador de tarefas, o qual se utiliza do *cron*¹ do sistema operacional Linux para controlar a ordem e o horário de cada tarefa.

Um aplicativo especial destaca-se neste servidor, o coletor de dados de prognóstico climático, o qual é composto por módulos responsáveis por efetuar o download de arquivos binários, resultantes da execução dos modelos de circulação global (modelo Eta²), e processar o seu conteúdo de acordo com coordenadas geográficas previamente estabelecidas (estações meteorológicas e municípios), obtendo os dados horários necessários para os modelos de simulação. As variáveis prognósticas do modelo são: precipitação pluviométrica, temperatura do ar, umidade relativa, pressão atmosférica e velocidade e direção do vento.

O SGDC, por meio de rotinas específicas, oferece aos modelos de simulação o suporte necessário para a realização da importação tem-

¹Recurso padrão dos sistemas Unix (ou Linux) que permite o agendamento de tarefas para serem executadas num momento específico ou em intervalos regulares.

²ETA - Modelo de mesoscala, em ponto de grade, de equações primitivas. Nome oriundo da letra grega η (Eta). No Brasil, roda operacionalmente no Centro de Previsão de Tempo e Estudos Climáticos (CPTEC), sendo hidrostático e cobre a maior parte da América do Sul e oceanos adjacentes. A sua resolução horizontal atual é de 40 km e a vertical, de 38 camadas (CPTEC, 2007).

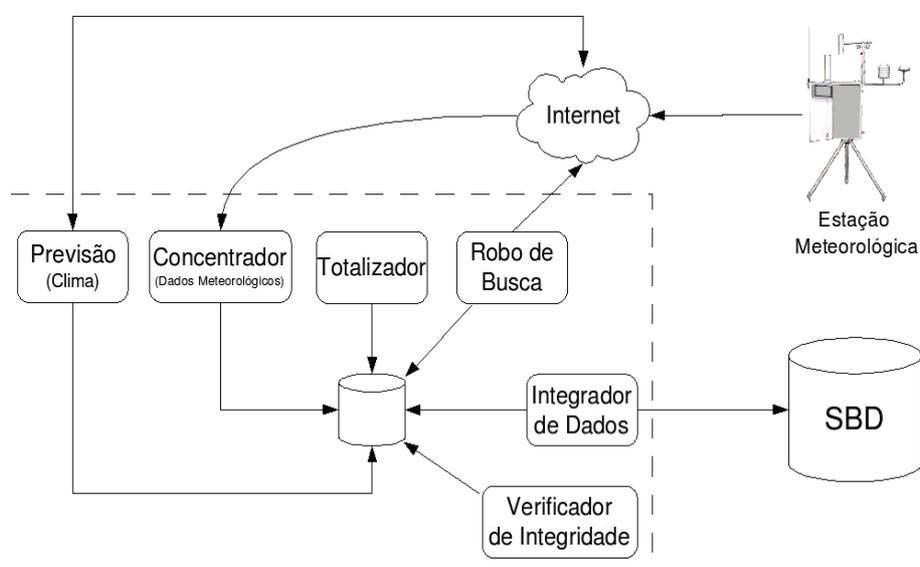


Figura 9: Servidor de Gerenciamento de Dados Climáticos (SGDC)

porária de dados climáticos, disponibilizados na forma de arquivos texto, no momento da execução dos modelos, sendo eliminados ao término da seção do usuário ou armazenados permanentemente, se assim for a decisão do usuário. Esses dados, no momento da importação, são repassados diretamente para o SBD.

SBD

O servidor SBD é responsável pela disponibilização dos dados armazenados aos demais servidores e modelos através de requisições SQL (Structured Query Language), sendo projetado e organizado para armazenar informações advindas das mais variadas fontes, formas e usuários. Os dados podem ser públicos, disponíveis a todos os usuários, ou de uso privado, apenas para o “proprietário” dos dados.

O SBD foi preparado para ser capaz de gerenciar todos os dados de

entrada necessários para a execução dos modelos de simulação, além de armazenar as saídas dos modelos. Gatilhos foram programados para realizar a limpeza do banco de dados, eliminando-se os dados não mais necessários e mantendo-se apenas aqueles selecionados pelo próprio usuário.

Para garantir o direito de propriedade dos dados de cada usuário o SBD foi organizado de tal maneira que para toda e qualquer informação inserida no banco de dados seja necessário informar a quem pertence, possibilitando, dessa forma, que as buscas por dados para as futuras execuções dos modelos sejam facilitadas.

Todas as conexões ao banco foram limitadas àquelas que estejam sendo realizadas na própria máquina ou dentro da estrutura definida (rede interna), evitando, assim, a tentativa de acesso indevido aos dados. Políticas de segurança, como criptografia de senhas e relacionamentos entre as entidades modeladas, foram implementadas com o intuito de minimizar as possibilidades de exceções.

SMPD

O SMPD é um servidor que tem como responsabilidade fornecer informações sobre o resultado da simulação sobre determinadas pragas de uma cultura; pode ser substituído por qualquer modelo de simulação que siga o padrão estabelecido, possuindo os métodos necessários para o acoplamento com os demais módulos.

Pode-se citar como exemplo o modelo criado e desenvolvido por Emerson Del Ponte (DELPONTE et al., 2004), o qual simula epidemias de uma doença que se destaca no cultivo do trigo, conhecida por giberela, causada pelo fungo *Gibberella zeae*, a qual está relacionada

com eventos climáticos como temperatura, precipitação e umidade relativa, principalmente no estágio fenológico de floração.

Este modelo segue o padrão definido e explicado anteriormente, possuindo a interface básica para que pudesse ser acoplado ao sistema. O modelo, a cada passo da aplicação, no período que compreende a fase da floração, executa rotinas internas calculando e fornecendo informações sobre a severidade da doença. Esta informação é de fundamental importância para alimentar o simulador da cultura (MSC), afetando processos internos e ocasionando modificações nos resultados simulados.

Estratégias de controle podem ser definidas no sistema, de modo a servir de entrada para os modelos, interferindo nos processos de simulação. Um exemplo dessas estratégias de controle é o uso de fungicidas sob certas condições de pressão da epidemia (DELPONTE et al., 2004).

SW

O servidor SW é o responsável por disponibilizar acesso via Web, através do servidor de aplicações Apache Tomcat (<http://www.apache.org>), de forma restrita ou irrestrita, de acordo com as regras implementadas para cada módulo do sistema. É um servidor de aplicações desenvolvido pela Fundação Apache utilizando tecnologia Java, tendo a habilidade de converter, automaticamente, qualquer programa JSP num Servlet equivalente, isto é, capaz de criar código fonte Java a partir de um documento HTML.

O desenvolvimento foi feito sob a arquitetura MVC e implementado sob o framework Struts, possibilitando o controle e a separação

das camadas da aplicação. Cada camada é tratada de forma independente, podendo ser reaproveitada caso já tenha sido desenvolvida.

Este servidor, quando se trata de interface Web, é o responsável por prover todo acesso às aplicações, possibilitando a sua adequada execução. Quando uma requisição do usuário é realizada, o Tomcat repassa-a para o controle do Struts, deixando a cargo deste o fluxo da execução da aplicação, permanecendo à espera de alguma nova requisição externa ou por parte do Struts.

Com relação à visão, foi implementada utilizando-se tecnologias como JSP, JSF, TagLibs, JavaScript, AJAX, entre outras, possibilitando uma boa organização de código, boa qualidade na apresentação e na execução das mesmas, oferecendo aos usuários um sistema dinâmico e reciclável, atualizando apenas as informações necessárias. O uso destas tecnologias levou a que o sistema ficasse amigável e leve (tempo de download e upload curto).

MSC

O servidor MSC é responsável por interagir com o servidor SMPD, recebendo informações sobre a dinâmica da praga e fornecendo os dados sobre os estádios da cultura, além de outras informações, como a área foliar existente num determinado momento. Essa interação é de fundamental importância, tendo em vista que os processos da dinâmica de populações e de desenvolvimento da planta dependem de informações como estas.

A implementação do MSC deu-se com a realização de alterações no modelo de simulação Cropsim-Wheat, tendo como objetivo a disponibilização de seus serviços na Web, executando-o em ambiente Li-

nux, sabendo-se que a sua versão original é para execução em sistema Windows® e com processamento local. No modelo Cropsim-Wheat foram realizadas as alterações necessárias para que pudesse ser executado e manipulado por meio da linguagem R, a qual é a ponte para os demais servidores deste sistema (Figura 10).

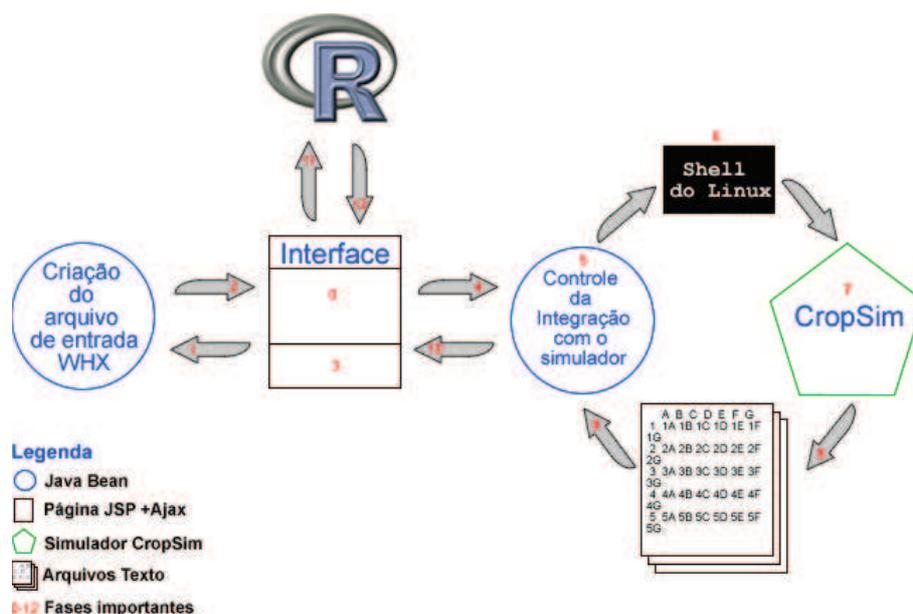


Figura 10: Integração R x CropSim-Wheat

Para possibilitar a integração entre a linguagem Java e a linguagem R, utilizou-se o Rserve, um servidor TCP/IP para a linguagem R, pelo qual se tornou possível que os módulos desenvolvidos em Java pudessem se utilizar das facilidades da linguagem R sem precisar iniciar uma sessão do R ou ligá-los a alguma biblioteca específica (Figura 11). Todas as conexões realizadas são separadas em espaços de trabalho (diretórios) diferentes no servidor. Com o uso do Rserve, possibilitou-se fazer conexões remotas ao R com o intuito de realizar a computação de modelos estatísticos, geração de gráficos e, principalmente, executar os modelos de simulação desenvolvidos em Fortran.

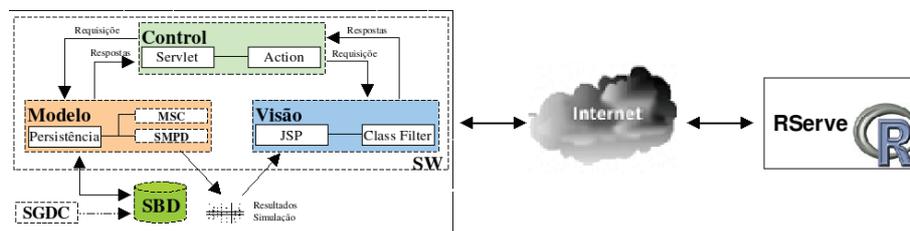


Figura 11: Integração Modelo x RServe

Para a realização das ligações entre os módulos dos servidores com o RServe utilizou-se uma biblioteca de classes denominada Rcli-ente, a qual realiza a integração da linguagem Java com o servidor TCP/IP RServe (Figura 12), responsável pela execução dos comandos da linguagem R, permitindo a ligação entre a heterogênea lista de tecnologias envolvidas. Obtiveram-se como resultado os dados necessários para certos servidores, como área foliar, estádios fenológicos etc.

```
// Conexão com o RServe
Rconnection c = new Rconnection();
// Execução de operações entre vetores
ArrayList sumArray = new ArrayList();
c.eval("x=0:10");
c.eval("y=40:50");
sumArray.add(eval("x+y"));
int max = c.eval("max(x)");
```

Figura 12: Executando código R na linguagem Java.

Além de conteúdos resultantes da integração entre os servidores e os modelos, a geração de gráficos bem acabados tornou-se possível em tempo de execução, para publicação direta. Com a interação entre tecnologias como AJAX e as de ligação dinâmica, já mencionadas, tornou-se possível a construção de interfaces amigáveis e de fácil

manipulação.

Discussão

Com a estrutura aqui apresentada, foi possível organizar e reutilizar conhecimentos construídos durante as últimas décadas sem a necessidade de alterações bruscas nos mesmos, melhorando a performance e a forma de disponibilização, não mais centrada em aplicações fechadas, mas abrangendo ambientes abertos e de acesso em massa.

A reutilização de modelos de simulação amplamente testados e validados torna as aplicações mais robustas no que diz respeito à simulação do crescimento e desenvolvimento de culturas, tendo em vista que o foco dos estudos fica direcionado ao problema em si, não à estrutura computacional.

Verificou-se a importância da aplicação de técnicas e padrões no desenvolvimento de sistemas complexos como os modelos de simulação, tornando-os mais robustos, bem documentados, reusáveis, compatíveis e facilmente acopláveis a outras soluções já existentes. A fácil manutenção e/ou substituição de módulos problemáticos assegura a qualidade e confiabilidade dos resultados. As tecnologias utilizadas permitem uma fácil conexão com bancos de dados, serviços disponíveis na Web, geradores de interfaces gráficas, entre outras.

A estrutura desenvolvida e apresentada permite a sua reutilização e aplicação para qualquer outro modelo de simulação integrante da suíte do DSSAT, tornando possível a integração entre diferentes modelos, maximizando a capacidade de desenvolvimento e facilitando a disponibilização dos resultados.

CAPÍTULO II

DESENVOLVIMENTO DE MODELOS DE SIMULAÇÃO DE DOENÇAS UTILIZANDO UML E ORIENTAÇÃO A OBJETOS

Willingthon Pavan

Faculdade de Agronomia e Medicina Veterinária

Universidade de Passo Fundo

C.P. 611, 99001-970

RESUMO

Nas últimas décadas, a modelagem matemática de doenças de plantas tem se expandido muito rapidamente, originando diferentes metodologias de desenvolvimento de modelos de simulação. Os recentes avanços na ciência da computação têm auxiliado nesse desenvolvimento, trazendo vantagens e facilidades operacionais para a construção de modelos que representem os complexos patossistemas agrícolas, de uma forma mais genérica. A utilização de modelos que simulam o crescimento e desenvolvimento de uma cultura e, ao mesmo tempo, contabilizam o impacto das doenças pode ser determinante para o auxílio à tomada de decisões. A produção de trigo no Brasil apresenta-se como um sistema complexo, afetado por um conjunto de fatores bióticos e abióticos, exercendo grande influência sobre o rendimento. Portanto, um dos grandes desafios da ciência agrônoma é a análise objetiva desses fatores, que, na maioria das vezes, atuam simultaneamente. Este trabalho teve como objetivo o desenvolvimento de um modelo genérico capaz de representar epidemias de doenças num hospedeiro em crescimento. Foi ilustrando o patossistema *Puccinia-Trigo* como um caso especial de simulação de epidemia acoplada a

um modelo de crescimento e desenvolvimento do trigo. A dinâmica do modelo de epidemia da ferrugem da folha foi comparada com dados observados. O modelo de simulação do patossistema *Puccinia*-Trigo simulou corretamente o crescimento da área foliar do trigo e progresso da ferrugem da folha, apresentando-se como uma alternativa para o estudo das relações entre o patógeno e o hospedeiro, podendo ser utilizado para a avaliação de danos provocados pela doença.

ABSTRACT

DEVELOPMENT OF DISEASE SIMULATION MODELS USING UML AND OBJECT-ORIENTED APPROACH

In the last few decades, the mathematical modeling of plant diseases, for example, has expanded very quickly, contributing for disease integrated management. The recent advances in the computer science have assisted in this development, bringing operational advantages and usefulness for the construction of models representing complex processes such as disease epidemics. Crop models that simulate the growth and development and at the same time account for disease impact on yield have a potential use in decision support systems for diseases management. The wheat production in Brazil is presented as a complex production system affected by a set of biotic and abiotic stresses which exerts great influence on grain yield and quality. Therefore, one of the great challenges for the agriculture science is to quantify these factors, that, most of the time, act simultaneously. This work had as objective the development of a generic model capable to represent disease epidemics in a growing host. It was illustrated pathosystem *Puccinia*-Wheat as a special case of simulation of an epidemic coupled to a wheat growth model. The dynamics of leaf rust epidemics was compared to observed data. The simulation model of the pathosystem *Puccinia*-Wheat reasonable well, representing leaf rust progress accordingly to the wheat leaf area growth. It can be used as a tool for studying pathogen/host relationship and for damage quantification.

Introdução

Diferentes modelos e metodologias de desenvolvimento de modelos de simulação têm sido extensivamente utilizados em epidemiologia. Estes incluem curvas descritivas do progresso de epidemia (CAMPBELL; MADDEN, 1990; GILLIGAN, 1990; BERGAMIN FILHO et al., 1995a; BOWEN, 1997) como também os simuladores de epidemias (WAGGONER; HORSFALL, 1969; ZADOKS, 1989; ROSSI et al., 1997; MAY et al., 2005).

A modelagem matemática das doenças das plantas é uma disciplina que se expande rapidamente na área da fitopatologia (MAANEN; XU, 2003b). Van der Plank, com o seu livro *Plant diseases: epidemics and control*, publicado em 1963 (PLANK, 1963), foi o pioneiro na introdução dos conceitos básicos que ainda hoje alicerçam a representação matemática das epidemias. O modelo logístico foi adaptado por Van der Plank para representar, de forma mais realística, as epidemias, uma vez que contabiliza a quantidade de tecido do hospedeiro em fase latente, infecciosa e removida.

Os modelos epidemiológicos podem ser classificados como descritivos, preditivos e conceituais. Os modelos descritivos servem para testar hipóteses em resultados experimentais sem, necessariamente, revelar o mecanismo, podendo prever a ocorrência ou a severidade de uma epidemia. Os modelos conceituais ou teóricos permitem a identificação do mecanismo separando as causas e os efeitos. Na construção do modelo conceitual procura-se representar os processos biológicos que governam a epidemia (MAANEN; XU, 2003b). A combinação de modelos pode levar à construção de modelos de simulação bastante complexos, os quais podem ser específicos para uma determinada doença ou genéricos.

O desenvolvimento de um modelo de simulação de epidemias de forma genérica apresenta vantagens operacionais. Os recentes avanços na ciência da computação possibilitam o desenvolvimento de modelos de simulação complexos, ainda que construídos de forma genérica. O uso de linguagens computacionais orientadas a objetos é adequado para o desenvolvimento de modelos de simulação com elevado nível de complexidade e modularidade (PORTER et al., 1999; HOOGENBOOM, 2000; JONES et al., 2001).

A importância da interação entre os patógenos e o hospedeiro foi negligenciada durante algum tempo (GILLIGAN, 1985), mas, felizmente, nos últimos anos vem recebendo a devida atenção (PINNSCHMIDT et al., 1995; WEBB et al., 2000; SAVARY et al., 2006). Por outro lado, os modelos de simulação do desenvolvimento e crescimento de culturas geram saídas que seriam o “alvo” do patógeno. É óbvio e natural que modelos de simulação de epidemias sejam acoplados como um subsistema do modelo de crescimento da cultura. Na literatura, há relatos de esforços feitos nesse sentido (BASTIAANS, 1993; WILLOCQUET et al., 2002).

Um modelo que simula o crescimento de uma cultura e, ao mesmo tempo, contabiliza o impacto das doenças pode ser muito importante para aquelas culturas e regiões onde a ocorrência da doença é um fator determinante para a viabilidade econômica da cultura. Este caso especial em muito se assemelha à cultura do trigo no Brasil.

O trigo é uma cultura que sofre as ameaças externas da globalização dos mercados e, ainda, é bastante influenciada pela variabilidade climática; sua produção, no Brasil, ocorre em maior parte na região Sul, abrangendo os estados do Paraná, Rio Grande do Sul e Santa Catarina. Também é nesses estados que a cultura está sujeita às mai-

ores adversidades climáticas, associadas à influência de fenômenos de grande escala ENOS (El Niño Oscilação Sul). A presença das fases dos ENOS (EL Niño e La Niña) altera significativamente a variabilidade interanual dos elementos meteorológicos.

Em anos de ocorrência de El Niño observam-se precipitações pluviiais acima da média histórica para o sul do Brasil, principalmente na primavera e no início do verão (período de maturação e colheita do trigo), ao passo que em anos de La Niña a tendência é que ocorram estiagens no mesmo período para a região (BERLATO; FONTANA, 2003; BERLATO et al., 2005).

Além dos fatores climáticos, a alegada falta de competitividade na cultura do trigo nos sistemas de produção tem sido uma das maiores dificuldades para que o Brasil se firme, de fato, como um país produtor de trigo. Os problemas são atinentes a aspectos da produção nacional, mas também têm relação com a importação de trigo de países que oferecem subsídios aos seus produtores e à cadeia de comercialização de trigo.

Entre os fatores que limitam a produção dos cereais de inverno, como o trigo, no Brasil destaca-se a ocorrência de um elevado número de doenças, como o oídio (*Blumeria graminis* f. sp. *tritici*), as ferrugens da folha e do colmo (*Puccinia triticina* e *Puccinia graminis* f. sp. *tritici*), a mancha da gluma (*Stagonospora nodorum*), a mancha marrom (*Bipolaris sorokiniana*), a mancha amarela (*Drechslera tritici-repentis*) e a giberela (*Gibberella zeae*) (PICININI; FERNANDES, 1995, 2003). Caso medidas de controle não sejam postas em prática, a cultura de trigo pode perder, em média, nas condições do sul do Brasil, 44,6% na produtividade de grãos (PICININI et al., 1996).

Entre as medidas de controle mais eficientes e de mais fácil adoção encontra-se a resistência genética das plantas às doenças. O conhecimento acumulado na manipulação gênica visando à obtenção de cultivares resistentes tem propiciado a disponibilidade de inúmeros cultivares resistentes. Infelizmente, mudanças na população patogênica e o surgimento de novas doenças acabam por alterar o comportamento dos cultivares. Assim, o uso de fungicidas tem sido preconizado como uma medida emergencial para diminuir o impacto das doenças no rendimento do trigo. Entretanto, esta prática onera o custo de produção, além de os produtos químicos serem agentes poluentes (PICININI et al., 1996).

A melhoria dos sistemas de produção de trigo no Brasil não é uma tarefa fácil, visto que trata de um sistema de produção muito complexo. Interações entre fatores do ambiente físico e de ordem socioeconômica acabam determinando o resultado final e condicionando o desempenho das explorações desse cereal em terras brasileiras. Em essência, seus problemas são de natureza sistêmica, não disciplinar e, como tal, sua solução requer também o uso de um enfoque de pesquisa transdisciplinar. Por essa característica, o mais adequado parece ser o uso do enfoque sistêmico, via técnicas de modelagem e simulação e teorias de decisão, integrando conhecimentos de diversas áreas e analisando possibilidades que possam otimizar a produção de trigo no Brasil, quer seja no planejamento agrícola, quer no manejo da cultura.

O sistema de produção de trigo no Brasil é muito complexo, incluindo interações entre fatores bióticos e abióticos. Alguns desses podem ser modificados via manejo da cultura, entretanto o clima ainda exerce a maior influência sobre o rendimento final de trigo. Um dos grandes desafios da ciência agrônoma é a análise objetiva de um

conjunto de fatores que atuam simultaneamente. Finalmente, combinar num único modelo a capacidade de predizer o crescimento e o desenvolvimento do trigo no sul do Brasil e, ainda, contabilizar os impactos em razão da ocorrência de doenças é uma demanda ainda não atendida pela pesquisa.

O objetivo deste trabalho foi (i) desenvolver um modelo genérico capaz de representar epidemias; (ii) usar o patossistema *Puccinia-Trigo* como um caso especial do modelo genérico; (iii) acoplar o modelo de epidemia de ferrugem da folha a um modelo de crescimento do trigo; (iv) avaliar a dinâmica do modelo de epidemia da ferrugem da folha.

Teoria e Metodologia

Características do Ciclo de Vida do Patógeno

O desenvolvimento epidemiológico dos patógenos varia de um ano para o outro de acordo com as necessidades ambientais de cada espécie, as quais estão estreitamente relacionadas com as condições climáticas regionais (FERNANDES, 2000).

As condições ambientais no Rio Grande do Sul são extremamente instáveis tanto no ano como entre anos, determinando diferenças no grau de intensidade das doenças predominantes em cada região. Os dois fatores ambientais mais importantes são o período de molhamento foliar, representado pela água livre na superfície das plantas, seja por chuva, orvalho ou neblina, e que é essencial ao surgimento das doenças, e a temperatura ambiente, que atua no sentido de retardamento ou aceleração do processo infeccioso (Michel & Svoboda, 1999).

Desenvolvimento do Patógeno

Esporos de fungos foliares, após serem depositados na superfície do hospedeiro, devem ali se fixar para, posteriormente, emitir o tubo germinativo e iniciar a penetração. A adesão dos esporos à superfície do hospedeiro é um fenômeno muito comum, essencial para o processo de infecção de muitas espécies de fungos. De modo geral, trata-se de um processo ativo que envolve secreção de fluidos capazes de alterar a superfície do hospedeiro, favorecendo a infecção (NICHOLSON et al., 1991). No processo de adesão ocorre o primeiro reconhecimento entre patógeno e hospedeiro.

Para iniciar a germinação, esporos de fungos necessitam de condições favoráveis de ambiente, tal como umidade elevada e requerimentos específicos de temperatura e luz. O final do processo infeccioso é caracterizado pelo estabelecimento de relações parasitárias estáveis entre o patógeno e seu hospedeiro. Tem início nesta fase o parasitismo propriamente dito, com a retirada de nutriente da planta pelo agente patogênico. Este ponto marca a transição entre infecção e colonização.

A duração da colonização, ou seja, do processo que tem início no estabelecimento de relações parasitárias estáveis e termina com a reprodução do patógeno, é difícil de ser quantificada, em decorrência de não ser possível medir com precisão o momento em que relações parasitárias estáveis são estabelecidas.

Período Latente

Para a determinação do período de patogênese de um organismo utiliza-se o período de latência, definido como o período de tempo

decorrido entre a inoculação (contato entre patógeno e hospedeiro) e o aparecimento de estruturas reprodutivas do patógeno.

O período latente de um microrganismo é muito utilizado em trabalhos epidemiológicos. Em doenças policíclicas, o período latente é um dos parâmetros que mais influenciam a velocidade de crescimento da epidemia. Este período representa, em termos epidemiológicos, o tempo da geração da espécie patogênica. Quanto maior o tempo decorrido entre a inoculação e a reprodução, menor o número de gerações produzidas por ciclo do hospedeiro; o raciocínio inverso também é válido (AMORIN et al., 1995).

Período Infeccioso

Infecção é definida por Gäumann (1950) como o processo que se estende desde a germinação até o estabelecimento de relações parasitárias estáveis com o hospedeiro. Em outras palavras, o processo de infecção engloba os mecanismos de pré-penetração, penetração e estabelecimento de relações parasitárias. Já o período infeccioso é compreendido entre o início e o final da produção de esporos da lesão ou urédia (PIVONIA; YANG, 2006).

Período Necrótico

O período necrótico é definido como sendo o momento em que uma lesão é incapaz de produzir novos propágulos (ABBAS et al., 2004; MIKLER et al., 2005), ou seja, incapaz de infectar outras áreas ou outros indivíduos. Abrange todos aqueles que não mais contabilizam para a infecção, pois passaram do período infeccioso (BERGER et al., 1997).

Expansão da Lesão

Trabalhos que reconhecem a importância do crescimento da lesão no desenvolvimento de uma epidemia são escassos tanto na literatura tropical quanto na temperada. Berger (1973) citado por Bergamin Filho e Amorim (2001), ao monitorar *Helminthosporium turcicum* na Flórida, foi um dos pioneiros nessa área, tendo constatado que “poucas lesões novas aparecem após o florescimento (...) e a maior parte do aumento da doença deveu-se ao crescimento das lesões”. Emge et al. (1975), também citados por Bergamin Filho e Amorim (2001), trabalhando em região tipicamente temperada (Oregon, EUA), deixaram claro o papel desempenhado pelo crescimento da lesão no desenvolvimento de *Puccinia striiformis* em trigo, relatando que o crescimento da lesão permite a intensificação da doença na ausência de condições favoráveis para a reinfeção e afetando profundamente a epidemiologia da doença.

Segundo Bergamin Filho e Amorim (1996), a quantidade de tecido doente que produz esporos praticamente não exerce influência sobre a severidade em virtude da expansão das lesões. Essa severidade devida à expansão das lesões representa a proporção de sítios ocupados exclusivamente pelo crescimento das lesões, ignorando-se seu tamanho original. Quanto maior o tamanho inicial da lesão, menor é o número de sítios sadios remanescentes, passíveis de serem ocupados pelo crescimento dessas lesões. O efeito da expansão das lesões é mais significativo em sistemas com latência prolongada, pelo maior tempo disponível para a lesão crescer. Nos sistemas que têm lesões iniciais grandes, a epidemia atinge seu máximo mais rapidamente e a contribuição da expansão das lesões na severidade final torna-se menor.

A baixa produção de esporos dos patógenos tropicais não constitui uma desvantagem evolutiva; ao contrário, é uma sábia estratégia, pois a energia para a sobrevivência é deslocada para o crescimento das lesões, este, sim, um parâmetro diferencial no progresso epidêmico. Para valores elevados de taxa de expansão das lesões, a taxa básica de infecção mostrou pouco efeito na severidade, reiterando que a taxa de expansão das lesões é um parâmetro bastante sensível e de grande importância na evolução de epidemias tropicais.

O fato de grande parte das doenças foliares importantes nos trópicos e subtropicais basear sua capacidade epidêmica mais no crescimento da lesão do que no aparecimento de novas lesões, traz conseqüências importantes em relação ao controle. Considere-se, como exemplo, o melhoramento genético visando à resistência: Rاپilly e Jolivet (1976) e Rاپilly (1988), trabalhando com *Septoria nodorum*, trigo e um modelo de simulação, demonstraram que o parâmetro epidemiológico que mais contribui para a elevação da taxa aparente de infecção é a velocidade de crescimento da lesão. O período latente, visto como um dos parâmetros mais importantes nos clássicos estudos de patossistemas temperados (ZADOKS; SCHEIN, 1979), mostrou desempenhar um papel bastante secundário nesse contexto.

Em epidemias naturais, a taxa de expansão da lesão pode ser variável dependendo da temperatura (KATO; KOZAKA, 1974; SUBBARAO; MICHAILIDES, 1995; VITTI et al., 1995; PRATES; FERNANDES, 2001; ALVES; FERNANDES, 2006), ou da posição da folha (ECKHARDT et al., 1984), reduzindo-se a velocidade com a idade da lesão ou hospedeiro (SHANER, 1983; KOCH; MEW, 1991), ou com o aumento da densidade de lesões (SHANER, 1983), ou aumentando com a idade (SHOEMAKER; LORBEER, 1977; VITTI et al., 1995; HOOD; SHEW, 1996), ou com longos períodos de orvalho

(SHOEMAKER; LORBEER, 1977; VITTI et al., 1995; EVERTS; LACY, 1996). Mehta (1981) relatou que uma simples lesão de *Bipolaris sorokiniana* num período de 33 dias ocupou 23% da área de uma folha de trigo.

Impacto do Patógeno sobre o Hospedeiro

Obter dados para estabelecer a função de dano é o objetivo da fase experimental de um projeto de pesquisa que vise à determinação dos prejuízos causados por um patógeno. A função de dano relaciona dano (D) com injúria (I), através da equação: $D = f(I)$, ao passo que a função de perda relaciona perda (P) com dano (D), onde: $P = f(D)$. A equação matemática que descreve a função de dano pode ser estabelecida com razoável precisão para qualquer patossistema. Os parâmetros variarão de acordo com o cultivar, local, tipo de solo e muitos outros fatores. A função de perda é mais difícil de ser determinada, pois depende de aspectos econômicos de cada região ou país.

O fracasso na obtenção da função de dano em alguns patossistemas não é surpreendente para (WAGGONER; BERGER, 1987; BERGAMIN FILHO et al., 1995b), que postulam dever-se a produção vegetal à absorção da área foliar sadia do hospedeiro, não da área destruída pelo patógeno. Em culturas nas quais o número de folhas é fixo, como nos cereais, área foliar sadia e área foliar doente são altamente correlacionadas, razão por que a função de dano tem sido usada com êxito em cereais.

Modelo de crescimento e desenvolvimento

O DSSAT é uma das mais conhecidas e difundidas aplicações dos modelos de simulação do crescimento e desenvolvimento das cultu-

ras, sendo desenvolvido pelo projeto IBSNAT (International Benchmark Sites Network for Agrotechnology Transfer) junto à Universidade do Havaí a partir de 1974. O projeto IBSNAT teve duração de cerca de dez anos, resultando no aplicativo DSSAT e numa rede internacional de pesquisadores envolvidos no desenvolvimento e em teste de modelos de simulação. Atualmente é mantido pelo consórcio ICASA (International Consortium for Agricultural Systems Applications - <http://icasa.net>), que reúne um grande número de cientistas de diversas universidades e centros de pesquisa, os quais trabalham de forma colaborativa para a construção dos modelos e suas aplicações (JONES et al., 2001, 2003; ICASA, 2007).

O DSSAT é um sistema composto por diversos modelos de simulação, estes orientados a processos, projetado para aplicações globais e que trabalha independentemente de local, estação e cultivar (VERHAGEN et al., 2001). É apropriado para que os estudos a longo prazo avaliem as estratégias eficientes da gerência da cultura e otimizem a produção (FARIA; BOWEN, 2003). O DSSAT combina dados de solo e de clima com modelos de culturas e aplicativos, a fim de simular os resultados de vários anos de estratégias no manejo de culturas (ICASA, 2007); é o resultado do trabalho colaborativo de muitos cientistas de diversas universidades e centros de pesquisa que fazem parte do consórcio .

Como um pacote de software que integra os efeitos do solo, características da espécie, coeficientes genéticos, dados climáticos e opções de manejo, o DSSAT permite que os usuários possam solicitar resposta a perguntas como “O que ... se ...?” e simular os resultados de experimentos em minutos num computador, o que poderia consumir uma parte significativa da vida profissional de um agrônomo (ICASA, 2007). A comparação das saídas dos modelos com os resultados ob-

servados faz parte da validação de um modelo.

O DSSAT é uma coleção de programas independentes que interagem entre si, na qual os modelos de simulação de culturas ficam no centro (Figura 13), ao passo que as bases de dados descrevem o clima, solo, observações e condições experimentais, além de informações sobre genótipos para aplicar nos modelos em diferentes situações (JONES et al., 2003). Os componentes de software são escritos em:

- FORTRAN (modelos);
- C (shell);
- Pascal (gráficos);
- Dbase (base de dados) e
- Basic (programas de gerenciamento de estratégia e risco).

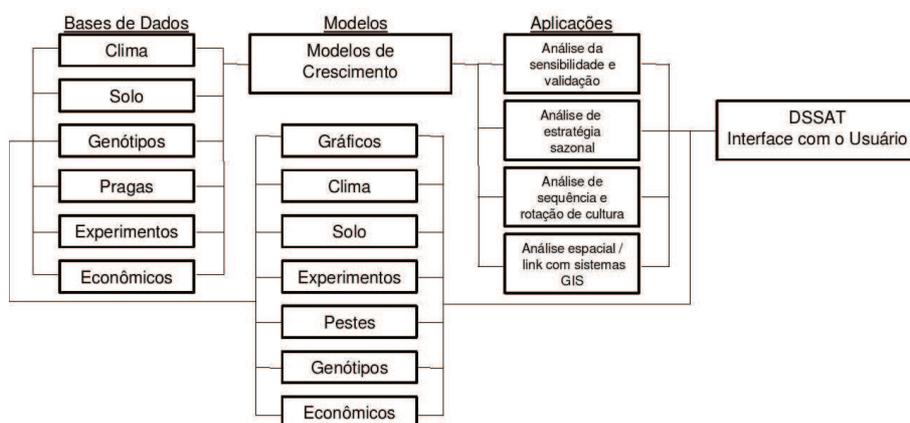


Figura 13: Estrutura do Modelo DSSAT (Adaptado de Jones et al. (2003))

O DSSAT fornece também a possibilidade de validação das saídas dos modelos, permitindo, assim, que os usuários possam comparar os

resultados da simulação com os resultados observados. Esta validação se dá pela entrada de dados por parte do usuário (dados observados), executando o modelo e comparando as saídas obtidas. Simulando resultados prováveis de estratégias da gerência da cultura, oferece ao usuário as informações necessárias para apreciar novas culturas, produtos e práticas a fim de adotá-las (ICASA, 2007).

Atualmente, a suíte DSSAT é composta por 27 diferentes modelos de culturas, formada por uma coleção de programas independentes que operam em conjunto, estando inseridos numa ferramenta que facilita a criação e o gerenciamento de arquivos de experimentos, de solo e de clima. Inclui aplicações que auxiliam na análise sazonal e seqüencial dos riscos econômicos e impactos ambientais associados com irrigação, uso de fertilizantes e nutrientes, mudanças climáticas, seqüestro de carbono do solo, variabilidade climática e agricultura de precisão (JONES et al., 2003; ICASA, 2007).

O DSSAT simula o crescimento, desenvolvimento e produção de uma cultura considerando numa área uniforme de terreno com as informações de manejo informadas ou simuladas, assim como mudanças no conteúdo da água do solo, carbono e nitrogênio que ocorrem em relação ao tempo. Possui uma estrutura composta por um programa principal, uma unidade módulo solo e os módulos para os componentes primários. A unidade módulo solo é responsável pela interface entre a aplicação e os componentes que interagem numa área uniforme de solo, ao passo que os componentes descrevem as mudanças temporais no solo e nas plantas em resposta ao clima e ao manejo (JONES et al., 2003). Cada módulo possui seis passos operacionais responsáveis pela estruturação e organização, os quais são: inicialização da execução, instalação da cultura, cálculo das taxas, integração, saídas diárias e sumário das saídas.

Dentre os componentes primários encontrados no DSSAT, como clima, solo-planta-atmosfera, CROPGRO (modelo para a cultura), planta, manejo e praga, o módulo solo é representado como um perfil unidimensional, sendo homogêneo horizontalmente e consistindo de um número de camadas verticais de solo. O módulo solo é composto por quatro submódulos: água no solo, temperatura do solo, carbono e nitrogênio do solo.

O submódulo água no solo computa as mudanças diárias no conteúdo de água do solo pela infiltração da chuva e irrigação, drenagem vertical, fluxo insaturado, evaporação e processo de absorção de água pelas raízes. A infiltração de água é computada pela subtração do *runoff* da precipitação que ocorre durante o dia, sendo o processo baseado no número da curva (*curve number*), que leva em conta a textura, declive e cobertura. A drenagem através do perfil é calculada com base num parâmetro assumido para que seja constante com a profundidade. Se a condutividade hidráulica saturada de qualquer camada for menor que a drenagem vertical computada, a água é acumulada na camada acima. Quando há a aplicação de irrigação, a quantidade de água é adicionada à precipitação para cálculo da infiltração e *runoff* diário (FARIA; BOWEN, 2003).

O submódulo solo-planta-atmosfera é responsável por computar diariamente a evaporação do solo e transpiração das plantas; traz junto ao solo as entradas da planta e atmosfera e computa a intercepção da luz pelo dossel, a evapotranspiração potencial e a transpiração da planta.

Com o módulo Praga o usuário pode entrar com suas observações sobre populações de insetos ou danos em diferentes partes da planta, severidade de doenças em diferentes tecidos da plantas e danos físicos

a plantas ou seus componentes. Os efeitos da praga e/ou doença no crescimento e produção são determinados de forma empírica, por meio de uma função de dano.

Os dados mínimos exigidos pelo modelo para sua operação abrangem o local onde o modelo será operado (latitude, longitude, elevação, médias anuais de temperatura e sua amplitude etc.), dados climáticos diários do período do ciclo de crescimento (radiação solar, temperatura do ar máxima e mínima e precipitação), características do solo (características básicas das camadas do solo), condições iniciais (cultura anterior, raiz etc.) e gerenciamento da cultura (tipo e nome do cultivar, data de plantio, profundidade, espaçamento etc.).

Conforme Verhagen et al. (2001), dentro da estrutura do DSSAT podem-se encontrar alguns modelos de simulação para diversas culturas, como os seguintes:

- a família CERES de modelos: CERES-maize, CERES-wheat, (CROPSIM-Wheat), CERES-rice, CERES-barley, CERES-sorghum e CERES-millet;
- a série de modelos CROPGRO para leguminosas: CROPGRO-soybean, CROPGRO-peanut, CROPGRO-dry bean (*Phaseolus*);
- a série de modelos CROPSIM para culturas de raiz: CROPSIM-cassava e SUBSTOR-potato;
- outras culturas: CROPGRO-Tomato, CROPGRO-Chickpea, Sugarcane e Sunflower.

Todos os modelos de culturas compartilham um formato comum de entrada e saída e são similares em nível de detalhe, operando num

passo de tempo diário; são baseados num entendimento de processos biofísicos. Os modelos são orientados a processo, projetados para aplicações globais e trabalham independentemente de local, estação climática, cultivar e sistemas de gerenciamento. Os modelos simulam o efeito do clima, água no solo, genótipo e dinâmica do nitrogênio na cultura e no solo em relação ao crescimento da cultura e produção (VERHAGEN et al., 2001).

Dentre as maiores limitações existentes no DSSAT relacionadas com os modelos de simulação de culturas estão a ausência do efeito imposto às culturas por pragas/doenças, o modo de preparo do solo, o excesso de água no solo, entre outros, deixando um campo aberto para a inserção de outros modelos, mesmo que isso implique gasto de energia para a integração dos diferentes modelos.

CropSim-Wheat

O CropSim-Wheat, substituto do Ceres-Wheat, é um dos modelos de simulação pertencentes à suite do DSSAT, responsável pela simulação do crescimento e desenvolvimento do trigo (Figura 13, Modelos de Crescimento).

No Brasil, o CropSim-Wheat tem sido testado, calibrado e validado por pesquisadores, sendo utilizado para simular o processo do desenvolvimento de cultivares de trigo. Este modelo permite a sua parametrização por meio de arquivos texto. Suas saídas, resultado da simulação, podem ser utilizadas para alimentar outros modelos, responsáveis pela simulação de doenças em plantas, por exemplo (FERNANDES et al., 2004; DELPONTE et al., 2005).

É composto por um módulo planta de trigo que se conecta com o

módulo de clima e de solo, pertencentes à suite do DSSAT, os quais computam a energia e a água disponível para o crescimento da planta de trigo, ao passo que o módulo planta de trigo simula os eventos fenológicos, expansão foliar, acúmulo de carboidratos e partição entre a parte aérea e as raízes. A alimentação do modelo consiste em características da planta do trigo (cultivares), dados de clima (radiação solar, temperatura mínima, temperatura máxima e precipitação), assim como práticas agronômicas, como data de semeadura e espaçamento (HUNT; PARARAJASINGHAM, 1995).

O modelo

Para o desenvolvimento deste modelo genérico, foram utilizadas as mais variadas e heterogêneas tecnologias. Para o desenvolvimento dos módulos do simulador, foi usada a linguagem de programação Java (CORNELL; HORSTMANN, 1997; SRIDHARAN et al., 1997), desenvolvida pela Sun Microsystems, completamente orientada a objetos, e o modelo do crescimento e desenvolvimento da cultura foi escrito em Fortran (ACOCK; TRENT, 1991; HUNT; PARARAJASINGHAM, 1995; JONES et al., 2001, 2003), uma linguagem de programação estruturada.

Para realizar a conexão entre as duas linguagens (Java e Fortran), foi usada a linguagem R (R Development Core Team, 2006), a qual permite a integração de diferentes linguagens de programação, além de oferecer ótimos recursos computacionais para operações estatísticas e geração de gráficos bem elaborados (RIBEIRO JUNIOR, 2001; SILVA JUNIOR, 2005). Outra ferramenta que auxilia na integração é o Rserve, um servidor TCP/IP desenvolvido por Simon Urbanek sob licença GPL (URBANEK, 2003; RSERVE, 2007).

Para o armazenamento das informações foi usado o Sistema Gerenciador de Banco de Dados PostgreSQL 8.1 (DOUGLAS; DOUGLAS, 2005), banco de dados relacional (<http://www.postgresql.org/>), hospedado num servidor Linux Ubuntu Dapper (<http://www.ubuntu.com>), onde foi armazenada a base de dados do sistema, tanto dados de entrada como de saída, como registros meteorológicos históricos, corrente e de previsão, coeficientes genéticos de cultivares, características de solo etc.

Além das tecnologias relatadas, utilizou-se também a linguagem XML (EXtensible Markup Language) para armazenar os parâmetros do modelo, da doença a ser simulada e do ambiente (Tabelas 1, 2 e 3). O modelo comentado podem ser visualizado na Figura 14, na qual se pode verificar todas as ligações existentes entre os módulos do sistema.

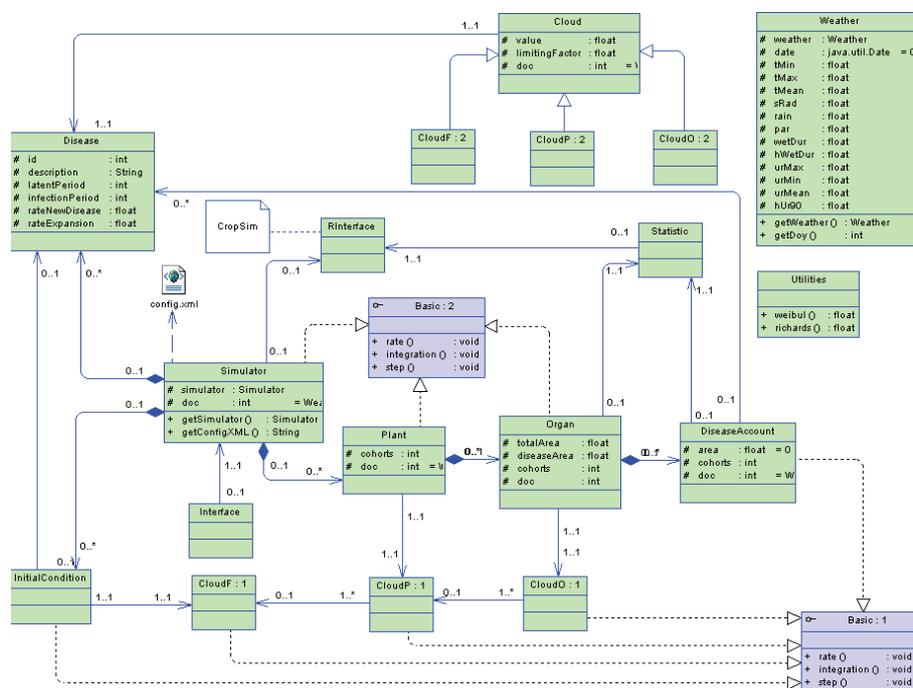


Figura 14: Estrutura do Modelo Genérico de Doenças.

Com uma visão de criar o modelo de forma a ser decomposto em partes coesas, mas fracamente acopladas (modularidade), que permitisse a substituição de diferentes versões dos módulos a qualquer tempo, adotaram-se como padrão as etapas de desenvolvimento descritas por Porter et al. (1999).

Para o desenvolvimento do modelo tomou-se por base o ciclo de vida do patógeno, descrito anteriormente, criando uma estrutura facilmente parametrizável, buscando, dessa forma, obter os resultados esperados. Como os modelos necessitam de integração (modelo de crescimento do trigo e do progresso da doença), todo processamento se dá pela troca de dados sobre o número de folhas criadas e a área disponível de cada uma (área sadia) naquele instante e pela informação sobre a área que a doença está utilizando (área infectada).

O modelo é composto pelo módulos Simulator (Simulador), InitialCondition (Condições Iniciais), Plant (Planta), Organ (Órgão), Disease (Doença), Cloud (nuvem), CloudO (nuvem do órgão), CloudP (nuvem da planta), CloudF (nuvem do campo), LesionCohort (Grupos de lesões), Interface (Interface com o usuário), RInterface (Interface com o R), Basic (Interface implementa pelos demais módulos), Weather (Clima) e Statistic (Estatísticas), desenvolvidos como classes da linguagem Java, os quais descrevem as características do grupo que representam, estando organizados de forma hierárquica a fim de representar, da forma mais próxima possível da realidade, o modelo.

O modelo é de passo diário, alimentando-se de dados climáticos armazenados em bancos de dados, tanto prognósticos como observados ou capturados de estações meteorológicas. Cada componente do modelo é entendido como um objeto ou um grupo de objetos, ou seja, uma entidade virtual que representa as características essenciais da

entidade real.

Módulo Simulator

É o módulo responsável pelo controle do fluxo de toda simulação, podendo ser independente ou controlado pelo módulo **Interface**, o qual pode determinar a velocidade da execução e solicitar ao mesmo os dados daquele instante da simulação, a fim de que possam ser visualizados e analisados (dados de um determinado "dia" da simulação). Possui uma estrutura básica como todos os demais módulos, com "inicialização", "taxa", "integração" e "saída" (PORTER et al., 1999; JONES et al., 2001).

Na inicialização são realizadas: (i) as leituras dos parâmetros do modelo (*config.xml*), como tipos e quantidades de doenças a simular, configurações do processo de simulação, quantidade de plantas a simular, datas de semeadura, localização da base de dados e suas configurações, entre outras; (ii) a criação dos objetos do módulo **Disease**; (iii) a inicialização do módulo **Weather**, o qual realiza a leitura dos dados climáticos e de prognóstico para o período da cultura; (iv) realiza a criação dos objetos do módulo **InitialCondition**, um para cada doença simulada.

A cada passo da simulação são realizadas:

- em taxa, (i) a verificação da existência de semeadura naquele "dia"; havendo, são criados os objetos do módulo **Plant**, conforme os parâmetros do modelo; (ii) a invocação dos métodos de cálculo de taxas para cada objeto dos módulos **InitialCondition** e **Plant**;
- em integração, a invocação dos métodos de integração para cada

objeto dos módulos **InitialCondition** e **Plant**;

- em saída, a invocação dos métodos de saída para cada objeto dos módulos **InitialCondition** e **Plant**.

Módulo RInterface

É o módulo em que ocorre a integração entre o modelo de simulação de doenças de plantas e o modelo de simulação do crescimento e desenvolvimento do trigo, CropSim-Wheat, desenvolvido em Fortran (HUNT; PARARAJASINGHAM, 1995; HUNT, 2007; ICASA, 2007).

Neste módulo, além da linguagem de programação Java (MICROSYSTEMS, 2002; SILVEIRA, 2007; MICROSYSTEMS, 2007), foi utilizada a linguagem R (SILVA JUNIOR, 2005; R Development Core Team, 2006), a qual possibilita a integração entre diferentes linguagens, como R e Fortran. Para possibilitar a integração entre a linguagem Java e a linguagem R utilizou-se o Rserve, um servidor TCP/IP para a linguagem R, o qual suporta conexões remotas, autenticação e transferência de dados e arquivos (Figura 15). O modelo de simulação CropSim-Wheat foi modificado e recompilado¹ para a plataforma Linux, de forma a ter o seu processamento (passo) controlado por este módulo.



Figura 15: Conexão entre o modelo genérico de doença com o CropSim-Wheat.

Ao final de cada passo da execução do modelo de doença, são repassadas ao CropSim-Wheat informações sobre a área doente, estas

¹Utilizando o compilador Fortran Intel 9.0

computadas para a simulação do crescimento e desenvolvimento da planta de trigo; por sua vez, as saídas deste modelo, como a quantidade de órgãos existentes, a área total de cada órgão e a área senescente de cada órgão, são utilizadas pelo modelo de doença para o cálculo de novas lesões, produção de esporos e expansão da lesão.

Módulo Interface

É responsável pelo gerenciamento da simulação e interface com o usuário, como passo da execução, ano da simulação, edição dos parâmetros da simulação, visualização dos resultados em forma de tabelas e gráficos, entre outros. Este módulo pode ser reestruturado de forma a servir aos interesses dos usuários, podendo ser utilizado como um aplicativo Desktop, baseado na Web (Web-based), rodando em browsers, e aplicativos para equipamentos de pequeno porte, como celulares, PDAs (Personal Digital Assistant) etc.

O controle da simulação pode ser feito localmente ou a distância, dependendo da configuração do sistema e do tipo de plataforma utilizada (Desktop, Web ou pequeno porte), pois a interface pode estar, por exemplo, num dispositivo móvel, e o modelo, num local remoto, como um servidor de aplicação remoto, conforme pode ser visto na Figura 16.



Figura 16: Controle da simulação pela interface com o usuário.

Módulo Weather

O módulo Weather é o responsável por alimentar os sistemas de simulação com dados climáticos, oriundos tanto de estações meteorológicas como de prognósticos. É uma classe que possui métodos (funções) estáticos (disponíveis a qualquer módulo), os quais disponibilizam desde informações como temperatura e precipitação até formas de controle de navegação (passar para o próximo dia, por exemplo).

Os dados climáticos podem ser oriundos de bases de dados relacionais, arquivos texto ou arquivos em formato XML (eXtended Markup Language), cujas configurações são realizadas no arquivo *config.xml*, carregados pelo módulo **Simulator** (Figura 17).

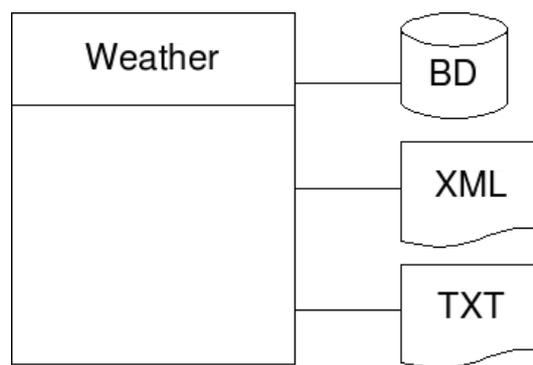


Figura 17: Interface Weather e suas fontes de dados.

Módulo Utilities

Neste módulo, como no **Weather**, estão contidos os métodos estáticos responsáveis pelo cálculo de diversos tipos equações, podendo-se citar: (i) *trapezoidalFunction*, utilizada, por exemplo, para o cálculo de dias fisiológico (ROSSI et al., 1997); (ii) *temperatureFavorability*, responsável pelo cálculo da favorabilidade do efeito da temperatura

(PIVONIA; YANG, 2006), por exemplo, no cálculo da maturação fisiológica dos esporos de uma doença (condição inicial) ou na eficiência de uma determinada infecção; (iii) *wetnessFavorability*, responsável pelo cálculo da favorabilidade do molhamento em eventos biológicos (VALE; ZAMBOLIM, 1996), por exemplo, como coadjuvante na determinação da maturação fisiológica dos esporos de uma doença; (iv) além de outros métodos auxiliares para o processo de simulação.

Módulo InitialCondition

A quantidade de inóculos primários de um patógeno é um importante fator de risco numa cultura (YOUNG et al., 2003), visto que o inóculo gerado nos restos culturais e em plantas voluntárias é assumido como a fonte predominante de infecção (AUDSLEY et al., 2005). Portanto, este módulo representa as condições iniciais de inóculo, determinando o momento em que há a liberação dos primeiros esporos do patógeno ao ambiente. O número de esporos liberados para o ambiente (nuvem de esporos) é determinado por meio do arquivo de configuração (*config.xml*), sendo o momento da liberação determinado pelo cálculo do acúmulo da favorabilidade, também determinado no arquivo de configuração para cada doença, utilizando, para tal, a temperatura média e a duração do molhamento. Esta determinação é feita no método “taxa” a cada passo da simulação, por meio da equação:

$$AF = \sum_{i=1}^t TF * WF$$

onde *AF* é acúmulo de favorabilidade; *TF*, a favorabilidade em relação à temperatura (*Utilities.temperatureFavorability*); *WF*, a favorabilidade em relação ao molhamento (*Utilities.wetnessFavorability*) e *t* é o dia.

Módulos Cloud?

Os módulos **CloudF**, **CloudP** e **CloudO** são responsáveis por representarem a quantidade de esporos disponíveis para novas infecções e foram divididos a fim de se estimar a concentração de esporos e, conseqüentemente, de novas infecções em três escalas organizacionais: campo (F), planta (P) e órgão (O). Essa forma de organização tem como objetivo estruturar a epidemia numa hierarquia espacial de deposição de esporos: autodeposição, deposição de um propágulo produzido numa lesão localizada na própria folha e allo-deposição, deposição de um propágulo produzido numa lesão localizada em outra folha da mesma planta ou em outra planta (Figura 18). Permite, dessa forma, a incorporação de parâmetros epidemiológicos para a propagação da doença em diferentes escalas, assim como as taxas de autodeposição e allo-deposição dos propágulos (WILLOCQUET; SAVARY, 2004).

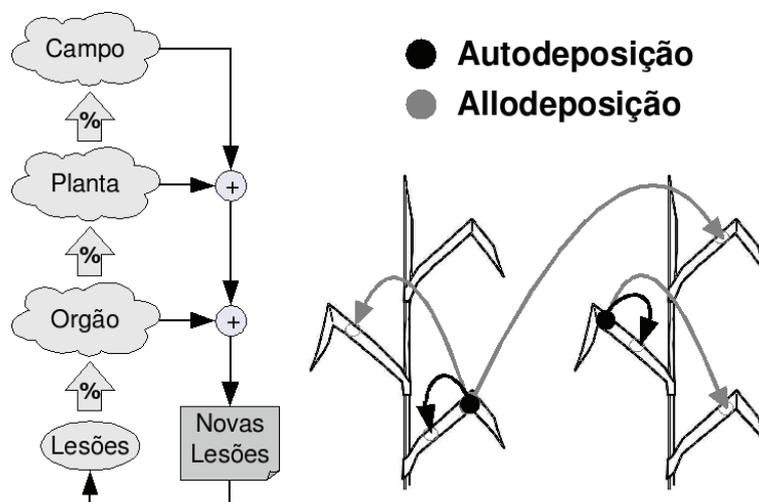


Figura 18: Auto e allo deposição (Adaptado de Willocquet e Savary (2004))

A contribuição inicial para a nuvem do campo se dá pelo módulo

InitialCondition, quando da liberação dos primeiros esporos, não sendo considerados propágulos vindos de fora. O valor inicial e mínimo da nuvem é determinado pelas configurações feitas e carregadas pelo simulador. A nuvem do campo (**CloudF**) não contribui para as demais, sendo seu valor apenas utilizado para novas infecções.

A cada passo da execução do modelo, novos esporos, criados por grupos de lesões num determinado órgão (módulo **LesionCohort**), podem ser adicionados à sua respectiva nuvem. Nos parâmetros epidemiológicos definidos para cada doença (*config.xml*), a taxa de contribuição da nuvem do órgão para a nuvem da planta e desta para a do campo é definida (Figura 19). Isso se dá em virtude da hipótese de que a infecção num órgão já infectado é maior do que num órgão sadio.

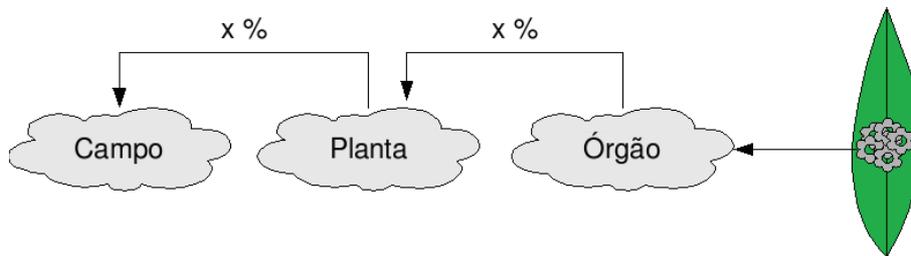


Figura 19: Alimentação e contribuição entre as nuvens

A eliminação de esporos da nuvem se dá de três formas: (i) quando do processo de infecção, subtraindo o número de novas infecções; (ii) precipitados em razão da ocorrência de chuvas (BERGER et al., 1995; AUDSLEY et al., 2005); (iii) quando a idade dos esporos ultrapassar o limite definido nos parâmetros de configuração (SACHE, 1997). A subtração pelas condições climáticas ocorre quando a precipitação (chuva) for superior a 30mm, diminuindo o seu valor em 50%.

Módulo Disease

Este módulo é composto por diversos objetos, sendo um para cada tipo de doença, criados pelo módulo **Simulator**, conforme os parâmetros de configuração. Serve como base de informação sobre as características da doença, possuindo métodos de acesso a essas características. Além dos métodos de acesso aos dados, há um método para o cálculo de novas lesões num determinado instante, sendo necessária a existência de área sadia e condições de molhamento, este superior ou igual à configuração para a corrente doença (*wetnessThreshold*) (MAGAREY et al., 2005). A determinação do número de novas lesões é dada pela equação:

$$NL = CD * HAP * IE * TF$$

onde *NL* (*number of lesions*) é o número de novas lesões; *CD* (*cloud density*), a densidade da nuvem; *HAP* (*healthy area proportion*), proporção da área sadia; *IE* (*infection efficiency*), eficiência de infecção, e *TF* (*temperature favorability*) é favorabilidade da temperatura.

Módulos Statistic e BasicInterface

O módulo **Statistic** está intimamente relacionado ao módulo **RInterface**, sendo responsável por disponibilizar aos demais módulos, por meio dos seus métodos, dados estatísticos como distribuições, assim como a execução de funções estatísticas especiais.

Para forçar que todos os módulos do sistema tivessem a mesma estrutura, evitando problemas de comunicação entre eles, desenvolveu-se uma interface básica (**BasicInterface**), a qual foi implementada por todos. Esta é uma interface Java, que força a que os módulos implementem os métodos *rate*, *integration* e *output*, deixando o sistema

mais robusto e modular.

Módulo **Plant**

O módulo **Plant** é responsável por representar a planta (neste caso, o trigo), com seus órgãos e suas respectivas áreas (sadias e infectadas); possui uma ligação forte com o módulo **RInterface**, em virtude da busca, a cada passo da simulação, de dados sobre o desenvolvimento da planta, atualizando seus respectivos dados, além de alimentar o simulador CropSim-Wheat com dados sobre áreas infectadas.

Objetos da classe **Plant** são criados pelo módulo **Simulator**, conforme a parametrização do sistema, a cada data de semeadura. Para cada objeto criado, no momento da sua inicialização é realizada a criação de objetos do módulo **CloudP**, sendo um para cada tipo de doença; portanto, cada objeto planta possui e alimenta uma nuvem de esporos, conforme a área infectada.

Da mesma forma que os demais módulos, o módulo **Plant** possui uma estrutura padrão; em relação ao método “taxas”, possui uma implementação simples, responsável apenas por invocar o mesmo método para cada um dos seus órgãos. Por sua vez, o método “integração”, quando invocado, é responsável por: (i) invocar o método “integração” de cada órgão ligado a ele; (ii) acumular a área total, a área doente total, a área doente em estágio de latência, a área doente em estágio infeccioso e a área necrótica; (iii) invocar os métodos “integração” de cada uma das nuvens ligadas a ele; e (iv) verificar, através do módulo **RInterface**, a existência de novos órgãos, realizando a criação destes, se necessário. Além dos métodos padrões, oferece um método especial que informa se a planta está “viva”, ou seja, não senescente, realizando

esta tarefa por meio de consultas a cada um dos seus órgãos.

Módulo Organ

É responsável por representar os órgãos de uma planta, computar o avanço da epidemia em seus três estádios (latente, infeccioso e necrótico), computar sua área sadia, área infectada, área necrótica e senescente. No momento da inicialização, cria objetos do módulo **CloudO**, um para cada tipo de doença, além de capturar do módulo **RInterface**, a sua área atual.

No instante em que o seu método “taxas” é invocado pelo objeto do módulo **Plant**, é realizada, em primeira instância, a atualização da área total e senescente, por meio de consulta ao módulo **RInterface**. Para cada grupo de lesões existentes no órgão é informada a proporção de área doente e, em seguida, é invocado o método ”taxas” de cada grupo.

Uma das atividades mais importantes do método “taxas” é a criação de um novo grupo de lesões e, para a determinação do número de lesões deste grupo, faz-se necessário obter a densidade da nuvem, utilizada na invocação do método “novas lesões” do módulo **Disease**. Conforme o resultado obtido deste método, um novo grupo é criado, o qual é anexado à lista dos grupos de lesões. No momento da criação do grupo, é informada a quantidade de lesões que este representa, assim como a nuvem do órgão sobre a qual este grupo deverá realizar a alimentação de novos esporos da doença.

Na execução do método “integração”, para cada nuvem e grupo de lesões é realizada a chamada de seus respectivos métodos de “integração”, seguida pelo cálculo de sua área total, da área em estádio

de latência, da área em estágio infeccioso e da área necrótica. Finalmente, é realizada a atualização da área senescente deste órgão, obtida do módulo **RInterface**.

Módulo **LesionCohort**

Pode-se considerar este módulo como um dos mais importantes, responsável pelo processo epidemiológico, contribuindo ou não para a propagação da doença. Como os demais, é composto pelos métodos “inicialização”, “taxa”, “integração” e “saída”.

No momento da inicialização, armazena o número de lesões que representa e realiza o cálculo da área inicial. Este cálculo é o resultado do produto do número de lesões pelo tamanho inicial de cada lesão (parâmetros carregados pelo módulo **Simulator**). Nas execuções do método “taxa” realiza-se o controle do dia fisiológico, sendo importante para indicar o estágio em que suas lesões se encontram (latente, infeccioso ou necrótico); se estiver no estágio infeccioso, cria novos esporos, conforme a sua idade fisiológica, pois possui diferentes taxas de produção de novos esporos. Para a determinação do número de esporos criados naquele passo do modelo é utilizada a equação:

$$NS = C * MSP * SPC * TF * SCF$$

onde NS é o número de esporos a serem criados; C o número de lesões que o grupo representa; MSP , a máxima produção de esporos para a doença corrente; SPC , a proporção de esporos que podem ser adicionados na nuvem; TF , o resultado da execução do método *Utilities.trapezoidalFunction*, a qual determina o poder de criação de novos esporos em relação à idade do grupo de lesões, e SCF é o fator de esporulação em virtude da proporção da área ocupada pela doença

(SACHE, 1997; VALLAVIEILLE-POPEA et al., 2000). O *SCF* é a resposta da execução do método *getSporulationCrowdingFactor*, contido no módulo **Disease**, que recebe como parâmetro a proporção da área doente do órgão.

No método “integração” é realizada a atualização da área ocupada pelo grupo de lesões, a qual se dá pela equação:

$$A_t = A_{t-1} + A_{t-1} * IF * RE$$

onde A_t é a área ocupada pelo conjunto de lesões; A_{t-1} , a área ocupada pelo conjunto de lesões no dia anterior; IF , o fator de interferência do crescimento para aquele tipo de lesão e órgão, e RE é a taxa de expansão da doença corrente. Além do cálculo da área ocupada pelo conjunto de lesões, a informação sobre o estágio do grupo é atualizada conforme a comparação do valor do período de cada estágio com valor do dia fisiológico atual.

No método “saída” (*output*), como em todos os demais módulos, é realizado o armazenamento dos resultados numa matriz, a qual pode ser utilizada para geração das saídas para o usuário, por meio do módulo **Interface**.

Resultados

O modelo apresentado foi desenvolvido com o objetivo de abranger um maior número possível de doenças (genérico), de forma que estas pudessem ser parametrizadas a fim de representar o mais realisticamente possível as observações feitas em campo.

A estrutura de classes construída com o auxílio da linguagem UML facilitou a visualização e interligação dos componentes de um

modelo genérico de simulação de epidemias. A utilização do Java, uma linguagem puramente orientada a objetos, para implementar o simulador permitiu que o sistema computacional ficasse bem próximo do sistema real, e a parametrização do simulador em arquivo de configuração XML complementou a modularidade e a versatilidade do simulador genérico de epidemias.

A possibilidade de ligar o simulador de epidemia com um modelo de crescimento de plantas expandiu o uso e aplicabilidade dos mesmos. Essa ligação, uma "via de mão dupla", permitiu que, além de receber dados do modelo de crescimento, o que é mais comum, o modelo de epidemias retroalimentasse, em tempo de execução, o modelo de crescimento, afetando o crescimento da planta de trigo por meio de informações sobre a área doente, cujo resultado afeta a doença, formando-se, dessa forma, um ciclo dinâmico.

O patossistema *Puccinia*-Trigo, escolhido para testar a infraestrutura proposta, foi modelado segundo as classes estabelecidas na Figura 14. Na implementação do modelo, seguindo a estrutura definida por Porter et al. (1999), na qual um sistema é decomposto de forma a torná-lo modular, foram criadas as seções de inicialização, taxas, integração e saída para o controle das variáveis de estado. A utilização desta estrutura facilitou o controle da lógica da execução, assim como a ligação entre os modelos de simulação. O simulador de epidemias de ferrugem da folha do trigo também foi implementado na linguagem Java, utilizando-se das características inerentes desta linguagem e do paradigma de orientação a objetos. Um fluxograma sumarizado do simulador implementado é apresentado na Figura 20.

Foi desenvolvida uma interface para facilitar a entrada de dados iniciais e apresentar os resultados dos modelos de simulação na forma

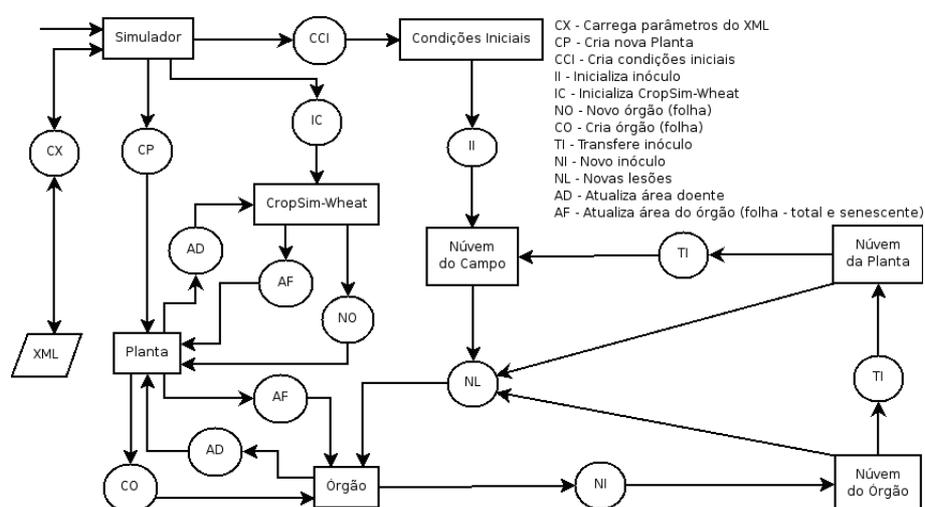


Figura 20: Fluxograma de funcionamento do modelo genérico

tabular e gráfica. No seu desenvolvimento foram utilizados, novamente, os recursos da linguagem Java, da linguagem R e de pacotes de software diversos, permitindo uma interação plena do usuário com o sistema, além de possibilitar um controle efetivo do fluxo de execução. Uma tela capturada do sistema pode ser visualizada na Figura 21.

Uma descrição dos parâmetros utilizados no simulador de epidemia de ferrugem da folha é apresentada nas Tabelas 1 e 2. Os valores dos parâmetros foram retirados da literatura (YANG et al., 1991; SACHE, 1997; PIVONIA; YANG, 2006) ou obtidos por meio de estimativas aproximadas e com algum sentido biológico.

O simulador de epidemias de ferrugem da folha do trigo inicia com a leitura dos dados contidos no arquivo de configuração XML (*config.xml*), tanto para a inicialização do simulador (Tabela 3) como da doença em si (Tabelas 1 e 2). Com os parâmetros carregados, o simulador inicia a classe Weather, responsável por fornecer os dados meteorológicos e prognósticos climáticos, e a classe de RInterface, responsável pela integração com o simulador de culturas CropSim-

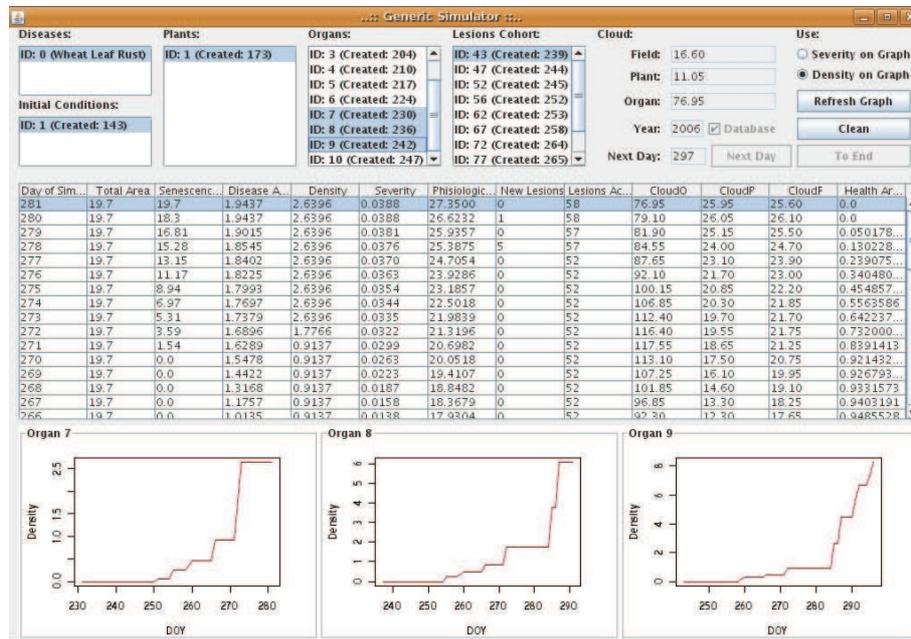


Figura 21: Tela da Interface visual do simulador genérico de doenças Wheat.

Entre os diversos parâmetros do simulador encontra-se a data de início da simulação, a qual foi ajustada para trinta dias antes da data de semeadura, com o intuito de realizar o cálculo do fator de favorabilidade e criação do inóculo primário, realizado (Figura 22) para cada doença especificada no arquivo de configuração.

```

64 public void rate() {
65     if(!stop) {
66         acumulateFavorability += Utilities.temperatureFavorability(getWeather().getTMean(),
67             cloud.getDisease().getInfectionTemperatureFavorabilitySet()*
68             Utilities.wetnessFavorability(getWeather().getWetDur());
69     }
70     cloud.rate();
71 }

```

Figura 22: Acúmulo de favorabilidade para liberação do inóculo inicial

O cálculo da favorabilidade para liberação do inóculo inicial foi realizado fazendo-se o somatório do produto da favorabilidade da temperatura (Figura 23A) pela favorabilidade do molhamento foliar (Fi-

Tabela 1: Parâmetros utilizados no simulador da ferrugem da folha do trigo (parte I)

Nome da variável	Descrição	Unidade	Exemplo
description	Descrição da doença parametrizada	descrição	Rust
infectionEfficiency	Eficiência da Infecção	proporção	0.17
environmentFavorability	Interferencia do ambiente no desenvolvimento da doença	proporção	1
depositionFrequency	Frequência da deposição de esporos	proporção	0.05
initialInoculum	Valor do inóculo inicial	unidades	10
vectorSizeCloudF	Limite de permanencia de esporos na nuvem do campo	unidades	5
vectorSizeCloudP	Limite de permanencia de esporos na nuvemda planta	unidades	7
vectorSizeCloudO	Limite de permanencia de esporos na nuvemdo órgão	unidades	10
maxSporeProduction	Produção máxima de esporos	unidades/dia	3000
cohortAge	Eficiência na produção de esporos (trapezóide) com relação à idade do <i>cohort</i>	dias	7, 15, 18, 37
scfA, scfB e scfC	Limitante da esporulação baseada na densidade da população	parâmetros	0.98669, 10.71894, 0.93374
maxSporeDensity	Densidade máxima de esporos na nuvem	unidades	12000
sporeProportionToCloud	Proporção de esporos produzidos que alimentam a nuvem do órgão	proporção	0.10
proportionFromOrganToPlantCloud	Proporção de esporos transferidos da nuvem do órgão para a nuvem da planta	proporção	0.04
proportionFromPlantToFieldCloud	Proporção de esporos transferidos da nuvem da planta para a nuvem do campo	proporção	0.02
infectionTemperatureFavorabilitySet	Temperaturas cardinais utilizadas para cálculo da favorabilidade da temperatura, no período infeccioso (máxima, mínima e ótima)	temperaturas	30, 2, 15

Tabela 2: Parâmetros utilizados no simulador da ferrugem da folha do trigo (parte II)

Nome da variável	Descrição	Unidade	Exemplo
latencyTemperatureFavorabilitySet	Temperaturas cardinais utilizadas para cálculo da favorabilidade da temperatura, no período latente (máxima, mínima e ótima)	temperaturas	40, 0, 27
latentPeriod	Comprimento do período latente	dias	6
latentGrowthFunction	Função utilizada para o cálculo da expansão da lesão no período latente (lesão visível)	função	gompertz
latentGrowthParameter	Parâmetros da função utilizada no cálculo da expansão da lesão no período latente (lesão visível)	parâmetros	0.0161858, 1.563509, 0.441721
infectionPeriod	Comprimento do período infeccioso	dias	20
infectionGrowthFunction	Função utilizada para o cálculo da expansão da lesão no período infeccioso (lesão visível)	função	exponential
infectionGrowthParameter	Parâmetros da função utilizada no cálculo da expansão da lesão no período infeccioso (lesão visível)	parâmetros	0.01507, 1.0104, 0.2125
initialPustuleSize	Tamanho inicial da pústula	cm ²	0.00001
maxPustuleSize	Tamanho máximo da pústula (vetor: latente e infeccioso) - lesão visível	cm ²	0.012F, 0.015F
invisibleGrowthFunction	Função utilizada para o cálculo da expansão da lesão (lesão invisível - área total)	função	exponential
invisibleGrowthParameter	Parâmetros da função utilizada no cálculo da expansão da lesão (lesão invisível - área total)	parâmetros	0.036, 1.0104, 0.3125
wetnessThreshold	Número mínimo de horas de molhamento necessário para criar novas lesões	horas	6
accumulateFavorability	Número de dias fisiológicos necessários para liberação dos primeiros esporos	dias fisiológicos	25
hostFactor	Fator de resistência do hospedeiro (1: sem resistencia)	proporção	1

Tabela 3: Parâmetros de configuração da planta e do simulador

Nome da variável	Descrição	Unidade	Exemplo
maxOrganNumber	Número máximo de órgãos por planta	unidades	10
dss	Dia de início da simulação (Calendário Juliano)	dia	143
maxOrganPhysiologicalLife	Tempo máximo de vida de um órgão da planta	dias fisiológicos	20
cardinalTempPhysiologicalLife	Temperaturas cardinais para cálculo dos dias fisiológicos de um órgão da planta	temperaturas	0,28,30,40
sowingDates	Datas de semeadura	datas (Juliano)	#173#180#
conection	URL de conexão para o banco de dados	URL	jdbc:postgresql://host/base
driver	Nome do driver utilizado para acesso ao banco de dados	descrição	org.banco.Driver
user	Nome do usuário no banco de dados	descrição	sisalert
password	Senha do usuários no banco de dados	descrição	senha

gura 23B), visto que ambas as funções respondem aos parâmetros definidos no arquivo de configuração. No caso da ferrugem da folha utilizaram-se os valores apresentados nas Tabelas 1 e 2.

```

public static float temperatureFavorability(float temp, float[] cardinalTemperatures) {
    float tempMax=cardinalTemperatures[0], tempMin=cardinalTemperatures[1],
        tempOpt=cardinalTemperatures[2];
    float tf, a, b;
    b = (tempMax-tempOpt)/(tempOpt-tempMin);
    a = 1 / ((tempOpt-tempMin) * (float) Math.pow(tempMax-tempOpt,b));
    tf = a * (temp - tempMin) * (float) Math.pow(tempMax - temp,b);
    return tf;
}

public static float wetnessFavorability(float wetDuration) {
    float wf, k = 0.99374F, b = 350.81064F, r = 0.88591F;
    wf = k/(1 + b * (float) Math.exp(-r * wetDuration));
    return wf;
}

```

A
B

Figura 23: Funções para cálculo da favorabilidade da temperatura e molhamento

Uma vez acumulado o valor definido para a liberação do inó-

culo inicial (*accumulateFavorability* - Tabela 1), o qual foi empiricamente colocado como sendo dez unidades infectivas (uredeósporos) (*initialInoculum* - Tabela 1), iniciou-se o processo de infecção com a criação de novas lesões quando da existência de tecido sadio nos órgãos (folhas) do hospedeiro.

A cada rodada do simulador da ferrugem do trigo (taxa, integração e saída), as informações sobre o número de órgãos e suas respectivas áreas foram atualizadas por meio de consultas ao simulador da planta de trigo (CropSim-Wheat); com essas informações foi possível determinar o processo de infecção diário. Para o cálculo de novas lesões, levou-se em consideração a proporção de área sadia (*área total* - *área doente* - *área senescente*), a densidade da nuvem de esporos, a eficiência de infecção (*infectionEfficiency* - Tabela 1), o número de horas de molhamento (*wetnessThreshold* - Tabela 1) e a favorabilidade da temperatura naquele instante (Figura 24).

```

/** Return the amount of new lesions based in cloud value and healthy area */
public int newLesions(float cloudDensity, float healthyAreaProportion, int organNumber) {
    int newLesions = 0;
    float fitWetnessThreshold = getWetnessThreshold();
    if(getWeather().getRain() == 0)
        fitWetnessThreshold += (float) 4*organNumber/getConfig().getMaxOrganNumber();
    if(healthyAreaProportion > 0 && getWeather().getWetDur() >= fitWetnessThreshold) {
        newLesions = (int) (cloudDensity*healthyAreaProportion*getInfectionEfficiency()*
            Utilities.temperatureFavorability(getWeather().getTMean(),
                getInfectionTemperatureFavorabilitySet()));
    }
    return newLesions;
}

```

Figura 24: Cálculo de novas lesões

A(s) lesão(ões) formada(s) em cada dia constituiu um grupo (*cohort*) que entrou em período de latência. O período de latência foi determinado por meio das configurações feitas no arquivo XML (*latentPeriod* - Tabela 1), sendo, para este estudo, o valor de 6 (seis) dias fisiológicos (...REF...). A determinação dos dias fisiológicos deu-se pela utilização do método *temperatureFavorability* (Figura 23A), passando a este a temperatura média e o conjunto de temperaturas

cardinais definidas para o período latente (*latencyTemperatureFavorabilitySet* - Tabela 1).

A(s) lesão(ões), agrupadas em *cohorts*, tiveram a expansão de suas áreas, visíveis e invisíveis (Figura 25), de acordo com a função e seus parâmetros especificados (*latentGrowthFunction*, *latentGrowthParameter*), para o valor do dia fisiológico corrente e a resistência do hospedeiro à doença (*hostFactor*). Os valores utilizados podem ser visualizados na Tabela 1.

```
tempVisibleArea = Utilities.growsFunction(disease.getLatentGrowsFunction(),
                                         disease.getLatentGrowsParameter(),
                                         getphysiologicalDay() *
                                         disease.getHostFactor() * cohorts;
tempInvisibleArea = Utilities.growsFunction(disease.getInvisibleGrowsFunction(),
                                           disease.getInvisibleGrowsParameter(),
                                           getphysiologicalDay() *
                                           disease.getHostFactor() * cohorts;
```

Figura 25: Cálculo da expansão das lesões (área visível e invisível) no período latente

Após o período de latência (seis dias fisiológicos), o *cohort* (conjunto) passa para o período infeccioso (vinte dias fisiológicos), tornando visível a sua massa esporulenta e tendo a expansão de suas áreas (Figura 26) governada pelas funções e parâmetros previamente configurados (Tabela 1).

```
tempVisibleArea = Utilities.growsFunction(disease.getInfectionGrowsFunction(),
                                         disease.getInfectionGrowsParameter(),
                                         getphysiologicalDay() *
                                         disease.getHostFactor() * cohorts;
tempInvisibleArea = Utilities.growsFunction(disease.getInvisibleGrowsFunction(),
                                           disease.getInvisibleGrowsParameter(),
                                           getphysiologicalDay() *
                                           disease.getHostFactor() * cohorts;
```

Figura 26: Cálculo da expansão das lesões (área visível e invisível) no período infeccioso

A partir da fase infecciosa, a quantidade de inóculo passou a ser modificada numa estrutura hierárquica de três escalas, ou seja, na folha, na planta e no campo, respectivamente (WILLOCQUET et

al., 2002). A produção de esporos (MEHTA; ZADOKS, 1970) pelos *cohorts* de lesões, quando da existência de área sadia (Figura 27), foi realizada levando-se em consideração o número máximo da produção de esporos diários por pústula (*maxSporeProduction* - Tabela 1), o fator de interferência baseado na densidade populacional da ferrugem (método *getSporulationCrowdingFactor* - Figura 28)(PIVONIA; YANG, 2006), a proporção de esporos que poderia ser repassada para a nuvem do órgão (*sporeProportionToCloud* - Tabela 1) e o fator idade do *cohort* (SACHE, 1997), este resultante da aplicação do método *trapezoidalFunction* (Figuras 27 e 29).

```

if(getProportionOrganHealthArea() > 0.01 && isInfectionPeriod()) {
    newSpores = (int) (cohorts * disease.getMaxSporeProduction() * disease.getSporeProportionToCloud() *
        Utilities.trapezoidalFunction(getWeather().getDay() - doc, disease.getCohortAge()) *
        disease.getSporulationCrowdingFactor(getProportionOrganDiseaseArea()));
}

```

Figura 27: Produção de novos esporos pelo *cohort*

```

public float getSporulationCrowdingFactor(float proportionDiseaseArea) {
    return Math.min(1/(scfA + scfB * (float) Math.pow(proportionDiseaseArea,scfC)),1);
}

```

Figura 28: Cálculo do fator de interferência com relação a área doente

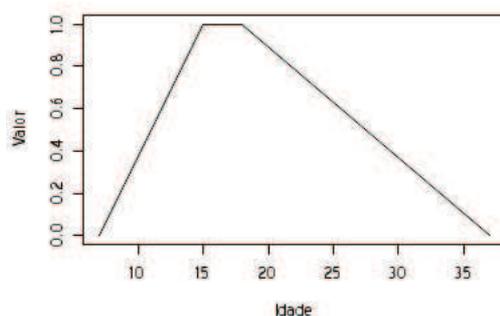


Figura 29: Interferência da idade do *cohort* na produção de esporos

Os esporos criados por cada um dos *cohorts* são transferidos para nuvem do órgão (folha); uma certa proporção deles vai para a nuvem

da planta e outra parte, para a nuvem do campo. Modificações nesses parâmetros provocaram grandes mudanças no resultado da simulação, o que pode ser visto na Figura 30. A obtenção desses parâmetros ideais (proporções) deu-se por tentativa e erro até se alcançar a aproximação de valores observados da doença no campo. Observou-se que pequenas modificações no parâmetro que governa a proporção de inóculo na folha resultam em grandes alterações no resultado final. Na Figura 31 é apresentado um conjunto de três gráficos, cada um representando uma folha de trigo com a melhor combinação de parâmetros encontrados; nos gráficos os valores de densidade de esporos *Puccinia triticina* por unidade (cm^2) de tecido são comparados com valores observados² (dados gentilmente cedidos por Paulo Pires).

A eliminação dos esporos das nuvens se dá de quatro formas: (i) pela chuva, precipitando-os e diminuindo-os em 50%; (ii) pelo tempo em que os esporos estão na nuvem, de modo que o grupo que atingir o tempo máximo de permanência na nuvem é eliminado (*vectorSizeCloud?* - Tabela 1); (iii) pelo processo de infecção, pelo qual os esporos são subtraídos da nuvem; (iv) quando a densidade de esporos ultrapassar a densidade máxima estabelecida (*maxSporeDensity* - Tabela 1), sendo, neste caso, ajustada a quantidade destes para o limite estabelecido.

Todo o ciclo da simulação se repete até que a planta atinja sua maturação fisiológica, sendo esta informação repassada pelo simulador da planta de trigo (CropSim-Wheat) ao simulador da ferrugem do trigo.

Utilizando-se o simulador desenvolvido, várias rodadas foram realizadas para os anos de 1979 a 2006, com o intuito de verificar o

²Informação pessoal.

comportamento do mesmo e analisar os resultados de severidade com referência às informações obtidas em campo (PICININI et al., 1996). Por meio da Figura 32 podem-se verificar os resultados obtidos com este experimento, demonstrando que o simulador acertou com relação aos anos epidêmicos e pouco epidêmicos.

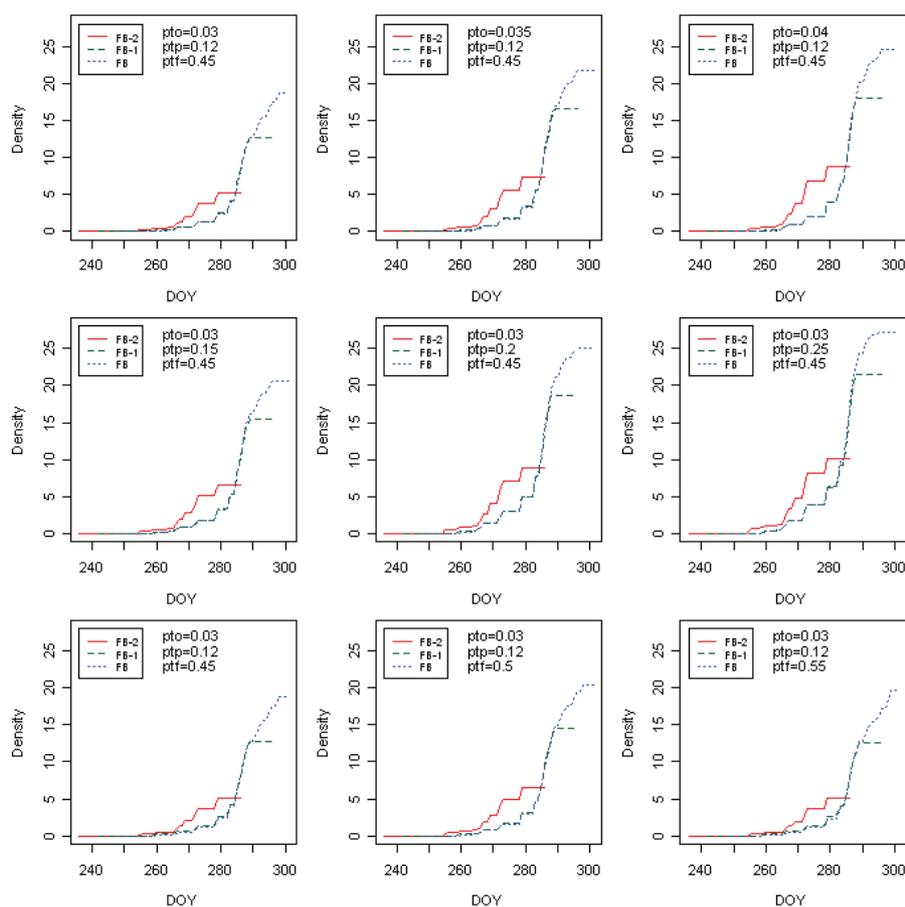


Figura 30: Resultado da modificação dos parâmetros de transferência de inóculo para/entre as nuvens, onde "FB" significa folha-bandeira; "pto", a proporção de de esporos transferidos para a nuvem da folha (Organ); "ptp", a proporção de esporos transferidos para a nuvem da planta (Plant) e "ptf" a proporção de esporos transferidos para a nuvem do campo (*field*).

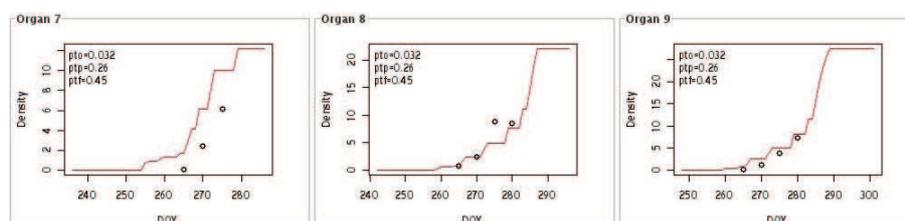


Figura 31: Comparação entre valores observados e simulados para três folhas de trigo (*Organ 9*: Folha-bandeira, *Organ 8*: Folha-bandeira -1 e *Organ 7*: Folha-bandeira -2)

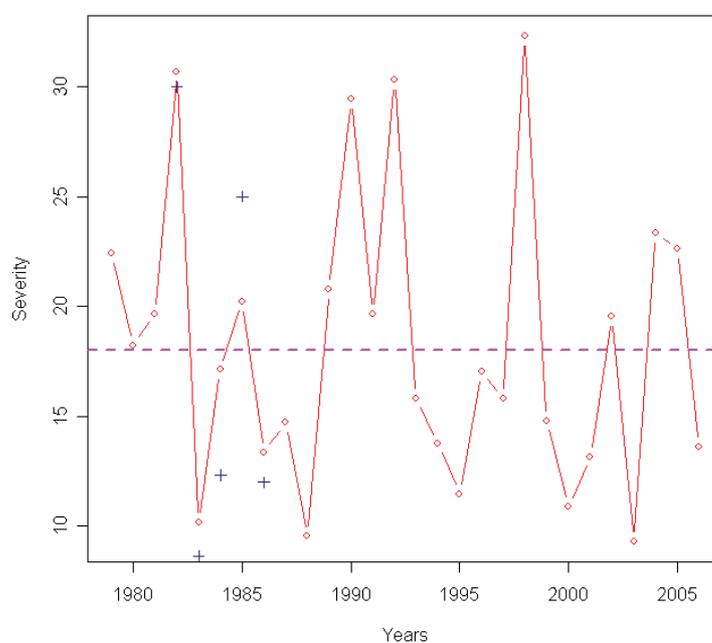


Figura 32: Severidade simulada para os anos de 1979 a 2006

O modelo foi avaliado por meio do comportamento do sistema, da dinâmica das variáveis de estado e das taxas de mudança destas variáveis. O formato da sigmóide da curva da progressão da ferrugem da folha do trigo (Figura 31) confirma o resultado esperado do modelo, constatando-se que mudanças nos parâmetros levaram a mudanças no

curso da epidemia.

Discussão

A utilização de técnicas como a UML e a orientação a objetos permitiu que, durante o desenvolvimento do modelo de simulação da ferrugem da folha do trigo, uma maior atenção fosse dada aos processos biológicos do que à implementação do modelo. Dessa forma, fitopatologistas, assim como especialistas de outras áreas, podem, ao examinar o diagrama de classes, visualizar e entender o relacionamento entre os componentes do modelo, permitindo, assim, que um processo colaborativo possa ser construído.

A engenharia de software, especialmente a UML e as linguagens orientadas a objetos, vem sendo largamente utilizada na área industrial. Até recentemente, pouco se utilizavam essas técnicas na área agrícola, sendo comum a utilização de métodos tradicionais no desenvolvimento de modelos para esta área. Entretanto, o seu uso possibilita o aumento da produtividade, flexibilidade, reúso e qualidade dos softwares produzidos (PAPAJORGI; PARDALOS, 2006). Esses autores apresentam em seu livro vários exemplos e vantagens dessas técnicas para o desenvolvimento de modelos de simulação para a agricultura, mostrando como resolver e implementar, com o uso de UML e orientação a objetos, muitos dos problemas normalmente encontrados.

O modelo de simulação do patossistema *Puccinia*-Trigo simula corretamente o crescimento da área foliar do trigo e o progresso da ferrugem da folha, apresentando-se como uma excelente alternativa para o estudo das relações entre o patógeno e o hospedeiro. Nesse

sentido, destacam-se a quantificação e modelagem do dano causado pelas doenças nas plantas. Modelos de simulação como o proposto são de grande valia para o entendimento dos processos em funcionamento, como o impacto das doenças no rendimento. Infelizmente, ainda são poucos os exemplos que utilizam modelos de simulação para a avaliação de danos (JOHNSON, 1992; ROSSING et al., 1994; PINNSCHMIDT et al., 1995). No caso do patossistema estudado, um dos desafios ainda restantes é a relação do nível de injúria com a radiação interceptada (RI) e a eficiência do uso da radiação (RUE - *Radiation use efficiency*) como pontos de acoplamento representando o dano (BOOTE et al., 1983). Verificada a relação, o modelo de simulação do patossistema *Puccinia*-Trigo poderá ser usado para estimar danos devidos à ferrugem da folha. Para isso, o modelo proposto contabiliza o crescimento do fungo como sendo lesão visível e não visível (lesão virtual), a primeira mais diretamente relacionada com a RI e a segunda, com a RUE.

O conceito de lesão virtual foi introduzido por Bastiaans (1991), quando da formalização da relação entre a fotossíntese e a proporção de área foliar doente, representando a técnica de predizer a fotossíntese de uma folha de trigo infectada com uma simples doença (ROBERT et al., 2004; NUTTER et al., 2006). Com base neste conceito, verifica-se que melhorias na forma de avaliação da doença se fazem necessárias, minimizando a subjetividade geralmente encontrada na quantificação da severidade de doenças, a qual é, na maioria das vezes, obtida de forma manual e visual.

Escalas diagramáticas têm sido propostas com o intuito de padronizar as observações, constituindo-se como uma das principais ferramentas de avaliação (LEITE; AMORIM, 2002; MARTINS et al., 2004). Apesar de se mostrarem como importantes ferramentas de au-

xílio, o erro está geralmente presente e as observação não se mostram precisas, ou seja, não contabilizam, por exemplo, a extensão da lesão virtual (ELINGS et al., 1999).

Os modelos de simulação refletem os dados que os alimentam, sendo extremamente dependentes de informações confiáveis e precisas, justificando-se, dessa forma, o uso de técnicas mais sofisticadas, como o registro da severidade real (uso de softwares e equipamentos óticos), a densidade de lesões por cm^2 (NUTTER et al., 2006) e a obtenção da real extensão das lesões (ELINGS et al., 1999) (lesões visíveis e virtuais), pela quantificação do DNA do fungo (SHETTY et al., 2007), pela quantificação de ergosterol (MORAES et al., 2003) e/ou pela termografia digital (LINDENTHAL et al., 2005). Com esses dados mais detalhados, modelos poderiam ser melhor calibrados, aliados às condições climáticas, podendo ser utilizados na previsão de epidemias e no auxílio à tomada de decisões.

Uma das variáveis de maior importância para os modelos de simulação de doenças é o período de molhamento foliar (PMF). Tal importância se deve ao fato de que o processo epidemiológico de doenças de plantas está intimamente relacionado ao PMF, uma vez que a maioria dos patógenos necessita da presença de água livre sobre as plantas para a germinação dos esporos e para a infecção do tecido do hospedeiro. Portanto, esta variável é fundamental para a identificação do potencial risco de ocorrência de doenças (SENTELHAS, 2004).

Com isso, além da importância de uma “boa” e completa coleta de dados de campo, informações sobre o PMF são de extrema importância para que os modelos reflitam o real progresso de epidemias, sendo, inclusive, importantíssimos na validação desses modelos. Esta é uma das variáveis mais difíceis de se obter pelo fato de poucas esta-

ções meteorológicas estarem preparadas para tal; entretanto, quando disponíveis, a variabilidade espacial pode vir a dificultar a sua utilização, em razão do grande número de variáveis que interferem na presença de água livre na superfície dos órgãos das plantas (KIM et al., 2005; SENTELHAS et al., 2005). A determinação do PMF pode ser realizada de diversas maneiras, desde medidas diretas, com uso de sensores, ou por meio de estimativas baseadas em relações empíricas (SENTELHAS, 2004; KIM et al., 2004). Neste trabalho utilizou-se como PMF uma estimativa baseada no número de horas com a umidade relativa acima de 90% (GLEASON et al., 1994), de modo que o modelo pudesse ser validado; porém, com dados mais precisos e mais confiáveis acredita-se que o modelo poderia ser melhor calibrado.

O uso de uma escala hierárquica (órgão, planta e campo) para controle do inóculo disponível e novas infecções mostrou-se uma técnica realística, na medida em que o ajuste dos parâmetros confirma as observações feitas em campo. Diferentes combinações de parâmetros podem ser necessárias para refletir as realidades de cada cultura, local e época do ano. Com pouco esforço, o modelo pode ser estendido para refletir outros fenômenos, como a vinda de inóculos de outras regiões carregados pelo vento (SACKETT; MUNDT, 2005).

O estudo das oscilações do ENSO (*El Niño/Southern Oscillation*) mostra-se como um dos mais importantes e melhores mecanismos de caracterização das variações climáticas globais, tendo em vista que essas oscilações geralmente afetam as temperaturas regionais, assim como influenciam nos padrões de precipitação, sendo as principais responsáveis pelo desenvolvimento de uma epidemia. Muitas das observações registradas sobre epidemias ao longo dos anos podem apresentar sinais de interferência do fenômeno (SCHERM; YANG, 1995).

Sabendo-se que a região de Passo Fundo (Rio Grande do Sul, Brasil) é uma das afetadas pelas oscilações do ENSO (BERLATO; FONTANA, 2003), um exercício foi realizado com o modelo de simulação desenvolvido, o qual foi rodado para vários anos (a partir de 1979), podendo-se analisar os resultados obtidos com alguns dados de severidade registrados ao longo do tempo (PICININI et al., 1996). As comparações realizadas demonstram que o modelo pode ser usado para estudar a variabilidade interanual das doenças, assim como estudos com mudanças climáticas. O modelo também pode ser utilizado com prognósticos climáticos a fim de determinar, estatisticamente, por exemplo, as melhores datas de semeadura. Com o seu uso, podem-se gerar cenários futuros para subsidiar tomadas de decisão e definir correções no manejo da cultura em função da evolução do quadro climático.

Com o desenvolvimento do simulador genérico para a ferrugem do trigo utilizando-se da modularidade, verificou-se a possibilidade de que mais de uma doença possa ser incorporada, ao mesmo tempo, ao modelo proposto, bastando para isso descrevê-las em arquivos de configuração XML. Podem-se configurar também diferentes tipos de manejo e datas de semeadura, tornando-o, além de genérico, adaptável às necessidades e condições do usuário.

Além de verificar as estimativas de crescimento do trigo e a incidência de doenças, pela união entre o modelo CropSim-Wheat e o modelo genérico de doenças, esta proposta poderá ser útil à realização de outros trabalhos, como teste de hipóteses, estratégias de controle de doenças (MAY et al., 2005), avaliação da resistência varietal a produtos químicos (YANG, 2003), avaliação do seu impacto na produtividade (ROERMUND; SPITTERS, 1990), quantificação de danos, análise de séries históricas e projeção de danos (SAVARY et

al., 2006). Pode ser utilizado, por exemplo, no desenvolvimento de sistemas de auxílio à tomada de decisão (AUDSLEY et al., 2005).

CAPÍTULO III

SISALERT - UMA PROPOSTA METODOLÓGICA PARA O AUXÍLIO À TOMADA DE DECISÃO NO MANEJO DE DOENÇAS

Willingthon Pavan

Faculdade de Agronomia e Medicina Veterinária

Universidade de Passo Fundo

C.P. 611, 99001-970

RESUMO

Previsão de doenças tem surgido como um componente bem estabelecido no manejo de doenças. Atualmente, um grande número de modelos de simulação de epidemias encontra-se disponível, porém poucos têm sido utilizados realmente como ferramentas de auxílio na tomada de decisões. Entretanto, os modelos de simulação de epidemias podem ter uma aplicação de forma mais generalizada se associados à pronta disponibilização de observações meteorológicas, aos prognósticos climáticos, às técnicas computacionais e aos simuladores orientados aos processos fisiológicos das plantas. Este trabalho tem como objetivo apresentar uma metodologia para o desenvolvimento de sistemas de auxílio à tomada de decisão capazes de reaproveitar o conhecimento legado, aliando-se modernas tecnologias computacionais em prol de um adequado manejo de doenças. A metodologia proposta foi baseada no *design pattern* MVC (Modelo-Visão-Controle), programação em camadas, linguagem orientada a objetos, entre outras tecnologias, dividida em servidores específicos para cada atividade, configurando-se como uma plataforma modular e adaptável. Foi utilizado neste trabalho o patossistema *Gibberella*-Trigo, em virtude da

importância econômica e das grandes perdas quando da ocorrência de epidemias. A sua implantação e validação deram-se numa importante região tritícola do país, a região dos Campos Gerais do Paraná, sob cobertura e responsabilidade da Fundação ABC, abrangendo 38 municípios e utilizando dez estações meteorológicas automáticas. A plataforma foi projetada para ser acessada das mais variadas formas possíveis, porém nesta validação utilizaram-se processamento em lote (lado do servidor) e interface Web (lado do cliente). A plataforma mostrou-se metodologicamente eficiente e potencialmente aplicável, resultando numa ferramenta de grande utilidade para o auxílio à tomada de decisões, provocando, com o sucesso alcançado, a expansão e a aplicação para outras regiões do país já na safra de 2007. Mostrou-se também aplicável a outras áreas e culturas, como por exemplo a maçã.

ABSTRACT

SISALERT - A METHODOLOGICAL APPROACH FOR DECISION MAKING ON PLANT DISEASE MANAGEMENT

Disease forecasting has become an established component of quantitative epidemiology and is a rapidly developing area in disease management. Currently, a great number of plant disease models are available, however few have really been used in decision making. However, plant disease models can have a more practical application by taking the advantage of combining on-line weather data, weather forecast, computational techniques and biophysical models. This work has as objective to present a novel approach for the development of a decision support system based on existing knowledge and up to date computational technologies aiming better disease management. The methodology proposed was mainly based on design pattern MVC (Model-Vision-Control), programming layers and object oriented languages. Specific servers were designed for different tasks. These was conceived as a modular and adaptable platform. The pathosystem *Gibberella*-Wheat was used in this work due to its economic importance and great losses observed in epidemics years. Its was implemented and

validated in an important wheat growing region in Brazil, known as the Campos Gerais of Paraná. The ABC Foundation, a farmer cooperative branch was the institution responsible for the system. The test enclosed 38 cities and a network of ten automatic meteorological stations. The platform was projected to be accessed of several forms, however in this test was used processing in batch (server side) and Web interface (client side). The platform revealed to functionally efficient and potentially applicable, resulting in a tool of great utility for use in decision aid. As result in 2007, it is been expanded to other wheat growing regions in the country. The approach used also was exported to other crops, as for example, apples.

Introdução

Previsão de doenças tem surgido como um componente bem estabelecido da epidemiologia quantitativa e que rapidamente está sendo incorporado ao manejo de doenças. A matemática do progresso de doenças tem amadurecido ao ponto de se tornar um poderoso e respeitado componente no manejo e na previsão de epidemias. Entretanto, muitos modelos de previsão de doenças de plantas não têm preenchido as expectativas de que teriam o papel principal no manejo de doenças. Entre as diversas razões encontra-se a premissa de que um modelo de previsão de doenças faça projeções futuras dos principais eventos no desenvolvimento das doenças, o que a maioria dos modelos de previsão não faz (SEEM, 2001).

Uma nova faceta nessa área é a possibilidade de se utilizarem de prognósticos climáticos como entrada em modelos de simulação de doenças para realmente fazer a previsão. Com a melhoria dos prognósticos climáticos, uma estimativa mais exata de variáveis importantes para modelos de doenças de plantas, tais como precipitação e duração do molhamento foliar, será possível fornecer estimativas sazonais sobre a probabilidade da ocorrência das doenças e previsões de severas epidemias. Isso é especialmente interessante para o manejo de doenças, tendo em vista que pulverizações desnecessárias, seguindo calendários, trazem um impacto significativo no custo de produção, além de que aplicações fora do momento apropriado podem resultar num inadequado controle da doença.

A modelagem computacional tem possibilitado o desenvolvimento de ferramentas que utilizam dados climáticos para prever epidemias. Tradicionalmente, modelos de doenças de plantas têm utilizado o acúmulo de horas de molhamento, combinado com os requisitos de

temperatura, para prever o processo de infecção e colonização e, então, identificar os riscos de uma epidemia. Esses tipos de modelos utilizam-se de registros climáticos observados a fim de rastrear quando uma quantidade favorável de horas ocorreu, indicando táticas ou estratégias de controle. Esses modelos têm sido usados como parte de sistemas de manejo integrado de doenças (JABRZEMSKI; SUTHERLAND, 2006).

Um manejo de doenças com maior eficiência requer o uso de múltiplas estratégias, incluindo o desenvolvimento de um sistema racional para a previsão de epidemias. Este pode ser uma ferramenta de grande utilidade para agricultores e agentes de extensão no sentido de apoiar a tomada de decisão no controle das doenças no campo, com a utilização de fungicidas, ou mesmo para conhecer previamente o nível de incidência da doença (VARGAS et al., 2000; WOLF; FRANCL, 2000).

Atualmente, os revolucionários avanços tecnológicos na informática e o surgimento da internet, com a conectividade global e a integração com linguagens modernas de programação, têm produzido novos conceitos e caminhos a serem ainda explorados na produção e transferência do conhecimento. A revolução nas tecnologias baseadas na Web está trazendo grandes mudanças no desenvolvimento e no emprego de sistemas de suporte à tomada de decisão para produtores e especialistas no manejo de doenças (FERNANDES et al., 2007, 2007b).

O presente trabalho apresenta uma metodologia no sentido do uso de linguagens de programação e tecnologias recentes visando ao desenvolvimento de um sistema baseado na Web para a implementação, execução e disponibilização de modelos. SISALERT é um sistema

multimodelo que se utiliza de dados climáticos, obtidos de estações meteorológicas automáticas, e de prognósticos horários para o processamento de sofisticados modelos de avaliação do risco de epidemias. Esses modelos interpretam os dados climáticos, fornecendo informações sobre o comportamento passado ou recente da doença, assim como a predição do nível de risco da epidemia. A chave do formato do SISALERT é a modularidade, a qual permite o acoplamento dos modelos de crescimento e desenvolvimento de culturas e os modelos de doenças, dependendo da cultura e da doença em destaque.

Em virtude das grandes perdas que provoca na cultura do trigo, a giberela é uma doença amplamente estudada e conhecida no Brasil. Ataca o trigo em locais com condições de umidade, causando severas perdas no rendimento e na qualidade de grãos de cultivares suscetíveis (OSÓRIO et al., 1998; PANISSON et al., 2003; CASA et al., 2004). Muitos trabalhos podem ser encontrados sobre o assunto, como os realizados por Emerson Del Ponte, o qual descreveu inclusive um modelo para a simulação desta importante doença (DELPONTE et al., 2004, 2005).

Conforme DelPonte et al. (2004), dentre as medidas atualmente empregadas para o controle da giberela, tem-se constatado que nenhuma tática isolada de manejo é eficiente para minimizar as perdas. Mesmo se tendo conhecimento da existência de resistência genética em alguns cultivares, esta pode ser insuficiente para evitar níveis epidêmicos (DELPONTE et al., 2004), além de o controle por uso de fungicidas tornar-se difícil, pois as anteras, principal porta de entrada do fungo, expõem-se não simultaneamente, possuindo uma janela de floração (antese) por um grande período, que chega a um mês (OSÓRIO et al., 1998). Por isso, a eficiência de fungicidas é extremamente dependente do momento e da qualidade da aplicação. Segundo Del-

Ponte et al. (2004), um manejo da doença que busque a maior eficiência deve considerar o manejo integrado, que, certamente, o modelo de simulação deve incluir.

Com este trabalho, busca-se demonstrar a utilidade dos modelos de simulação como auxiliares na otimização da tomada de decisões e no controle das doenças, considerando-se a época, o curso e a severidade da epidemia. Os objetivos do trabalho são: (i) utilizar modelos genéricos e modulares de doenças de plantas, acoplados à fenologia do hospedeiro; (ii) rodar modelos usando registros meteorológicos em tempos quase real e de prognósticos; (iii) disponibilizar informações sobre grupos de risco de ocorrência de epidemias; (iv) demonstrar a aplicação do SISALERT no manejo da giberela do trigo.

Material e Métodos

Plataforma

O SISALERT foi desenvolvido baseado no *design pattern* MVC (Model-View-Controller - Modelo-Visão-Controle), o qual é um modelo de desenvolvimento de aplicações que utiliza as características da programação em camadas, sendo dividido em três camadas, ou áreas funcionais: Modelo, Visão e Controle (VEIT; HERRMANN, 2003). O Modelo representa a “lógica do negócio”, ou seja, o estado e o comportamento dos componentes, gerenciando e conduzindo todas transformações; a Visão disponibiliza os dados produzidos pelo Modelo, gerenciando o que pode ser visto do seu estado, apresentando-os em forma de imagens, gráficos e dados tabulados, através de páginas Web ou equipamentos de pequeno porte, como celulares e Palms; e o Controlador determina o fluxo total da aplicação, gerenciando a

interação do usuário/sistema com o Modelo (Figure 33).

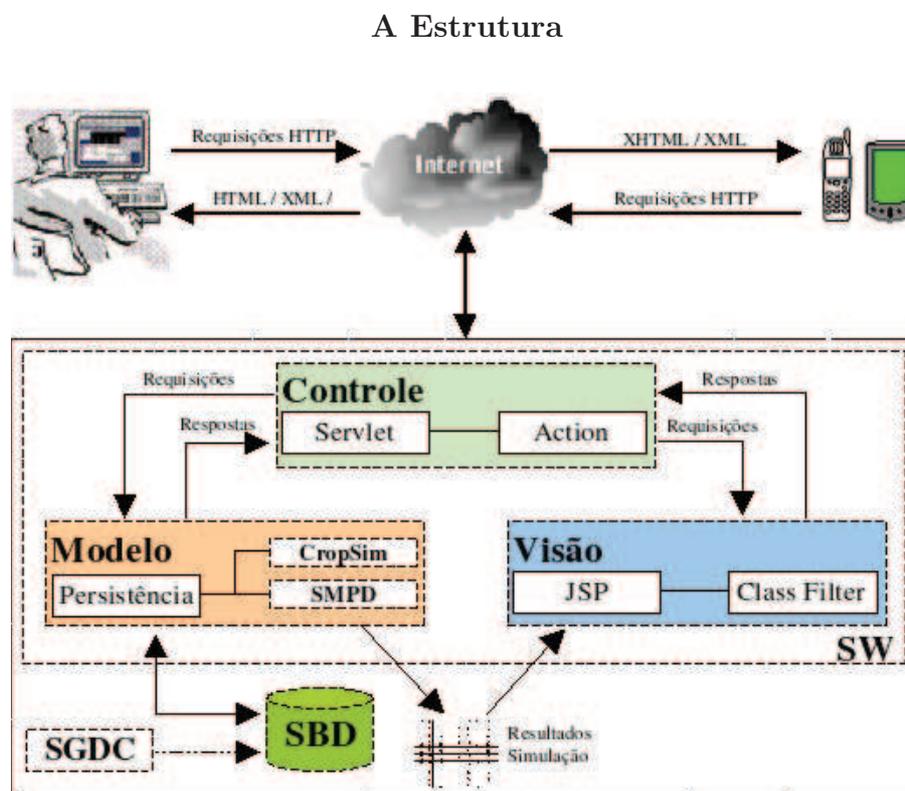


Figura 33: Estrutura de implementação do sistema SISALERT

Para garantir a execução do SISALERT, servidores foram projetados para processar as requisições feitas pelos demais. Estes servidores dividem-se em: (i) servidor de gerenciamento de dados climáticos (SGDC); (ii) servidor de banco de dados (SBD); (iii) servidor de modelos de previsão de doença (SMPD), em particular, nesta proposta, para a giberela, causada pelo fungo *Gibberella zeae* (DELPONTE et al., 2004); (iv) servidor Web (SW); (v) servidor de modelo de simulação de culturas (MSC), neste caso, o modelo de crescimento e desenvolvimento do trigo, o CROPSIM-Wheat (HUNT; PARARAJASINGHAM, 1995; HUNT, 2007).

O SGDC consistiu num módulo para a recuperação de dados climáticos de estações meteorológicas automatizadas, situadas em locais remotos, e um módulo para atualização dos dados no SBD. O módulo de recuperação de dados foi executado a cada hora pelo comando *crontab* do Linux, procurando por arquivos de dados gerados por estações meteorológicas em diretórios pré-configurados nos *scripts* do módulo, baixando-os via Web para o servidor.

Para facilitar o acesso aos dados das estações, um programa específico foi desenvolvido e implantado em computadores remotos, concentrando os dados de uma determinada rede de estações. Este foi desenvolvido de forma genérica e configurável, podendo ser implantado em qualquer tipo de sistema operacional e configuração de estações meteorológicas, filtrando, validando e preparando os dados para armazenamento e transferência (arquivos texto). Robôs de busca e coleta também foram desenvolvidos para capturar dados disponíveis na Web, tanto de domínio público (INMET - Instituto Nacional de Meteorologia) como privados, com autorização.

Uma vez baixados, os dados foram processados pelo módulo de atualização, sendo o resultado desse processamento os dados climáticos, atualizados no SBD. Além do processamento dos dados das estações meteorológicas, prognósticos climáticos horários para cinco dias também foram processados. Esses dados de prognósticos foram fornecidos pelo INPE (Instituto Nacional de Pesquisas Espaciais) e também capturados por robôs de busca. Além dos prognósticos fornecidos pelo INPE, para algumas localidades arquivos disponibilizados pelo CPTEC (Centro de Previsão de Tempo e Estudos Climáticos) foram processados por demanda.

O SBD armazenou os dados, assim como os identificadores das es-

tações climáticas e os parâmetros da execução, como o cultivar, datas de semeadura, culturas anteriores e outros dados utilizados pelo MSC (CROPSIM-Wheat). Os dados armazenados no SBD foram disponibilizados aos demais servidores por meio de requisições SQL (Structured Query Language). O SBD estava preparado para gerenciar todos os dados de entrada necessários para a execução dos modelos de simulação, além de armazenar as saídas dos mesmos.

Na camada de visão (SW) as informações foram recuperadas por meio de requisições à camada de modelo, não existindo uma ligação forte entre a visão e o SBD, sendo essas requisições fruto de interfaces disponibilizadas aos usuários por sistemas Web, acessados através de browsers. A estrutura projetada foi do estilo cliente-servidor, estando, de um lado, a interface com o usuário e, do outro, os servidores de aplicação.

O passo do sistema pode ser tanto horário como diário, utilizando para o processamento dados climáticos históricos, correntes e de prognósticos.

A Implementação

O patossistema *Gibberella*-Trigo foi usado para testar a viabilidade da proposta metodológica. A giberela é uma doença que se destaca no cultivo do trigo, causada pelo fungo *Gibberella zeae*, e está intimamente relacionada com eventos climáticos, como temperatura, precipitação e umidade relativa, principalmente no estágio fenológico de floração (DELPONTE et al., 2004).

Nesta proposta metodológica criou-se uma estrutura composta por diversas tecnologias computacionais (linguagens de programação,

bancos de dados etc.) e dois modelos de simulação: um de crescimento e desenvolvimento do trigo e um modelo de simulação para a giberela. Para este último seguiu-se o modelo de giberela apresentado por DelPonte et al. (2005), implementando-o dentro da proposta apresentada no capítulo ??, de forma a possuir características modulares e uma estrutura genérica, podendo ser parametrizada para adequação às condições específicas de cada cultivar ou local, visando representar o mais realisticamente possível as observações realizadas em campo.

Como modelo de crescimento e desenvolvimento utilizou-se o CropSim-Wheat, substituto do Ceres-Wheat, um dos modelos de simulação pertencentes à suite do DSSAT, para a simulação do crescimento e desenvolvimento do trigo. Este modelo tem sido testado, calibrado e validado por pesquisadores de diversas partes do mundo para simular o desenvolvimento e o crescimento de cultivares de trigo (HUNT; PARARAJASINGHAM, 1995; FERNANDES et al., 2004; DELPONTE et al., 2005).

Para o armazenamento das informações foi usado o Sistema Gerenciador de Banco de Dados Oracle 10g (<http://oracle.com>), banco de dados objeto relacional, onde foi armazenada a base de dados do sistema, tanto dados de entrada como de saída. Foram criadas tabelas adicionais e especiais na base de dados para esta proposta, diferentemente do padrão das demais bases de dados criadas, tendo em vista a integração com os sistemas de auxílio à tomada de decisão da Fundação ABC (Figuras 34, 35, 36, 37 e 38).

O banco de dados foi hospedado num servidor Intel Xeon 2.4 Dual (Bi-Processado), com 3 GB de memória RAM e sistema operacional Linux Fedora Core 5 (<http://fedoraproject.org>), integrante de uma estrutura formada por sete servidores, cada um com responsabilidades

```

TABLE ESTACAO
(
  COD_ESTACAO VARCHAR2(20 BYTE) NOT NULL,
  TIPO VARCHAR2(150 BYTE),
  ALTITUDE INTEGER,
  CORDENADA_METRICA_N VARCHAR2(30 BYTE),
  CORDENADA_METRICA_E VARCHAR2(30 BYTE),
  MUNICIPIO VARCHAR2(150 BYTE),
  NOME VARCHAR2(100 BYTE),
  COD_INMET VARCHAR2(50 BYTE),
  LATITUDE FLOAT(126),
  LONGITUDE FLOAT(126)
)

```

Figura 34: Tabela que armazena os dados das estações meteorológicas automáticas com suas coordenadas geográficas e município onde estão instaladas.

```

TABLE MUNICIPIO
(
  COD_MUNICIPIO INTEGER NOT NULL,
  NOME VARCHAR2(200 BYTE) NOT NULL,
  ALTITUDE NUMBER(12,2),
  COD_ESTACAO_REFERENCIA VARCHAR2(20 BYTE) NOT NULL
)

```

Figura 35: Tabela que armazena dados dos municípios abrangidos pelo sistema.

```

TABLE PREVISAO
(
  COD_PREVISAO INTEGER NOT NULL,
  DATA DATE,
  PRESSAO FLOAT(126),
  TEMPERATURA FLOAT(126),
  RADIACAO FLOAT(126),
  UMIDADE_RELATIVA FLOAT(126),
  VENTO FLOAT(126),
  PRECIPTACAO FLOAT(126),
  COD_MUNICIPIO INTEGER,
  COD_ESTACAO VARCHAR2(200 BYTE)
)

```

Figura 36: Tabela que armazena prognósticos climáticos previamente processados.

específicas: um servidor de segurança de rede (Firewall¹), responsável por proxy de internet, antivírus e roteamento entre as redes; um servidor de aplicação², tanto para a intranet como para a internet, assim

¹Firewall é o nome dado ao dispositivo que tem por função regular o tráfego de rede entre redes distintas, com o objetivo de impedir a transmissão de dados nocivos ou não autorizados de uma rede a outra.

²Servidor de aplicação é um software que disponibiliza um ambiente para a instalação e execução de certas aplicações, geralmente para aplicações Web.

```

TABLE DADOS_ESTACAO
(
  DATA DATE NOT NULL,
  PRESMED NUMBER(12,2),
  PRESMIN NUMBER(12,2),
  PRESMAX NUMBER(12,2),
  TMED NUMBER(12,2),
  THIN NUMBER(12,2),
  THAX NUMBER(12,2),
  TSOLO NUMBER(12,2),
  RAD NUMBER(12,2),
  UR NUMBER(12,2),
  VTMED NUMBER(12,2),
  DIRVTMED NUMBER(12,2),
  VTHAX NUMBER(12,2),
  DIRVTHAX NUMBER(12,2),
  PRECIPACUM NUMBER(12,2),
  PRECIP NUMBER(12,2),
  DPM NUMBER(12,2),
  THOL NUMBER(12,2),
  COD_ESTACAO VARCHAR2(20 BYTE) NOT NULL,
  HORA TIMESTAMP(6) NOT NULL,
  URMAX NUMBER,
  URMIN NUMBER,
  PTORVALHOMIN NUMBER,
  PTORVALHOMAX NUMBER,
  PTORVALHOMED NUMBER
)

```

Figura 37: Tabela que armazena informações climáticas coletados pelas estações meteorológicas.

```

TABLE GIBERELA
(
  COD_ESTACAO VARCHAR2(20 BYTE),
  COD_MUNICIPIO VARCHAR2(20 BYTE),
  DATA_SIMULACAO DATE,
  DATA_ESPIGAMENTO DATE,
  DATA_PROCESSAMENTO DATE,
  SEVERIDADE FLOAT(126),
  ANTERAS_VIVAS FLOAT(126),
  ANTERAS_ACUMULADAS FLOAT(126),
  RADIACAO FLOAT(126),
  PRECIPITACAO FLOAT(126),
  UMIDADE_RELATIVA FLOAT(126),
  TEMPERATURA FLOAT(126)
)

```

Figura 38: Tabela que armazena resultados do modelos de simulações.

como serviços de e-mail, utilizando Conectiva Linux 10; um servidor de dados geoespaciais³, utilizando o Conectiva Linux 10; um servidor de backup utilizando Conectiva Linux 10; um servidor com Microsoft Windows^(r) 2003 Server, para armazenamento dos documentos e sistemas internos; e um servidor anti-spam, sob Linux Fedora Core 5. O

³Dados geoespaciais indicam a geometria e a topologia de objetos espaciais ou de coordenadas geográficas, ou seja, indicam atributos de objetos espaciais.

acesso aos dados deu-se por um link de internet de 1 Mbps (Figura 39), onde foi recebida uma média de 180 acessos diários no período da safra.

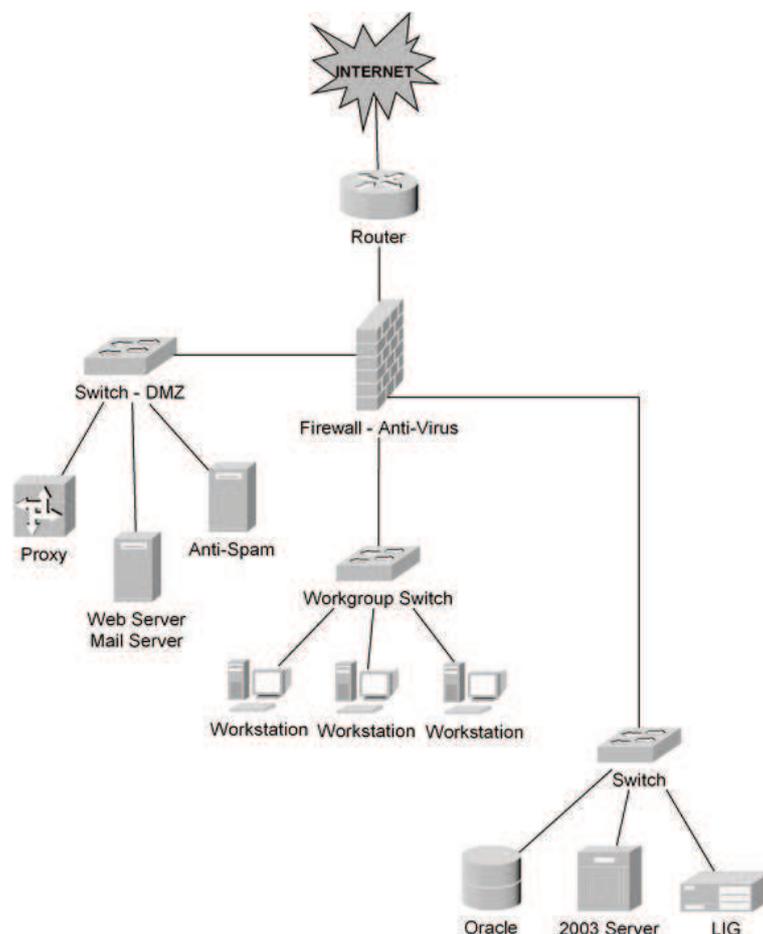


Figura 39: Topologia de servidores utilizado

Foram envolvidos neste estudo 38 municípios da região dos Campos Gerais do Paraná, os quais foram cobertos por dez estações meteorológicas automáticas (Figura 40), formando uma rede de estações que forneceu dados como precipitação, radiação solar, temperatura, umidade relativa, velocidade e direção do vento, entre outros. Os dados oriundos das estações meteorológicas foram processados, ana-

lisados e armazenados na base de dados. Para as localidades que não possuíam estações meteorológicas foram realizadas interpolações para a geração dos dados faltantes.

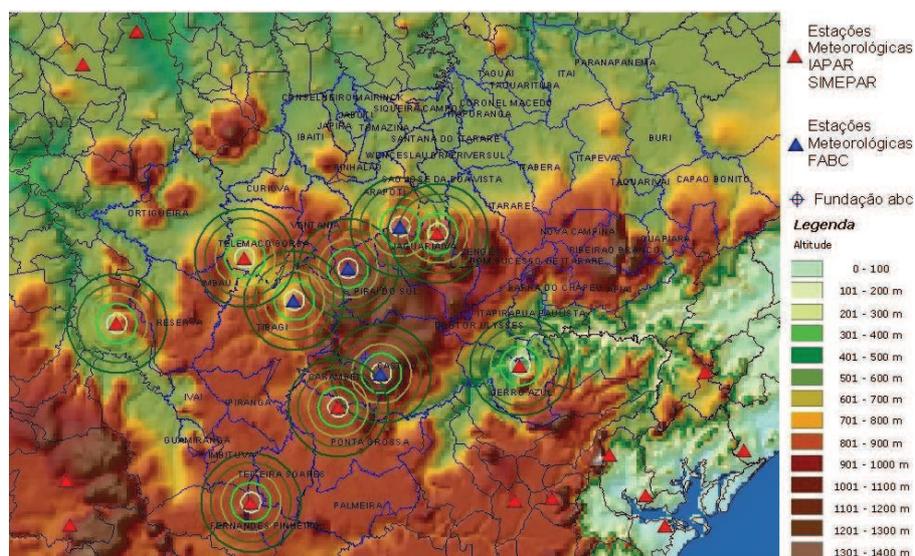


Figura 40: Estações meteorológicas e áreas de abrangência.

Os prognósticos climáticos utilizados neste estudo foram obtidos do INPE/CPTEC, gerados pelo modelo ETA, um modelo de mesoescala, em ponto de grade, de equações primitivas. A versão do modelo ETA é hidrostática e cobre a maior parte da América do Sul e oceanos adjacentes (GUARNIERI et al., 2006). Os resultados do modelo ETA foram baixados, gratuitamente, do servidor de FTP dos institutos, através do endereço ftp://tucupi.cptec.inpe.br/pub/produtos/prod_eta/eta_sx6/regions/SU/, processados e armazenados na base de dados do sistema, de acordo com coordenadas geográficas previamente estabelecidas, conforme as estações e municípios.

Os arquivos disponibilizados pelo INPE/CPTEC estavam num formato baseado na linguagem de scripts GrADS (Grid Display Ana-

lisy System), o qual é essencialmente um software para visualização e análise de dados em pontos de grade. Tratando-se de visualização, este é o software mais utilizado na área de pesquisa e meteorologia, de distribuição gratuita, podendo ser implementado em diversos sistemas operacionais, tais como Unix, Linux, Windows e DOS (CHAVES; RODRIGUEZ, 2007). Para este caso foi utilizado o GrADS, na versão 1.8 SL 11, instalado num sistema operacional Windows^(r)XP Profissional.

Além das tecnologias já mencionadas, foram utilizadas as linguagens: (i) Java, para o desenvolvimento dos modelos de simulação (CORNELL; HORSTMANN, 1997; SRIDHARAN et al., 1997); (ii) PHP, para o desenvolvimento da interface Web (CONVERSE, 2003); (iii) Flash, para a geração de interface gráfica em páginas Web (UNDERDAHL, 2003); (iv) XML (*EXtensible Markup Language*), para armazenar os parâmetros do modelo de simulação de giberela e do ambiente (DORNELES et al., 2004).

O trabalho foi realizado com o auxílio da Fundação ABC (Paraná, Brasil), instituição de pesquisa e assessoria técnica, vinculada às Cooperativas Agro-Pecuária Capal, Batavo e Castrolanda, a qual desenvolve trabalhos em agricultura de precisão, fertilidade do solo, fitotecnia e sistemas de informação, dentre outras.

Resultados

O patossistema *Gibberella*-Trigo, escolhido para esta proposta, foi modelado segundo as classes estabelecidas na Figura 41. A sua implementação deu-se seguindo a estrutura definida por Porter et al. (1999), decompondo-o de forma a torná-lo modular. Para cada módulo fo-

ram criadas as seções de inicialização, taxas, integração e saída. Um fluxograma sumarizado do simulador implementado pode ser visto na Figura 42.

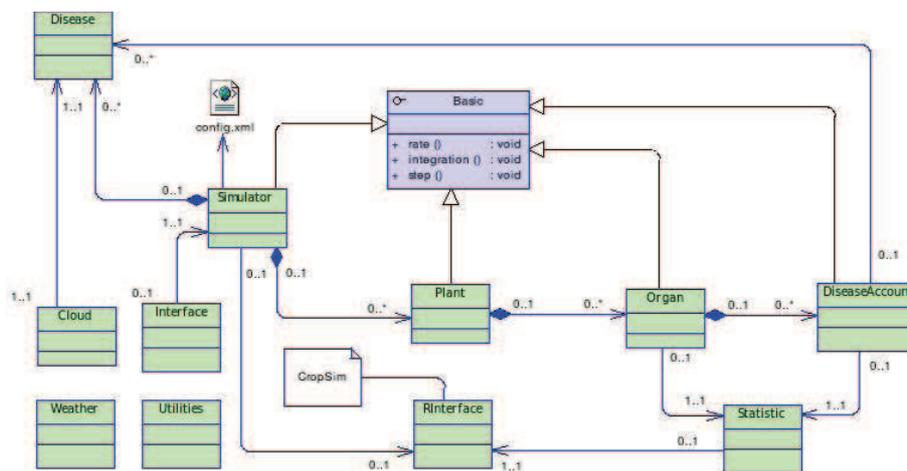


Figura 41: Fluxograma de funcionamento do SISALERT

O processamento dos modelos de simulação foi realizado sem uma interface com o usuário, configurado através de arquivos XML (data inicial, data final, intervalo entre as datas para a simulação, utilização de prognóstico climático etc.), devidamente armazenados junto da aplicação (simulador) no servidor de aplicação. Pelas informações contidas nas tabelas do banco de dados, o sistema executou as operações programadas, conforme as chamadas do *cron*⁴ do sistema operacional, sendo os resultados das simulações armazenados na base de dados para posterior utilização no sistema de auxílio à tomada de decisões (Figuras 45 e 46).

Os valores dos parâmetros utilizados nos modelos de simulação criados e no desenvolvimento da plataforma sistema SISALET foram

⁴O cron é um recurso presente nos sistemas operacionais da família Unix/Linux que permite a execução automática de tarefas/programas em datas e horários predeterminados.

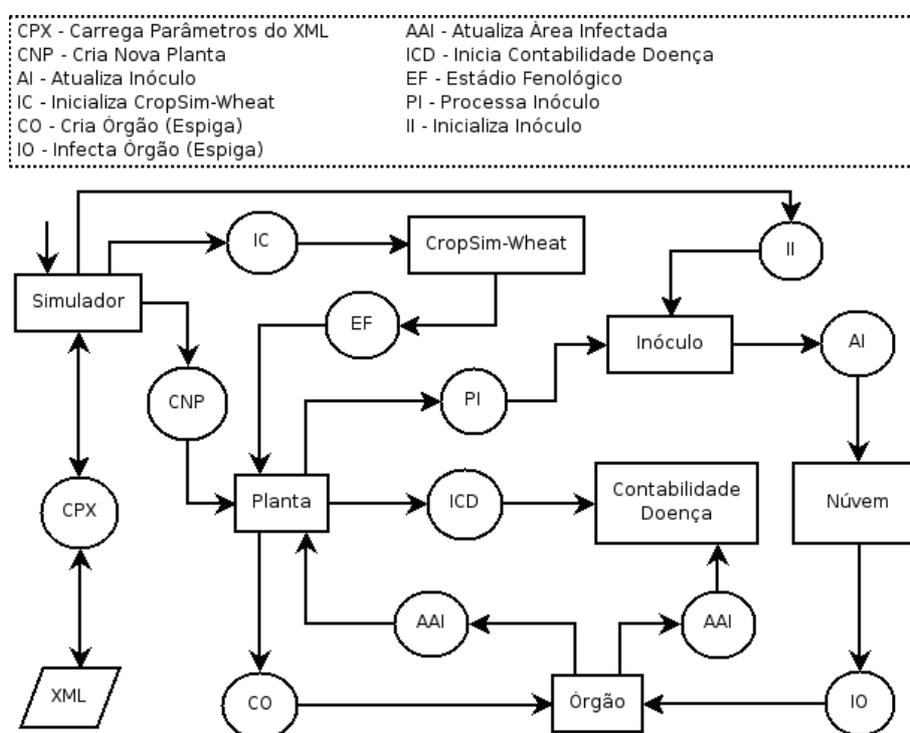


Figura 42: Fluxograma de funcionamento do SISALERT

retirados da literatura (DELPONTE et al., 2004, 2005), de resultados de trabalhos realizados em projetos de pesquisa (PAVAN et al., 2006), ou obtidos por meio de estimativas aproximadas e com algum sentido biológico.

No que diz respeito à obtenção de dados de prognósticos climáticos, o processo ocorreu da seguinte forma (Figura 43): (i) duas vezes ao dia foi realizado, com o uso robôs (agentes de software desenvolvidos), o download dos arquivos resultantes das execuções dos modelos numéricos de previsão de tempo, do INPE/CPTEC; (ii) os arquivos foram filtrados e processados pelo uso do software GrADS; (iii) o resultado foi verificado e analisado baseando-se em padrões estabelecidos e conhecidos para as variáveis climáticas, eliminando-se toda e qualquer discrepância que pudesse ter sido gerada pelos modelos; (iv)

então, os dados faltantes foram recalculados e as estimativas obtidas de hora em hora; (v), por fim, os prognósticos horários para cinco dias foram armazenados na tabela PREVISÃO (Figura 36), do banco de dados do sistema.



Figura 43: Obtenção de prognósticos climáticos do INPE/CPTEC

Para os dados das estações meteorológicas, a obtenção ocorreu por meio de robôs de busca, que capturaram os dados de cada uma das estações e alimentaram o banco de dados. Rotinas específicas foram responsáveis por verificar a integridade dos dados e realizar a interpolação para os dados faltantes (municípios que não possuíam estações automáticas instaladas). Após esse processamento, realizado de hora em hora, o sistema pôde contar com os dados para o processo das simulações (Figura 44).

Mesmo com a possibilidade de ligação do modelo com o CropSim-Wheat para o desenvolvimento da ferramenta de auxílio à tomada de decisão com a Fundação ABC, optou-se por utilizar apenas a data de espigamento. Como na região de abrangência da fundação havia uma diversidade de datas de semeadura, resultando em diferentes datas de espigamento, configurou-se o SISALERT para iniciar o seu processamento, para cada município, a partir do dia 1º de agosto de 2006. A execução do SISALERT foi realizada duas vezes ao dia, sempre iniciando no dia mencionado e prosseguindo até o dia da execução, mais cinco dias (prognósticos), ou seja, a cada dia todo o processamento

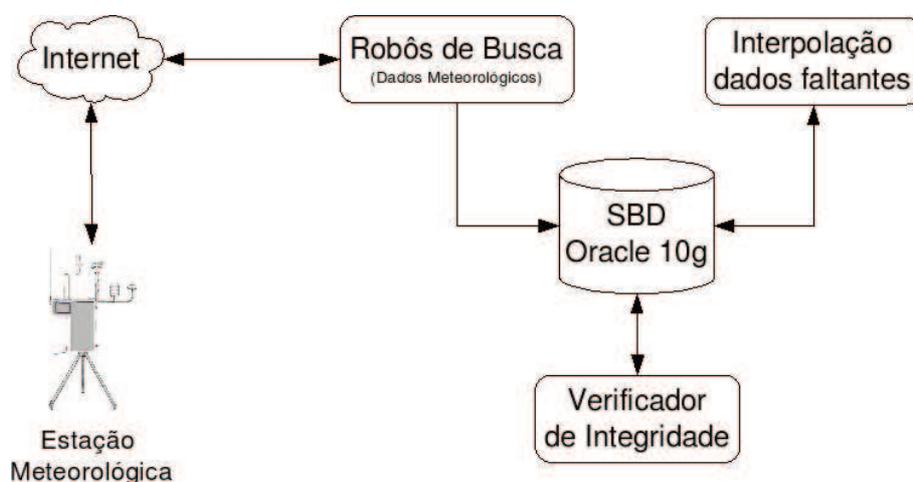


Figura 44: Obtenção de dados das estações meteorológicas automáticas

era refeito, armazenando-se novamente, para cada município e data, os resultados do modelo.

Dividiu-se o sistema Web, principalmente, em dois módulos. No primeiro, um programa em linguagem Flash foi desenvolvido, solicitando que o usuário fizesse a seleção do dia em que houvera a emergência da primeira espiga, sendo solicitado em seguida o município desejado a fim de obter os resultados da simulação (Figura 45). Com a informação oferecida pelo usuário, um programa, desenvolvido em PHP e hospedado no servidor de aplicação, era executado, retornando um conjunto de dados, em formato XML, ao programa Flash, sendo processados e visualizados em forma de gráficos (Figura 46). No primeiro gráfico pôde-se visualizar a severidade da doença; no segundo, a liberação das anteras; no terceiro, as anteras acumuladas; no quarto, a precipitação registrada para a localidade e, no quinto, a temperatura média e a umidade relativa.

O processamento do SISALERT, neste caso para a giberela, ini-

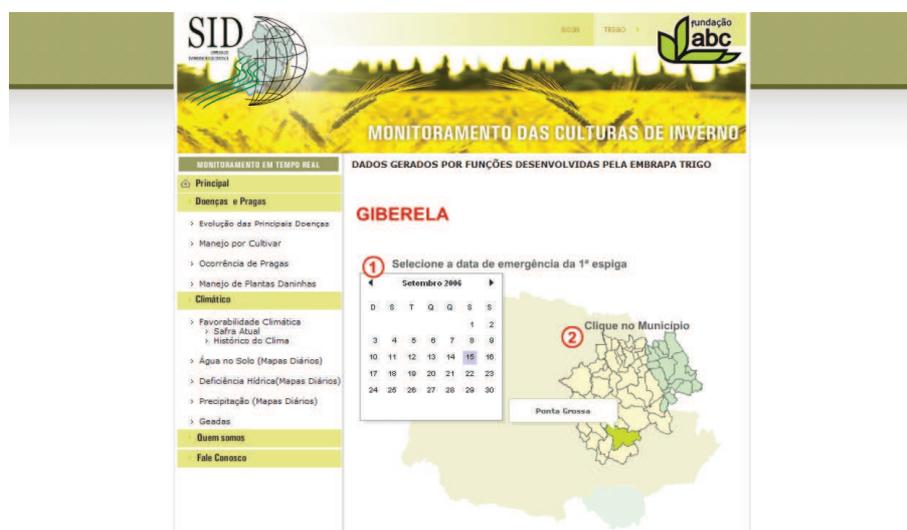


Figura 45: Sistema de auxílio à tomada de decisões: seleção de data de espigamento e município

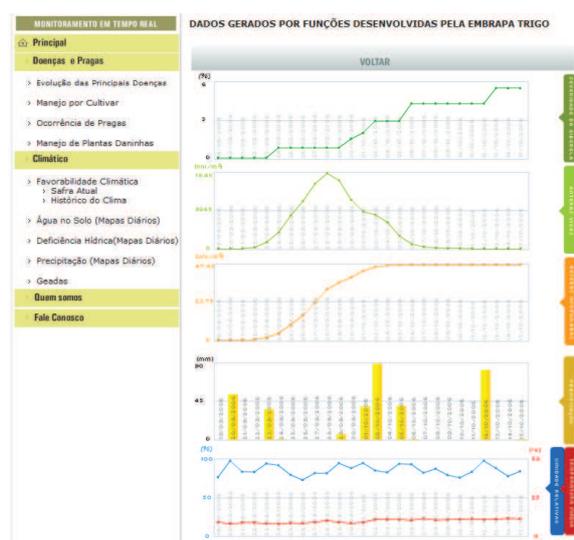


Figura 46: Sistema de auxílio à tomada de decisões: visualização dos dados simulados e climáticos em forma de gráficos

ciou com a leitura dos dados contidos no arquivo de configuração XML (*config.xml*), tanto para inicialização (Tabela 4) como para a doença em si. Com os parâmetros carregados, o sistema iniciou com

a instanciação⁵ das classes `Weather`, contendo dados meteorológicos e prognósticos climáticos, e `RInterface`, a qual realizou a integração com o simulador de culturas `CropSim-Wheat`.

Tabela 4: Parâmetros de configuração do SISALERT

Nome da variável	Descrição	Unidade	Valor exemplo
<code>amountHeadsByPlant</code>	Quantidade de espigas por planta	unidade	2
<code>amountPlants</code>	Quantidade de plantas por m ²	unidade	300
<code>useCropSim</code>	Simulação com interação com o <code>CropSim-Wheat</code>	booleano	“true”
<code>sowingDates</code>	Datas de semeadura	datas (Juliano)	“#173#180#”
<code>dss</code>	Dia de início da simulação (Calendário Juliano)	dia	143
<code>sendSMS</code>	Enviar mensagens SMS alertando sobre infecções	booleano	“false”
<code>connection</code>	URL de conexão para o banco de dados	URL	“jdbc:banco://host/base”
<code>driver</code>	Nome do driver utilizado para acesso ao banco de dados	descrição	“org.postgresql.Driver”
<code>user</code>	Nome do usuário no banco de dados	descrição	“sisalert”
<code>password</code>	Senha do usuários no banco de dados	descrição	“senha”

Após o processo de instanciação das classes `RInterface` e `Weather`, foram instanciadas as classes `Plant` e `Inoculum`. A classe `Plant` foi desenvolvida buscando-se oferecer uma dupla programação, com possibilidade de iniciar o processo de simulação a partir da data de espigamento ou conforme as informações sobre o seu estágio fenológico ($GSTD \geq 51$), fornecidas quando da execução concomitante do modelo `CropSim-Wheat`. Em ambos os processos o cálculo de taxas da

⁵Instanciação é o processo de criação de um objeto a partir de uma classe. Corresponde à alocação de memória através do ponto de vista da computação, para armazenar informações sobre um certo objeto (CORNELL; HORSTMANN, 1997; SRIDHARAN et al., 1997).

classe Plant deu-se apenas no período de espigamento. O objeto da classe Inóculo seguiu o mesmo procedimento, tendo o método `taxas` sido executado apenas quando do início do espigamento (Figura 47).

```

85 public void rate() {
86     double pHeadsTmp=0;
87     if(isActivatedCropSim() && cropsim.getGSTD() < 51)
88         return;
89     days++;
90     pHeadsTmp = 1 - Math.exp(-0.0127*Math.pow(days, 2.4352));
91     newHeads = 0;
92     if(pHeadsTmp > pHeads) {
93         newHeads = (int) ((pHeadsTmp-pHeads) * config.getAmountHeadsByPlant() * config.getAmountPlants());
94         pHeads = pHeadsTmp;
95     }
96     inoculum.rate();
97     for(Organ head: heads)
98         head.rate();
99 }

```

Figura 47: Método `rate` (taxas): verificação da ligação com o CropSim-Wheat e cálculo de novas espigas

No momento da inicialização (construtor⁶) do objeto da classe Plant, instanciou-se um objeto da classe DiseaseAccount, o qual é repassado como referência aos objetos da classe Organ (espigas) no momento da criação dos mesmos, que contabiliza todos os processos envolvidos na simulação da giberela. Para a criação das espigas utilizou-se a equação ajustada de Weibull, descrita por DelPonte et al. (2005)

$$pNewHeads = 1 - \exp(-0.0127t^{2.4352}) \quad (2.1)$$

onde t representa o tempo do processamento (dias) e $pNewHeads$, uma proporção de espigas emergidas, esta utilizada para calcular o número de novas espigas (Figura 47). O número de espigas calculado foi utilizado na inicialização de um novo *cohort*, representando-as (Figura 48).

⁶Construtor é um método que possuiu o mesmo nome da classe, sendo responsável pela inicialização dos atributos de uma classe.

```

101 public void integration() {
102     for(Organ head: heads)
103         head.integration();
104     inoculum.integration();
105     diseaseAccount.integration();
106     if(newHeads > 0)
107         heads.add(new Organ(newHeads, diseaseAccount));
108 }

```

Figura 48: Método *integration* (integração) de Plant: invoca a integração dos demais módulos e cria novo cohort de espigas

Para cada *cohort* de espigas (*Organ*) criado foram repassados o número de espigas que ele representava e a referência ao objeto da classe *DiseaseAccount* criado. Cada objeto, que representava um conjunto de espigas, foi responsável pelo cálculo da extrusão e remoção das anteras. A extrusão seguiu outra equação de Weibull, também descrita por DelPonte et al. (2005):

$$pNewAnthers = 1 - \exp(at^b) \quad (2.2)$$

onde $a = 0.255 - 0.029T + 0.0009T^2$; $b = -5.773 + 0.966T - 0.0278T^2$; t representa o tempo do processamento (dias); T , a temperatura média diária ($^{\circ}C$) e $pNewAnthers$, uma proporção de anteras emergidas naquele dia, sendo esta proporção utilizada na criação dos *cohorts* de anteras (Figura 49). A remoção dos *cohorts* de anteras ocorreu após estas completarem um período de dois dias com radiação solar acima de $10 \text{ MJ}/\text{m}^2/\text{dia}^1$, ou simplesmente um período de cinco dias (DELPONTE et al., 2005) (Figura 50).

A nuvem de esporos da doença foi calculada no método *taxas* da instância de *Inoculum* (Figura 51), invocado a cada passo do simulador pela instância de *Plant*, sendo esse valor transferido à instância de *Cloud*. Este cálculo deu-se pela equação descrita por DelPonte et al. (2005):

```

63 public void rate() {
64     double pAnthersTmp,a,b,tMean;
65     tMean = getWeather().getTMean();
66     days++;
67     a = 0.255 - 0.029 * tMean + 0.0009 * Math.pow(tMean, 2);
68     b = -5.773 + 0.966 * tMean - 0.0278 * Math.pow(tMean, 2);
69     pAnthersTmp = 1 - Math.exp(-a*Math.pow(days,b));
70     newAnthers = 0;
71     if(days > 3 && pAnthersTmp > pAnthers) {
72         newAnthers = (int) ((pAnthersTmp - pAnthers) * amountHeads * config.getAmountAnthersByHead());
73         pAnthers = pAnthersTmp;
74     }
75     for(Anther anther: anthers) {
76         anther.rate();
77     }
78 }

```

Figura 49: Método *rate* (taxas): cálculo de novas anteras

```

80 public void removeCohortAnthers(Anther a) {
81     if(a.getDay() > 2) {
82         if(getWeather().getSRad() >= 10 || a.getDay() > 5){
83             anthers.remove(a);
84         }
85     }
86 }

```

Figura 50: Método para remoção de um *cohort* de anteras

$$gz = (-0.6306 + 0.0152RH + 0.1076CRD)^2 \quad (2.3)$$

onde RH representa a umidade relativa média do dia e CRD , uma variável auxiliar que teve o seu valor definido conforme o número de dias consecutivos com precipitações maiores que $0.3mm$ (valores para 5 dias consecutivos: $CRD = 1;2;2.5;0.3;0.3$). O valor e a média no final do dia, ou seja, na chamada do método integração, foram repassados para a nuvem.

A velocidade da infecção (Figura 52) foi calculada como uma proporção por meio da equação

$$inf = 0.0010289exp(0.1957T) \quad (2.4)$$

onde T é a temperatura média das últimas 48 horas. Esta proporção

```

38 public void rate() {
39     double[] vectorCRD = {0,1,2,2.5,0.3,0.3};
40     double crd;
41     if(getWeather().getTotalDaysRain() <= 5)
42         crd = vectorCRD[getWeather().getTotalDaysRain()];
43     else
44         crd = 0;
45     gzMean = gz;
46     gz = Math.pow(-0.6306 + 0.0152 * getWeather().getUrMean() + 0.1076 * crd,2);
47     gzMean = (gzMean + gz)/2;
48 }
49
50 public void integration() {
51     cloud.update(gz,gzMean);
52     cloud.integration();
53 }

```

Figura 51: Métodos *rate* (taxas) e *integration* (integração) da classe Inóculo: cálculo da nuvem de esporos, atualização e integração

foi utilizada para calcular a proporção de anteras infectadas a cada dia, desde que uma regra simples de combinação de fatores de umidade relativa e precipitação fosse obedecida (Figura 53): (i) dois dias com precipitações $\geq 0.3mm$ (totalizados pela classe Weather) e umidade relativa média das últimas 48 horas $\geq 80\%$; ou (ii) precipitação no dia anterior $\geq 0.3mm$ e umidade relativa média no dia $\geq 85\%$; ou (iii) precipitação $\geq 0.3mm$ no dia corrente, com umidade relativa média do dia anterior $\geq 85\%$ (DELPONTE et al., 2005).

```

30 public static double temperatureFavorability(double t) {
31     return 0.0010289 * Math.exp(0.1957 * t);
32 }
33
34 }

```

Figura 52: Cálculo da velocidade da infecção baseada na temperatura (*inf*)

O processo da contabilização da infecção (Figura 53) ocorreu, a passos diários, pela equação

$$dailyGib = st * gzmean * inf \quad (2.5)$$

onde *st* é a proporção de anteras vivas no instante da simulação;

gzmean, a média da nuvem; *inf*, a suscetibilidade da infecção e *dailyGib*, a proporção de anteras infectadas. Uma regra especial foi implementada para o valor de *st* (Figura 54): (i) se o valor de *st* fosse menor que 0.25, houvesse anteras vivas e o valor fosse menor que o valor máximo atingido por ele (*stMax* - pico), era assumido o valor mínimo de 0.25; (ii) caso não existissem mais anteras e o valor de *st* fosse menor que *stMax*, era assumido o valor de 0.25, até sete dias após remoção de todas as anteras, 0.1 até 14 dias após a remoção das anteras e 0 para qualquer quantidade de dias superiores a 14.

```

88 public void integration() {
89     double inf = 0, st = 0;
90     if((getWeather().getTotalDaysRain() >= 2 && getWeather().getURMean48() >=80) ||
91        (getWeather().getRainYesterday() >= 0.3 && getWeather().getUrMean() >=85) ||
92        (getWeather().getRain() >= 0.3 && getWeather().getURMeanYesterday() >= 85)) {
93         inf = Utilities.temperatureFavorability(getWeather().getTMean48());
94     }
95     st = proportionAntherArea();
96     dailyGib = st * cloud.getGZMean() * inf;
97     diseaseAccount.update(dailyGib);
98     for(Anther anther: anthers) {
99         anther.integration();
100        removeCohortAnthers(anther);
101    }
102    if(newAnthers > 0)
103        anthers.add(new Anther(newAnthers));
104 }

```

Figura 53: Método integration (integração): cálculo da giberela diária (*dailyGib*)

```

106 public double proportionAntherArea() {
107     double st = (double) getAmountAnthersLive()/getTotalAnthers();
108     if(st > stMax) stMax = st;
109     if(st < stMax && getAmountAnthersLive() > 0 && st < 0.25)
110         st = 0.25;
111     else if(getAmountAnthersLive() <= 0 && st < stMax) {
112         daysAfter++;
113         if(daysAfter <= 7)
114             st = 0.25;
115         else if(daysAfter > 7 && daysAfter <=14)
116             st = 0.1;
117         else
118             st = 0;
119     }
120     return st;
121 }

```

Figura 54: Cálculo da proporção de anteras (*st*)

A severidade, por sua vez, foi calculada pela equação:

$$sev = 0.7442 + 1.46 * finalGib \quad (2.6)$$

onde *sev* é a severidade da doença naquele dia e *finalGib*, o somatório dos valores de giberela calculados diariamente e multiplicados por 100 (valor em porcentagem). A operação de cálculo da severidade (Figura 55) foi realizada na instância de *DiseaseAccount*, a cada passo do simulador, por meio do método *integração*. O valor da severidade para cada dia foi armazenado no banco de dados para posterior geração de relatórios e gráficos. Todo o ciclo da simulação se repetiu até que não existissem mais anteras extrusadas ou chance de infecção, trinta dias após a queda de todas as anteras (DELPONTE et al., 2005).

```
31 public void integration () {
32     finalGib = getFinalGib();
33     if(finalGib > 0)
34         severity = 0.7442 + 1.46 * finalGib;
35 }
```

Figura 55: Cálculo da severidade a cada rodada (*severity*)

Discussão

A proposta metodológica apresentada e implementada confirmou as expectativas de sucesso depositadas sobre o conjunto de tecnologias utilizadas. Tanto a linguagem de programação Java como a UML mostraram-se eficientes e facilitadoras no que diz respeito ao desenvolvimento, possibilitando o aumento da produtividade, da flexibilidade, reúso e qualidade dos softwares produzidos, permitindo que pessoas sem um profundo conhecimento de programação possam entender a estrutura e auxiliar no desenvolvimento (BECK et al., 2002; PAPA-JORGJI; PARDALOS, 2006). Além disso, as linguagens de programação orientadas a objetos, por meio de suas características inerentes,

são capazes de facilitar o desenvolvimento de módulos e componentes com características muito similares às encontradas no meio ambiente (ROBERTS; DESSOUKY, 1998).

As diversas formas de disponibilização de informações empregadas nesta proposta possibilitam uma grande abrangência da plataforma, estando à disposição dos interessados nos mais variados lugares. A flexibilidade oferecida pelas interfaces Web (FISHWICK, 1996; FRAISSE et al., 2006) conduz a que a tomada de decisão seja facilitada tendo em vista a praticidade e a rápida obtenção das respostas sobre o problema em foco. O uso de linguagens visuais como o Flash e Applets Java permite disponibilizar informações de qualidade, de forma simples e objetiva, sem a necessidade de grandes treinamentos (SRIDHARAN et al., 1997; UNDERDAHL, 2003; ALBANO et al., 2003).

A aplicação de padrões de projetos, como o MVC, na separação das camadas da aplicação permite a utilização de equipamentos portáteis, como PDAs (*Personal digital assistants*) e celulares, na camada de visão, sem a necessidade de alterar a camada de modelo ou de controle (VEIT; HERRMANN, 2003). Com esses equipamentos, é possível acompanhar os resultados das simulações, assim usar on-line o simulador ou realizar o agendamento de processamento dos modelos, estando presente em qualquer lugar ou horário, pois estes equipamentos são geralmente itens pessoais, deixando de ser um item de ficção e tornando-se uma ferramenta poderosa à disposição dos tomadores de decisão (OGLIARI et al., 2005, 2006). Além do uso de sistemas embarcados em tecnologias móveis, recursos mais simples da plataforma podem ser utilizados, como o envio de alertas e de resultados das simulações através de mensagens SMS (Short Message Service) ou e-mails.

A grande potencialidade do reaproveitamento do conhecimento legado deu-se pelo uso da linguagem de programação R (PETZOLDT, 2003; TIERNEY, 2003; R Development Core Team, 2006), por meio do uso de scripts que possibilitaram a criação de “uma via de mão dupla” entre os modelos existentes. O caso do CropSim-Wheat pode ser citado como exemplo, podendo ser estendido para uma grande quantidade de outros modelos, criando, quem sabe, num futuro próximo uma grande rede de modelos de simulação.

Tendo em vista o caráter modular aplicado no desenvolvimento do SISALERT, pode-se, por meio da parametrização do sistema, optar pela execução concomitante do modelo de crescimento e desenvolvimento do trigo (CropSim-Wheat), obtendo-se, dessa forma, diversas informações sobre o desenvolvimento da planta, como, por exemplo, o estágio fenológico (HUNT, 2007). No caso do estágio fenológico, pode ser utilizado para determinar o momento exato do início e fim do processo de espigamento (HUNT; PARARAJASINGHAM, 1995), um fator muito importante para a determinação do processo de infecção da giberela (DELPONTE et al., 2004, 2005). Com este acoplamento, estudos baseados em dados históricos e datas de semeadura podem ser realizados para melhor entender os fenômenos ocorridos ao longo dos anos (LIMA et al., 2002).

Em virtude da modularidade (REYNOLDS; ACOCK, 1997), o SISALERT pode ser adaptado e reutilizado, com grande facilidade, em outros sistemas de auxílio à tomada de decisões, tendo em vista a utilização de bancos de dados centralizados, esses facilitadores no compartilhamento de informações. Com isso, execuções individuais ou em lote (*batch*) podem ser realizadas, armazenando e recuperando um grande conjunto de resultados, auxiliando no entendimento de muitos fatores envolvidos numa epidemia.

A ferramenta desenvolvida mostrou-se metodologicamente eficiente e pode ser de grande utilidade para o auxílio à tomada de decisão. No caso específico da giberela do trigo, usada como demonstração neste estudo, o sistema foi utilizado com sucesso, provendo informação sobre o risco de ocorrência de epidemias de giberela (DELPONTE et al., 2005). O sistema foi implantado numa importante região tritícola do país, região dos Campos Gerais do Paraná, de cobertura da Fundação ABC, a qual foi a responsável pela aplicação e validação a campo dos resultados deste sistema, interagindo com os agricultores por meio de sua equipe técnica, a qual permitiu o grande envolvimento dos beneficiários no sistema. O trabalho de parceria entre a pesquisa e os agentes de campo mostrou-se como um fator indispensável para o sucesso da implantação de sistemas como este. Os resultados obtidos com a implantação do sistema refletiram a expectativa, obtendo-se valores muito próximos dos observados e mostrando-se como uma ferramenta de suma importância, sendo, inclusive, ampliada para a safra atual (2007), contando com um maior número de estações meteorológicas automáticas e abrangendo um maior número de municípios (Comunicação pessoal, Rodrigo Yoiti Tsukahara - Fundação ABC).

O uso da plataforma vai muito além do auxílio no controle da doença, podendo contribuir na melhoria da qualidade, na redução da presença de micotoxinas provocadas pela incidência de giberela (PANISSON et al., 2003), na redução dos custos de produção (DELPONTE et al., 2004), evitando a aplicação de produtos químicos em momentos desnecessários (PICININI; FERNANDES, 2001), e na garantia de uma melhor qualidade e uma conseqüente melhora na colocação dos produtos no mercado.

Com a instalação de uma malha de estações meteorológicas automáticas pelo INMET, outras regiões do país poderão contar com o

sistema, expandindo-se, dessa forma, a sua aplicação; apenas é necessária a sua instalação em servidores públicos ou de instituições, como Embrapa, cooperativas e universidades. Nesse sentido, uma outra experiência de sucesso foi realizada com a cooperativa Cotrijal de Não-Me-Toque/RS, na qual se contou com o acompanhamento do grupo de técnicos em propriedades da região, juntamente com a assessoria de pesquisadores da Embrapa-Trigo de Passo Fundo (FERNANDES et al., 2007, 2007b). Disponibilizou-se a estrutura com interfaces Web e envios de mensagens SMS quando da ocorrência de novos índices de severidade.

A utilização da plataforma SISALERT, em conjunto com prognósticos climáticos, apresenta um grande potencial no gerenciamento de doenças, graças à possibilidade de ligação com modelos de simulação do crescimento e desenvolvimento de culturas, podendo indicar, por exemplo, a necessidade da antecipação da semeadura ou uso de cultivares mais precoces, na tentativa de escapar do período de suscetibilidade da doença, embora podendo correr o risco de sofrer danos por geada (SCHEEREN et al., 2000) ou outra doença. Com isso, cenários podem ser montados e simulados a fim de determinar a melhor data de semeadura, tentando minimizar, dessa forma, os efeitos maléficos provocados pelo clima e/ou possíveis doenças.

Estudos sobre o impacto das mudanças climáticas sobre o cultivo do trigo (FERNANDES et al., 2004b) podem ser realizados com o auxílio dessa plataforma, sendo apenas necessária a alimentação do banco de dados com os prognósticos climáticos e as datas de semeadura ou espigamento. Além disso, com o uso desses prognósticos, várias rodadas podem ser realizadas, buscando-se encontrar as melhores datas de semeadura para determinadas regiões.

Tendo em vista a característica polimórfica da plataforma proposta (ROBERTS; DESSOUKY, 1998), esta foi transportada para outra cultura, visando organizar o conhecimento sobre a cultura da maçã sob sistemas dinâmicos computadorizados. Foi empregada para controle em pomares da região de Vacaria/RS, com uma base de dados climática centralizada, alimentada por sete estações meteorológicas automáticas. A coleta de dados dá-se em tempo quase real, em intervalos de 15 minutos, além de contar com prognósticos climáticos para sete dias, disponibilizados pelo INPE. Como forma de interação com o usuários criaram-se interfaces Web, disponibilizando os dados em mapas cicláveis relativos à área de abrangência de cada uma das estações meteorológicas e interfaces para equipamentos de pequeno porte, como Palms e celulares, utilizando a tecnologia Java Micro Edition. O sistema fica em execução permanente e, quando do risco de ocorrência do ataque de patógenos, os usuários são automaticamente informados por meio de mensagens enviadas para o telefones celulares (SMS) e correio eletrônico (PAVAN et al., 2006; SANHUEZA, 2007).

Além do enfoque dado, o auxílio à tomada de decisões, esta proposta metodológica pode ser utilizada para fins didáticos (FISHWICK, 1996; ALBANO et al., 2003; ANDRADE-PIEDRA et al., 2005), demonstrando pelo uso da simulação os possíveis efeitos provocados pelo clima, pelo manejo da doença, pelas diferentes datas de semeadura, pela suscetibilidade do hospedeiro, entre outros.

3 CONSIDERAÇÕES FINAIS

A aplicação de modernas técnicas de engenharia de software, como a UML, linguagens de programação orientadas a objetos e padrões de projeto, especialmente o MVC, mostrou-se apropriada e robusta para orientar o desenvolvimento de modelos de simulação bem projetados, reusáveis, compatíveis e facilmente acopláveis a outras soluções. Até recentemente, estas técnicas eram pouco utilizadas e, muitas vezes, desconhecidas na área agrícola.

O conjunto das tecnologias estudadas e utilizadas neste trabalho comprova a possibilidade do reaproveitamento do conhecimento legado, muito ativo e utilizado até os dias de hoje, mas com as limitações impostas pela evolução da computação. Linguagens de programação modernas, como o R, permitiram a comunicação entre Fortran e Java, por exemplo.

Com o desenvolvimento do modelo de simulação do patossistema *Puccinia*-Trigo foi possível acompanhar a evolução do crescimento da área foliar do trigo juntamente com o progresso da ferrugem da folha, mostrando-se como uma excelente alternativa para o estudo das relações patógeno/hospedeiro e a quantificação de danos.

A aplicação dos resultados obtidos no desenvolvimento da plataforma SISALERT, demonstrou a viabilidade da aplicação de sistemas de simulação para o auxílio aos tomadores de decisões. A plataforma demonstrou, ainda, a aplicabilidade da união de diversas tecnologias, como a captura e centralização de dados de estações meteorológi-

cas e prognósticos climáticos, além do processamento on-line ou em lote dos modelos. Esta plataforma pode ser expandida para outros usos/culturas.

REFERÊNCIAS BIBLIOGRÁFICAS

ABBAS, K. et al. Computational epidemiology: Bayesian disease surveillance. In: *Proc. of the International Conference on Bioinformatics and its Applications*. [S.l.: s.n.], 2004.

ABRAHAMSEN, P.; HANSEN, S. Daisy: an open soil-crop-atmosphere system model. *Environmental Modelling and Software*, v. 15, n. 2-3, p. 313–330, 2000.

ACOCK, B.; TRENT, A. *The soybean crop simulator, GLYCIM: documentation for the modular version 91*. Moscow, Idaho: University of Idaho, 1991. 242 p.

ADAMSON, C. *What Is Java*. 2006. Disponível em: <http://www.onjava.com/pub/a/onjava/2006/03/08/what-is-java.html?page=1>. Acesso em 13/04/2007.

ALBANO, G. et al. Web based simulations for virtual scientific experiment: Methodology and tools. In: *1st International ELeGI Conference on Advanced Technology for Enhanced Learning*. [S.l.: s.n.], 2003.

ALVES, K. J. P. *Influência da Temperatura e Umidade na Esporulação de Magnaporthe grisea (Dissertação de Mestrado)*. Passo Fundo, RS, Brasil: Universidade de Passo Fundo (UPF), 2005.

ALVES, K. J. P.; FERNANDES, J. M. C. Influência da temperatura e da umidade relativa do ar na esporulação de Magnaporthe grisea em trigo. *Fitopatologia Brasileira*, scielo, v. 31, p. 579–584, 12 2006. ISSN 0100-4158. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-415820060006000007&nrm=iso>.

AMORIN, L. et al. A simulation model to describe epidemics of rust of phaseolus beans. ii : Validation. *Phytopathology*, v. 85, n. 6, p. 722–727, 1995. ISSN 0031-949X.

ANDRADE-PIEDRA, J. L. et al. Qualification of a plant disease simulation model: Performance of the lateblight

model across a broad range of environments. *Phytopathology*, v. 95, n. 12, p. 1412–1422, 2005. Disponível em: <<http://apsjournals.apsnet.org/doi/abs/10.1094/PHYTO-95-1412>>.

ANGRA, D. *Transferência da resistência à ferrugem da folha através de cruzamentos intergenéricos entre Triticum aestivum e Agropyron elongatum (Dissertação de Mestrado)*. Pelotas, RS, Brasil: Universidade Federal de Pelotas (UFPEL), 1995. 68 p.

ARIS, V. *Use of Weather-based Modeling for Disease Management of Early Leaf Spot of Peanut and Glume Blotch of Wheat (Dissertação de Mestrado)*. 2000. 105 p. Disponível em: <http://www.lib.ncsu.edu/theses/available/etd-19991104-074954/>. Acesso em 28/05/2007.

AUDSLEY; MILNE; PAVELEY. A foliar disease model for use in wheat disease management decision support systems. *Annals of Applied Biology*, v. 147, p. 161–172(12), 2005. Disponível em: <<http://www.ingentaconnect.com/content/bsc/aab/2005/00000147/00000002/art00005>>.

AZEVEDO, S.; MICHEREFF, S.; MARIANO, R. Epidemiologia comparativa da podridão negra e da alternariose do repolho no agreste de pernambuco. *Fitopatologia brasileira*, v. 27, n. 1, p. 17–26, 2002. ISSN 0100-4158. Disponível em: <http://www.scielo.br/scielo.php>. Acesso em 05/12/2006.

BANNAYAN, M.; CROUT, N. M. J.; HOOGENBOOM, G. Application of the cereals-wheat model for within-season prediction of winter wheat yield in the united kingdom. *Agron J*, v. 95, n. 1, p. 114–125, 2003. Disponível em: <agron.sci-journals.org/cgi/content/abstract/agrojnl;95/1/114>.

BARROS, B. d. C.; CASTRO, J. L. d.; PATRÍCIO, F. R. A. Resposta de cultivares de trigo (*Triticum aestivum* L.) ao controle químico das principais doenças fúngicas da cultura. *Summa Phytopathologica*, sciELO, v. 32, p. 239–246, 09 2006. ISSN 0100-5405. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-54052006000300005&nrm=iso>.

BASTIAANS, L. Ratio between virtual and visual lesion size as a measure to describe reduction in leaf photosynthesis of rice due to leaf blast. *Phytopathology*, v. 81, p. 611–615, 1991.

BASTIAANS, L. *Understanding yield reduction in rice due to leaf blast (Ph.D. Thesis)*. Wageningen, The Netherlands: Wageningen Agricultural University, 1993.

BAUER, C.; KING, G. *Java Persistence With Hibernate*. [S.l.]: Manning, 2005. ISBN 1-932394-88-5.

BAYLES, R. Poorly filled grain in the cereal crop 1. the assessment of poor grain filling. *Natl. Inst. Agric. Bot.*, v. 14, p. 232–240, 1977.

BECK, H. et al. Object-oriented approach to crop modeling: Concepts and issues. *Proceedings of the World Congress of Computers in Agriculture and Natural Resources*, ASAE, Iguacu Falls, Brazil, p. 297–305, 2002.

BELL, G. *The history of wheat cultivation*. In: LUPTON, F.G.H. London: [s.n.], 1987. 31-50 p.

BERGAMIN FILHO, A.; AMORIM, L. *Doenças de plantas tropicais: epidemiologia e controle econômico*. [S.l.]: Editora Agronômica Ceres Ltda, 1996.

BERGAMIN FILHO, A.; AMORIM, L. Epidemiologia comparativa entre os patossistemas temperado e tropical: conseqüências para a resistência a fungicidas. *Fitopatologia Brasileira*, scielo, v. 26, p. 119–127, 06 2001. ISSN 0100-4158. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-41582001000200001&nrm=iso>.

BERGAMIN FILHO, A.; AMORIM, L.; KIMATI, H. *Manual de Fitopatologia*. [S.l.]: Agronômica Ceres, 1995. ISBN 85-318-0006-4.

BERGAMIN FILHO, A. et al. Avaliação de danos causados por doenças de plantas. *Revisão Anual de Patologia de Plantas*, v. 3, p. 133–184, 1995. ISSN 0104-0383.

BERGER, R. Helminthosporium turcicum lesion numbers related to numbers of trapped spores and fungicide sprays. *Phytopathology*, v. 63, p. 930–933, 1973. ISSN 0031-949X.

BERGER, R.; BERGAMIN FILHO, A.; AMORIM, L. Lesion expansion as an epidemic component. *Phytopathology*, v. 87, n. 10, p. 1005–1013, 1997. Disponível em: <<http://apsjournals.apsnet.org/doi/abs/10.1094/PHYTO.1997.87.10.1005>>.

BERGER, R. et al. A simulation model to describe epidemics of rust of phaseolus beans. i. development of the model and sensitivity analysis. *Phytopathology*, The American Phytopathological Society, v. 85, p. 715–721, 1995.

BERLATO, M. A.; FARENZENA, H.; FONTANA, D. C. Associação entre el niño oscilação sul e a produtividade do milho no estado do rio grande do sul. *Pesquisa Agropecuária Brasileira*, scielo, v. 40, p. 423–432, 05 2005. ISSN 0100-204X.

BERLATO, M. A.; FONTANA, D. C. *El Niño e La Niña; Impactos no clima, na vegetação e na agricultura do Rio Grande do Sul. Aplicações de previsões climáticas na agricultura*. [S.l.]: Editora da Universidade Federal do Rio Grande do Sul(EDURGS), 2003. 110 p. ISBN 85-7025-687-6.

BOOTE, K. et al. Coupling pests to crop growth simulators to predict yield reductions. *Phytopathology*, v. 73, p. 1581–1587, 1983.

BOWEN, K. L. *Analytical Models of Disease Progression (Capítulo 4)*. In: *Exercises in Plant Disease Epidemiology*. [S.l.]: APS Press, 1997. ISBN 0-89054-224-4.

BRAGA, R.; PINTO, P.; COELHO, J. *Os Modelos de Simulação e a Investigação de Sistemas de Agricultura*. 2005. Disponível em: http://www.esaelvas.pt/ricardo_braga/ordem.html. Acesso em 26/05/2005.

BRUM, A.; HECK, C. A economia do trigo no rio grande do sul: Breve histórico do cereal na economia do estado. *Análise*, Porto Alegre, RS, Brasil, v. 16, n. 1, p. 29–44, 2005.

CAMPBELL, C.; MADDEN, L. V. *Introduction to Plant Disease Epidemiology*. [S.l.]: John Wiley & Sons, 1990. ISBN 0-471-83236-7.

CASA, R. T. et al. Danos causados pela infecção de *Gibberella zeae* em trigo. *Fitopatologia Brasileira*, scielo, v. 29,

p. 289–293, 06 2004. ISSN 0100-4158. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-41582004000300008&nrm=iso>.

CENTURY. *General Model Description*. 2007. Disponível em: <http://www.nrel.colostate.edu/projects/century/>. Acesso em 25/05/2007.

CHARLES-EDWARDS, D. A. *Physiological determinants of crop growth*. [S.l.]: London: Academic Press, 1982. 161 p.

CHAVES, R. R.; RODRIGUEZ, D. A. *Apostila de Grads*. 2007. Disponível em: http://br.geocities.com/daisuke_oyama/fonte/grads.pdf. Acesso em 14/04/2007.

CHWIF, L. *Redução de Modelos de Simulação de Eventos Discretos na Sua Concepção: Uma Abordagem Causal (Tese de doutorado)*. 1999. Disponível em: http://www.livrosimulacao.eng.br/download/tese_chwif.pdf. Acesso em 21/05/2007.

CONVERSE, T. *PHP 4: a bíblia*. [S.l.]: Editora Campus, Rio de Janeiro, 2003. 868 p. ISBN 8535211306.

CORAZZA, D. *Utilizando Design Pattern MVC no Desenvolvimento de Aplicações Web (Monografia de Trabalho de Conclusão de Curso)*. 2004. Disponível em: <http://inf.upf.br:8080/bibdig/salvarArquivo.jsp?chave=59>. Acesso em: 19/03/2006.

CORNELL, G.; HORSTMANN, C. S. *Core Java (2nd ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997. ISBN 0-13-596891-7.

COX, H. *Users' guide to WhopperCropper*. 2006. Disponível em: <http://www.apsru.gov.au/apsru/>. Acesso em: 17/04/2007.

CPTEC, C. d. P. d. T. e. E. C. *Portal Previsões Numéricas*. 2007. Disponível em: http://www.cptec.inpe.br/prevnum/exp_eta.shtml. Acesso em: 11/05/2007.

DELPONTE, E. et al. Giberela do trigo: aspectos epidemiológicos e modelos de previsão. *Fitopatologia Brasileira*, scielo,

v. 29, p. 587–605, 12 2004. ISSN 0100-4158. Disponível em: <<http://www.scielo.br/scielo.php>>.

DELPONTE, E. M.; FERNANDES, J. M. C.; PAVAN, W. A risk infection simulation model for fusarium head blight of wheat. *Fitopatologia Brasileira*, scielo, v. 30, p. 634–642, 12 2005. ISSN 0100-4158. Disponível em: <<http://www.scielo.br/scielo.php>>.

DELPONTE, E. M. et al. Models and applications for risk assessment and prediction of asian soybean rust epidemics. *Fitopatologia Brasileira*, scielo, v. 31, p. 533–544, 12 2006. ISSN 0100-4158.

DORNELES, C. F. et al. Measuring similarity between collection of values. In: *WIDM '04: Proceedings of the 6th annual ACM international workshop on Web information and data management*. New York, NY, USA: ACM Press, 2004. p. 56–63. ISBN 1-58113-978-0.

DOUGLAS, K.; DOUGLAS, S. *PostgreSQL (2nd Edition) (Developer's Library)*. [S.l.]: Pearson, 2005. 1032 p. ISBN 0672327562.

ECKHARDT, H.; STEUBING, L.; KRANZ, J. Das koloniewachstum von *erysiphe graminis* d.c. f.sp. *hordei marchal* in abh angigkeit von temperatur and insertionshole der blatter. *Angewandte Botanik*, v. 58, p. 433–443, 1984.

ELINGS, A.; ROSSING, W. A. H.; WERF, W. van der. Virtual lesion extension: A measure to quantify the effects of bacterial blight on rice leaf co₂ exchange. *Analytical and Theoretical Plant Pathology*, The American Phytopathological Society, v. 89, n. 9, p. 789–795, 1999.

EMGE, R.; KINGSOLVER, C.; JOHNSON, D. Growth of sporulating zone of *puccinia striiformis* and its relationship to stripe rust epiphytology. *Phytopathology*, v. 65, p. 679–681, 1975. ISSN 0031-949X.

EVERTS, K.; LACY, M. Factors influencing infection of union leaves by *alternaria porri* and subsequent lesion expansion. *Plant Disease*, v. 80, p. 276–280, 1996. ISSN 0191-2917.

FARIA, R. T. d.; BOWEN, W. T. Evaluation of DSSAT soil-water balance module under cropped and bare soil conditions. *Brazilian Archives of Biology and Technology*, scielo, v. 46, p. 489–498, 12 2003. ISSN 1516-8913. Disponível em:

<http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1516-89132003000400001&nrm=iso>.

FERNANDES, J. et al. Modelling fusarium head blight in wheat under climate change using linked process-based models. in: Canty, s.m. and boring, t. and wardwell, j. and ward, r.w. (eds.). In: *2nd International Symposium on Fusarium Head Blight; incorporating the 8th European Fusarium Seminar*. Orlando, FL, USA: Michigan State University, East Lansing, MI, 2004. p. 441–444.

FERNANDES, J. et al. *Web-Based System to True-Forecast Disease Epidemics - Case Study for Fusarium Head Blight of Wheat*. In: Sivakumar, Mannava V. K. and Hansen, James (Eds.). <http://www.springerlink.com/content/g7326w65km36u4h7>: Springer Berlin Heidelberg, 2007. 265-271 p. ISBN 978-3-540-44649-1.

FERNANDES, J. et al. *Web-Based System to True-Forecast Disease Epidemics* In: Buck, H. T., Nisi, J. E. and Salomón, N. (Eds.). <http://www.springerlink.com/content/m20221577m052311>: Springer Netherlands, 2007b. 259-264 p. ISBN 978-1-4020-5496-9.

FERNANDES, J. M. C. et al. Modeling fusarium head blight in wheat under change using linked process-based models. In: *Proceedings of the 2nd International Symposium on Fusarium Head Blight*. Orlando, FL, USA: Michigan State University, East Lansing, 2004b. p. 441–443.

FERNANDES, J. M. C. et al. Simulação de epidemias. *Revisão Anual de Patologia de Plantas*, Passo Fundo: RAP, Passo Fundo, Brasil, v. 2, n. 1, 1994. ISSN 0104-0383.

FERNANDES, J. M. C. O. *Doenças em sistema de plantio direto com ênfase em mancha foliar e giberela*. [S.l.]: Embrapa Trigo, 2000.

FISHWICK, P. A. Web-based simulation: some personal observations. In: *WSC '96: Proceedings of the 28th conference on Winter simulation*. [S.l.: s.n.], 1996. p. 772–779. ISBN 0-7803-3383-7.

FRAISSE, C. et al. Agclimate: A climate forecast information system for agricultural risk management in the southeastern usa. *Computers and Electronics in Agriculture*, Elsevier B.V., <http://www.sciencedirect.com/science/article/B6T5M-4JMB0KB-1/2/608a1626d8a4c92dfc6f83ce85480992>, v. 53, n. 1, p. 13–27, 2006.

FUNDATION, A. S. *Struts*. 2007. Disponível em: <http://struts.apache.org/>. Acesso em 01/04/2007.

GILLIGAN, C. A. *Construction of temporal models: III. Disease progress of soil-borne pathogens*. In *Mathematical modelling of crop disease*. [S.l.]: New York: Academic, 1985. 66-108 p.

GILLIGAN, C. A. Comparison of disease progress curves. *New Phytologist*, v. 115, n. 2, p. 223–242, 1990.

GLEASON, M. L. et al. Development and validation of an empirical model to estimate the duration of dew periods. *Plant Disease*, v. 78, p. 1011–1016., 1994. Disponível em: <http://www.apsnet.org/pd/SEARCH/1994/PD_78_1011.asp>.

GOODING, M. et al. Green leaf area decline of wheat flag leaves: the influence of fungicides and relationships with mean grain weight and grain yield. *Annals of Applied Biology*, v. 136, n. 1, p. 77–84, 2000. ISSN 0003-4746. Disponível em: <<http://www.ingentaconnect.com/content/aab/annals/2000/00000136/00000001/art99110>>.

GRAVES, A. et al. Crop simulation models as tools in computer laboratory and classroom-based education. *Journal of Natural Resources and Life Sciences Education*, v. 31, p. 48–54, 2002.

GUARNIERI, R.; PEREIRA, E.; CHOU, S. Improving solar radiation forecasts from Eta/CPTEC model using statistical post-processing. In: . [S.l.: s.n.], 2006. (COSPAR, Plenary Meeting, v. 36), p. 2537.

GäUMANN, E. *Principles of Plant Infection: A Text-book of General Plant Pathology for Biologists, Agriculturists*. [S.l.]: Lockwood, London, 1950.

HANSEN, S. *Daisy, a flexible Soil-Plant-Atmosphere system Model*. 2002. Disponível em: <http://www.dina.kvl.dk/~daisy/ftp/DaisyDescription.doc>. Acesso em 26/11/2006.

HAYMAN, P.; EASDOWN, W. An ecology of a dss: reflections on managing wheat crops in the northeastern australian grains region with wheatman. *Agricultural Systems*, v. 74, p. 57–77, 2002. Disponível em:

<<http://www.ingentaconnect.com/content/els/0308521x/2002/00000074/00000001/art00018>>.

HILLSIDE. *Patterns Library*. 2007. Disponível em: <http://hillside.net/patterns>. Acesso em: 21/03/2007.

HOOD, M.; SHEW, H. Pathogenesis of thielaviopsis basicola on a susceptible and a resistant cultivar of burley tobacco. *Phytopathology*, v. 86, p. 38–44, 1996. ISSN 0031-949X.

HOOGENBOOM, G. Contribution of agrometeorology to the simulation of crop production and its applications. *Agricultural and Forest Meteorology*, v. 103, n. 1-2, p. 137–157, 2000.

HUNT, L. *Modelo de Simulação Cropsim. Comunicação pessoal*. 2007. Disponível em: thunt@uoguelph.ca. Acesso em: Jan/2007.

HUNT, L. A.; PARARAJASINGHAM, S. Cropsim-wheat: A model describing the growth and development of wheat. *Canadian Journal Plant Science*, Ottawa, Canada, v. 75, p. 612–632, 1995.

ICASA. *ICASA - International Consortium for Agricultural Systems Applications*. 2007. Disponível em: <http://icasa.net>. Acesso em 14/04/2007.

JABRZEMSKI, R.; SUTHERLAND, A. An innovative approach to weather-based decision-support for agricultural models. In: *22nd International Conference on Interactive Information Processing Systems for Meteorology, Oceanography, and Hydrology*. Washington DC, USA: American Meteorological Society, 2006. Disponível em: http://ams.confex.com/ams/Annual2006/techprogram/paper_101626.htm. Acesso em 26/06/2007.

JOHNSON, K. Evaluation of a mechanistic model that describes potato crop losses caused by multiple pests. *Phytopathology*, v. 82, n. 3, p. 363–369, 1992. ISSN 0031-949X.

JONES, J.; RITCHIE, J. *Crop Growth Models*. [S.l.]: American Society of Agricultural Engineers, 1990. 63-89 p.

JONES, J. W. et al. The dssat cropping system model. *European Journal of Agronomy: Modelling Cropping Systems: Science, Software and Applications*, v. 18, n. 3-4, p. 235–265, 2003.

JONES, J. W.; KEATING, B. A.; PORTER, C. H. Approaches to modular model development. *Agricultural Systems*, Elsevier Science Ltd, v. 70, n. 2-3, p. 421–443, 2001. ISSN 0308-521X.

KATO, H.; KOZAKA, T. Effect of temperature on lesion enlargement and sporulation of *pyricularia oryzae* in rice leaves. *Phytopathology*, v. 64, p. 828–830, 1974. ISSN 0031-949X.

KEATING, B. et al. An overview of apsim, a model designed for farming systems simulation. *Europeanjournal of Agronomy*, v. 18, p. 267–288(22), 2003. Disponível em: <<http://www.ingentaconnect.com/content/els/11610301/2003/00000018/00000003/art00108>>.

KEEN, R.; SPAIN, J. *Computer simulation in biology: A BASIC introduction*. New York, USA: Wiley-Liss, 1992. 498 p.

KIM, K.; TAYLOR, S.; GLEASON, M. Development and validation of a leaf wetness duration model using a fuzzy logic system. *Agricultural and Forest Meteorology*, v. 127, n. 1-2, p. 53–64, 2004. Disponível em: <<http://www.sciencedirect.com/science/article/B6V8W-4DTJS5R2/2/98a57608c713190a1bdb6a1a46cda560>>.

KIM, K. et al. Estimation of leaf wetness duration using empirical models in northwestern costa rica. *Agricultural and Forest Meteorology*, v. 129, n. 1-2, p. 53–67, 2005. Disponível em: <<http://www.sciencedirect.com/science/article/B6V8W-4FFNBP0-1/2/e2f22f21e950310c03cb123f8235cc46>>.

KOCH, M.; MEW, T. Rate of lesion expansion in leaves as a parameter of resistance to *xanthomonas campestris* pv. *oryzae* in rice. *Plant Disease*, v. 75, p. 897–900, 1991. ISSN 0191-2917.

KOGAN, M. Integrated pest management: historical perspectives and contemporary developments. *Annual Review of Entomology*, v. 43, n. 1, p. 243–270, 1998. Disponível em: <<http://arjournals.annualreviews.org/doi/abs/10.1146/annurev.ento.43.1.243>>.

LATORRE, M. R. D. O.; CARDOSO, M. R. A. Análise de séries temporais em epidemiologia: uma introdução sobre os aspectos metodológicos. *Rev. bras. epidemiologia*, v. 4, n. 3, p. 145–152, 2001. ISSN 1415-790X.

LEAL, S. *Modelação Matemática uma Proposta Metodológica para o Curso de Economia (Dissertação de mestrado)*. Florianópolis, SC, Brasil: [s.n.], 1999. Disponível em: <http://www.eps.ufsc.br/disserta99/leal>. Acesso em 30/11/2006.

LEITE, L. F. C.; MENDONÇA, E. d. S. Modelo century de dinâmica da matéria orgânica do solo: equações e pressupostos. *Ciência Rural*, sciELO, v. 33, p. 679–686, 08 2003. ISSN 0103-8478.

LEITE, L. F. C.; MENDONÇA, E. S.; MACHADO, P. L. O. A. Simulação pelo modelo century da dinâmica da matéria orgânica de um argissolo sob adubação mineral e orgânica. *Revista Brasileira de Ciência do Solo*, sciELO, v. 28, p. 347–358, 04 2004. ISSN 0100-0683.

LEITE, R. M.; AMORIM, L. Elaboração e validação de escala diagramática para mancha de alternaria em girassol. *Summa Phytopathologica*, v. 28, n. 1, p. 14–19, 2002.

LIMA, M. I. P. M.; FERNANDES, J. M.; PICININI, E. C. Controle cultural de giberela em trigo. In: *Resumos do XXXV Congresso Brasileiro de Fitopatologia*. [S.l.: s.n.], 2002. v. 27, p. 125–125.

LINDENTHAL, M. et al. Effect of downy mildew development on transpiration of cucumber leaves visualized by digital infrared thermography. *Phytopathology*, v. 95, n. 3, p. 233–240, 2005. Disponível em: <<http://apsjournals.apsnet.org/doi/abs/10.1094/PHYTO-95-0233>>.

MAANEN, A. v.; XU, X. Modelling plant disease epidemics. *Europeanjournal of Plant Pathology*, The American Phytopathological Society, v. 109, n. 7, p. 669–682, 2003. ISSN 0929-1873.

MAANEN, A. van; XU, X. M. Modelling plant disease epidemics. *Europeanjournal of Plant Pathology*, Springer, v. 109, n. 7, p. 669–682, 2003. ISSN 0100-4158.

MAGAREY, R. et al. Nappfast: An internet system for the weather-based mapping of plant pathogens. *Plant Disease*, The American Phytopathological Society, v. 91, n. 4, p. 281–299, 2007. ISSN 91-4-0336.

MAGAREY, R. D.; SUTTON, T. B.; THAYER, C. L. A simple generic infection model for foliar fungal plant patho-

gens. *Phytopathology*, v. 95, n. 1, p. 92–100, 2005. Disponível em: <<http://apsjournals.apsnet.org/doi/abs/10.1094/PHYTO-95-0092>>.

MARTINS, M. et al. Escala diagramática para a quantificação do complexo de doenças foliares de final de ciclo em soja. *Fitopatologia Brasileira*, v. 29, n. 2, p. 179–184, 2004.

MAVROMATIS, T. et al. Developing Genetic Coefficients for Crop Simulation Models with Data from Crop Performance Trials. *Crop Sci*, v. 41, n. 1, p. 40–51, 2001. Disponível em: <<http://crop.scijournal.org/cgi/content/abstract/cropsci;41/1/40>>.

MAY, C. L. et al. Improvement and validation of a pea crop growth model to simulate the growth of cultivars infected with ascochyta blight (*mycosphaerella pinodes*). *Europeanjournal of Plant Pathology*, v. 112, n. 1, p. 1–12, 2005. Disponível em: <<http://www.springerlink.com/content/g0327x861k5t4248>>.

MCCLAY, A. *CLIMEX Models to Predict the Potential Naturalised Range of the European Bumblebee *Bombus Terrestris* (L.) in Australia (Report)*. 2005. Disponível em: <http://www.environment.gov.au/biodiversity/trade-use/invitecomment/pubs/bombus-terrestris-appendices.pdf>. Acesso em 28/05/2007.

MEDEIROS, H. R. *Avaliação de modelos matemáticos desenvolvidos para auxiliar a tomada de decisão em sistemas de produção de ruminantes em pastagens (Tese de doutorado)*. Piracicaba, SP, Brasil: [s.n.], 2003. 112 p. Disponível em: <http://www.teses.usp.br/teses/disponiveis/11/11139/tde-27112003-100019/publico/henrique.pdf>. Acesso em 25/11/2006.

MEHTA, Y. Produção de conídios, período de esporulação e extensão da lesão por *helminthosporium sativum* nas folhas-bandeiras de trigo. *Pesquisa Agropecuária Brasileira*, v. 16, p. 77–79, 1981. ISSN 0100-204X.

MEHTA, Y. R.; ZADOKS, J. C. Uredospore production and sporulation period of *puccinia recondita* f. sp. *tritricina* on primary leaves of wheat. *Journal of Plant Pathology*, v. 16, p. 267–276, 1970.

MICHEREFF, S. *EPIDEMIOLOGIA DE DOENÇAS DE PLANTAS*. 2006. Disponível em: <http://www.ufrpe.br/fitopatologia/teoricas/T13.pdf>. Acesso em 10/11/2006.

MICROSYSTEMS, S. *Designing Enterprise Applications with the J2EE Platform, Second Edition*. 2002. Disponível em: http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e. Acesso em 15/04/2007.

MICROSYSTEMS, S. *The Java Technology Phenomenon*. 2007. Disponível em: <http://java.sun.com/docs/books/tutorial/getStarted/intro/definition.html>. Acesso em 13/04/2007.

MIKLER, A. R.; VENKATACHALAM, S.; ABBAS, K. Modeling infectious diseases using global stochastic cellular automata. *Journal of Biological Systems*, v. 13, n. 4, p. 421–439, 2005. Disponível em: <<http://ejournals.worldscientific.com.sg/129/13/1304/S0218339005001604.html>>.

MORAES, R. et al. Dosagem de ergosterol como indicador de contaminação fúngica em milho armazenado. *Arq. Inst. Biol.*, v. 70, n. 4, p. 483–489, 2003. Disponível em: <http://www.biologico.sp.gov.br/ARQUIVOS/V70_4/moraes.pdf>.

NICHOLSON, R. et al. *Adhesion of fungi to the plant surface: prerequisite for pathogenesis*. In: Cole, G.T. and Hoch, H.C. [S.l.]: Plenum Press, New York, 1991. 3-24 p. ISBN 0306434547.

NOAA, N. O. A. A. *HYbrid Single-Particle Lagrangian Integrated Trajectory Model*. 2007. Disponível em: http://www.arl.noaa.gov/ready/hysp_info.html. Acesso em 28/05/2007.

NORWOOD, F. B.; ROBERTS, M. C.; LUSK, J. L. Reply: Ranking crop yield models. *American Journal of Agricultural Economics*, v. 88, n. 4, p. 1111–1112, 2006. Disponível em: <<http://www.blackwell-synergy.com/doi/abs/10.1111/j.1467-8276.2006.00920.x>>.

NUTTER, F. W.; ESKER, P. D.; COELHO, R. A. Disease assessment concepts and the advancements made in improving the accuracy and precision of plant disease data.

European journal of Plant Pathology, Springer Netherlands, <http://www.springerlink.com/content/850114u838304484>, v. 115, n. 1, p. 95–103, 2006. ISSN 0929-1873.

NYGAARD, K.; DAHL, O.-J. The development of the simulalanguages. ACM Press, New York, NY, USA, p. 439–480, 1981.

OGLIARI, R. et al. Migrando sistemas legados de tomada de decisão para ambientes móveis. In: *V Simpósio de Informática da Região Centro do RS*. Santa Maria, Brasil: [s.n.], 2006.

OGLIARI, R. d. S. et al. Agricultura, computação móvel e sistemas baseados em localização. In: *VI Simpósio de Informática do Planalto Médio*. Passo Fundo, Brasil: UPF, 2005.

OSÓRIO, E. A. et al. Correlação de suscetibilidade à giberela com caracteres da planta de trigo. *Rev. Bras. de Agrociência*, <http://www.ufpel.tche.br/faem/agrociencia/v4n2/v4n29.htm>, v. 4, n. 2, p. 111–114, 1998. ISSN 0104-8996.

PANISSON, E.; REIS, E. M.; BOLLER, W. Quantificação de danos causados pela giberela em cereais de inverno, na safra 2000, em Passo Fundo, RS. *Fitopatologia Brasileira*, scielo, v. 28, p. 189–192, 04 2003. ISSN 0100-4158. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-41582003000200013&nrm=iso>.

PAPAJORGI, P. J.; PARDALOS, P. M. *Software Engineering Techniques Applied to Agricultural System - An Object-Oriented and UML Approach*. [S.l.]: Springer, 2006. ISBN 0-387-28170-3.

PARTON, W. J. et al. Analysis of factors controlling soil organic matter levels in great plains grasslands. *Soil Science Society of America journal*, Soil Science Society of America, Madison, v. 51, n. 5, p. 1173–1179, 1987. ISSN 0361-5995.

PAVAN, W. et al. Web-based system to true-forecast disease epidemics - sisalert. In: *Proceedings of Computers in Agriculture and Natural Resources, 4th World Congress Conference*. [S.l.]: American Society of Agricultural and Biological Engineers, 2006. ISBN 701P0606.

PETZOLDT, T. R as a simulation platform in ecological modeling. *R News*, v. 3, n. 3, p. 8–16, December 2003. Disponível em: <<http://CRAN.R-project.org/doc/Rnews/>>.

PICININI, E. Controle das doenças da parte aérea do trigo (*triticum aestivum*) com fungicidas em 1989. In: *23º Congresso Brasileiro de Fitopatologia*. Goiânia, GO, Brasil: Sociedade Brasileira de Fitopatologia, 1990. v. 15, p. 134–134.

PICININI, E.; FERNANDES, J. *Doenças em cereais de inverno: Aspectos epidemiológicos e controle*. [S.l.]: Embrapa Trigo, 1995. 59 p.

PICININI, E.; FERNANDES, J. *Efeito da época de pulverização com fungicidas sobre o controle de Gibberella zeae em trigo. Comunicado Técnico*. [S.l.]: Embrapa Trigo, 2001.

PICININI, E. et al. Impacto econômico do uso do fungicida propiconazole na cultura do trigo (*triticum aestivum*). *Fitopatologia Brasileira*, Sociedade Brasileira de Fitopatologia, v. 21, p. 326–368, 10 1996. ISSN 0100-4158.

PICININI, E. C.; FERNANDES, J. M. C. Efeito do tratamento de sementes com fungicida sobre o controle de doenças na parte aérea do trigo. *Fitopatologia Brasileira*, scielo, v. 28, p. 515–520, 10 2003. ISSN 0100-4158. Disponível em: <<http://www.scielo.br/scielo.php>>.

PINNSCHMIDT, H.; BATCHELOR, W.; TENG, P. Simulation of multiple species pest damage in rice using ceres-rice. *Agricultural Systems*, v. 48, p. 193–222(30), 1995. Disponível em: <<http://www.ingentaconnect.com/content/els/0308521x/1995/00000048/00000002/art00012>>.

PIVONIA, S.; YANG, X. Relating epidemic progress from a general disease model to seasonal appearance time of rusts in the united states: implications for soybean rust. *Ecology and Phytopathology*, The American Phytopathological Society, v. 96, p. 400–407, 2006.

PLANK, J. Van der. *Plant diseases: epidemics and control*. New York, USA: Academic Press, 1963.

PORTER, C. H.; BRAGA, R. P.; JONES, J. W. *An approach for modular crop. model development*. 1999. Disponível

em: <http://www.icasa.net/modular/pdf/modular.pdf>. Acesso em 13/04/2007.

PRATES, L. G.; FERNANDES, J. M. C. Avaliando a taxa de expansão de lesões de *Bipolaris sorokiniana* em trigo. *Fitopatologia Brasileira*, scielo, v. 26, p. 185–191, 06 2001. ISSN 0100-4158. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-41582001000200012&nrm=iso>.

R Development Core Team. *R: A language and Environment for Statistical Computing*. Vienna, Austria, 2006. ISBN 3-900051-07-0. Disponível em: <<http://www.R-project.org>>.

RAPILLY, F. Partial resistances of wheat to septoria nodorum berk. : From the mathematical simulation to the study of resistance chromosomes. *Z. f. Pflanzenkr. u. Pflanzensch.*, http://www.numdam.org/item?id=RSA_1976_24_3_31_0, v. 95, n. 4, p. 419–427, 1988.

RAPILLY, F.; JOLIVET, E. Construction d'un modèle (episept) permettant la simulation d'une épidémie de septoria nodorum berk. sur blé. *Rev. Sta. Appl.*, http://www.numdam.org/item?id=RSA_1976_24_3_31_0, v. 3, p. 31–60, 1976.

REIS, E. M.; CASA, R. T.; MEDEIRA, C. A. *Diagnose, patometria e controle de doenças de cereais de inverno*. Londrina, PR, Brasil: Londrina: ES, 2001.

REYNOLDS, J.; ACOCK, B. Modularity and genericness in plant and ecosystem models. *Ecological Modelling*, v. 94, n. 1, p. 7–16, 1997. ISSN 0304-3800.

RIBEIRO JUNIOR, P. J. *Introdução ao R*. 2001. Disponível em: <http://www.leg.ufpr.br/Rtutorial>. Acesso em 10/04/2007.

ROBERT, C. et al. Analysis and modelling of effects of leaf rust and *Septoria tritici* blotch on wheat growth. *J. Exp. Bot.*, v. 55, n. 399, p. 1079–1094, 2004. Disponível em: <jxb.oxfordjournals.org/cgi/content/abstract/55/399/1079>.

ROBERTS, C. A.; DESSOUKY, Y. M. An Overview of Object-Oriented Simulation. *SIMULATION*, v. 70, n. 6, p. 359–368, 1998. Disponível em: <<http://sim.sagepub.com/cgi/content/abstract/70/6/359>>.

ROBINSON, D. C.; GEILS, B. W. Modelling dwarf mistletoe at three scales: life history, ballistics and contagion. *Ecological Modelling*, Elsevier, v. 199, n. 1, p. 23–38, 2006.

ROERMUND, H. J. W. V.; SPITTERS, C. J. T. Simulation of yield reduction by leaf rust in winter wheat, applied to the analysis of genetic variation in partial resistance. *Europeanjournal of Plant Pathology*, Springer Netherlands, <http://www.springerlink.com/content/r6061521r61j6172>, v. 96, n. 1, p. 17–28, 1990.

ROSSI, V. et al. A simulation model for the development of brown rust epidemics in winter wheat. *Europeanjournal of Plant Pathology*, The American Phytopathological Society, v. 103, n. 5, p. 453–465, 1997. ISSN 0929-1873.

ROSSING, W. A. H.; DAAMEN, R. A.; HENDRIX, E. M. T. Framework to support decisions on chemical pest control under uncertainty, applied to aphids and brown rust in winter wheat. *Crop Protection*, v. 13, n. 1, p. 25–34, 1994.

RSERVE. *About Rserve*. 2007. Disponível em: <http://www.rforge.net/Rserve/>. Acesso em: 15/04/2007.

SACHE, I. Effect of density and age of lesions on sporulation capacity and infection efficiency in wheat leaf rust (*puccinia recondita* f.sp. *tritici*). *Plant Pathology*, v. 46, n. 4, p. 581–589, 1997. Disponível em: <<http://www.blackwell-synergy.com/doi/abs/10.1046/j.1365-3059.1997.d01-33.x>>.

SACKETT, K. E.; MUNDT, C. C. The effects of dispersal gradient and pathogen life cycle components on epidemic velocity in computer simulations. *Phytopathology*, v. 95, n. 9, p. 992–1000, 2005. Disponível em: <<http://apsjournals.apsnet.org/doi/abs/10.1094/PHYTO-95-0992>>.

SANHUEZA, R. M. V. A produção integrada a maçã no brasil. In: *III Seminário Brasileiro de Produção Integrada de Frutas*. Fortaleza, Brasil: Embrapa Agroindústria Tropical, 2007. p. 53.

SAVARY, S. et al. Quantification and modeling of crop losses: A review of purposes. *Annual Review of Phytopathology*, v. 44, n. 1, p. 89–112, 2006. Disponível em: <<http://arjournals.annualreviews.org/doi/abs/10.1146/annurev.phyto.44.070505.143342>>.

SCHEEREN, P. et al. *Efeito do frio em trigo (Embrapa Trigo. Comunicado Técnico Online, 57)*. 2000. Disponível em: http://www.cnpt.embrapa.br/biblio/p_co57.htm. Acesso em 20/06/2007.

SCHERM, H.; YANG, X. B. Interannual variations in wheat rust development in china and the united states in relation to the el niño/southern oscillation. *Phytopathology*, The American Phytopathological Society, http://www.apsnet.org/phyto/PDFS/1995/Phyto85n09_970.PDF, v. 85, p. 970–976, 1995.

SEEM, R. Plant disease forecasting in the era of information technology. In: *Plant Disease Forecast: Information Technology in Plant Pathology*. Kyongju, Republic of Korea: [s.n.], 2001.

SENTELHAS, P. C. *Duração do período de molhamento foliar: aspectos operacionais da sua medida, variabilidade espacial em diferentes culturas e sua estimativa a partir do modelo de Penman-Monteith*. 2004. Disponível em: <http://www.teses.usp.br/teses/disponiveis/livredocencia/11/tde-07062006073629>. Acesso em 18/06/2007.

SENTELHAS, P. C. et al. Spatial variability of leaf wetness duration in different crop canopies. *International Journal of Biometeorology*, v. 49, n. 6, p. 363–370, 2005. Disponível em: <<http://www.springerlink.com/content/lp77648264258g47>>.

SHANER, G. Growth of uredinia of puccinia recondita in leaves of slow- and fast- rusting wheat cultivars. *Phytopathology*, v. 73, p. 931–935, 1983. ISSN 0031-949X.

SHANNON, R. *Systems simulation: the art and science*. New Jersey, USA: Prentice-Hall, 1975. 387 p.

SHAW, R. *Yield loss estimation*. [S.l.]: St. Paul: APS Press, 1997. 143 p.

SHETTY, N. P. et al. Role of hydrogen peroxide during the interaction between the hemibiotrophic fungal pathogen septoria tritici and wheat. *New Phytologist*, <http://www.ingentaconnect.com/content/bsc/nph/2007/00000174/00000003/art00019>, v. 174, p. 637–647, 2007.

SHOEMAKER, P.; LORBEER, J. The role of dew and temperature in the epidemiology of botrytis leaf blight of onion. *Phytopathology*, v. 67, p. 1267–1272, 1977. ISSN 0031-949X.

SILVA JUNIOR, V. V. *Perspectivas para Uso da Linguagem R*. 2005. Disponível em: <http://mov.void.cc/articles/r-perspectivas/r-perspectivas.pdf>. Acesso em 10/04/2007.

SILVEIRA, I. F. *Linguagem JAVA*. 2007. Disponível em: <http://www.infowester.com/lingjava.php>. Acesso em 13/04/2007.

SRIDHARAN, C.; RIEKEN, B.; PETERSON, L. *Advanced Java Networking*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.

SUBBARAO, K.; MICHAILIDES, T. Effects of temperature on isolates of fusarium moniliforme causing fig endosepsis and aspergillus niger causing smut. *Phytopathology*, v. 85, p. 662–668, 1995. ISSN 0031-949X.

SUTHERST, R. W.; MAYWALD, G. F. A computerised system for matching climates in ecology. *Agricultural Ecosystem Enu.*, v. 13, p. 281–299, 1985.

SWANEY, D. et al. Using crop models for management: Impact of weather characteristics on irrigation decisions in soybeans. *Transactions of the American Society of Agricultural Engineers*, v. 26, n. 6, p. 1808–1814, 1983.

TIERNEY, L. Name space management for R. *R News*, v. 3, n. 1, p. 2–6, June 2003. Disponível em: <http://CRAN.R-project.org/doc/Rnews/>.

TRIGO, C. S.-B. de Pesquisa em. *36^a Reunião da Comissão Sul-Brasileira de Pesquisa de Trigo*. Passo Fundo, RS, Brasil: Universidade de Passo Fundo (UPF), 2004. 152 p.

UNDERDAHL, B. *Macromedia Flash MX 2004: the complete reference*. [S.l.]: McGraw-Hill/Osborne, 2003. ISBN 0072229209.

URBANER, S. *Rserve A fast way to provide R functionality to applications*. 2003. Disponível em: <http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Drafts/Urbaner.pdf>. Acesso em: 15/04/2007.

VALE, F.; JUNIOR, W.; ZAMBOLIM, L. *Epidemiologia aplicada ao manejo de doenças de plantas*. Belo Horizonte, MG, Brasil: Perfil, 2004. ISBN 85-89800-02-4.

VALE, F. X. R. D.; ZAMBOLIM, L. Influência da temperatura e da umidade no desenvolvimento de doenças de plantas. *Revisão Anual de Patologia de Plantas*, Passo Fundo, RS, Brasil, v. 4, p. 149–207, 1996. ISSN 0104-0383.

VALLAVIEILLE-POPEA, C. d. et al. Assessment of epidemiological parameters and their use in epidemiological and forecasting models of cereal airborne diseases. *Agronomie*, v. 20, p. 715–727, 2000. Disponível em: <http://www.agronomy-journal.org/articles/agro/pdf/2000/07/a0703.pdf>.

VARGAS, P. et al. Simulação de epidemia de giberela em trigo. *Fitopatologia Brasileira*, Sociedade Brasileira de Fitopatologia, v. 25, p. 497–504, 2000. ISSN 0100-4158.

VEIT, M.; HERRMANN, S. Model-view-controller and object teams: a perfect match of paradigms. In: *AOSD '03: Proceedings of the 2nd international conference on Aspect-oriented software development*. New York, NY, USA: ACM Press, 2003. p. 140–149. ISBN 1-58113-660-9.

VERHAGEN, A.; CONIJN, S.; SCHAPENDONK, A. *Quicks-can of simulations models*. [S.l.: s.n.], 2001. Disponível em: http://library.wur.nl/wasp/bestanden/LUWPUBRD_00121624_A502_001.pdf. Acesso em 01/12/2006.

VITTI, A. et al. Comparative epidemiology of common maize rust and northern corn leaf blight. i. effect of environment on monocyclic parameters. *Summa Phythopathologica*, v. 21, p. 127–130, 1995.

WAGGONER, P.; BERGER, R. Defoliation, disease, and growth. *Phytopathology*, American Phytopathological Society, v. 77, n. 3, p. 393–398, 1987. ISSN 0031-949X.

WAGGONER, P.; HORSFALL, J. Epidem, a simulator of plant disease written for a computer. *Conn. Agric. Exp. Stn. Bull.*, v. 698, n. 1, p. 1–80, 1969.

WAGGONER, P.; HORSFALL, J.; LUKENS, R. Epimay, a simulator of southern corn leaf blight. *Conn. Agric. Exp. Stn. Bull.*, v. 729, 1972.

WEBB, C.; GILLIGAN, C.; ASHER, M. Modelling the effect of temperature on the dynamics of polomyxa betae, the vector of rhizomania disease. *Plant Pathology*, v. 49, p. 600–607, 2000.

WILLOCQUET, L.; SAVARY, S. An epidemiological simulation model with three scales of spatial hierarchy. *Analytical and Theoretical Plant Pathology*, The American Phytopathological Society, v. 94, n. 8, p. 883–891, 2004.

WILLOCQUET, L. et al. Structure and validation of ricepest, a production situation-driven, crop growth model simulating rice yield response to multiple pest injuries for tropical asia. *Ecological Modelling*, v. 153, p. 247–268(22), 2002. Disponível em: <<http://www.ingentaconnect.com/content/els/03043800/2002/00000153/00000003/art00014>>.

WIT, C. D. *Coordination of models*. In: *PENNING DE VRIES, F.W.T. AND Van LAAR, H.H. (Ed.)*. [S.l.]: Wageningen: PUDOC, 1982.

WIT, C. de. Photosynthesis of leaf canopies. *Agric. Res. Rep.*, Pudoc, Wageningen, the Netherlands, v. 663, 1965.

WOLF, E. D. D.; FRANCL, L. J. Neural network classification of tan spot and stagonospora blotch infection periods in a wheat field environment. *Phytopathology*, The American Phytopathological Society, v. 90, n. 2, p. 108–113, 2000.

YANG, X. B. Risk assessment: concepts, development, and future opportunities. *Plant Health Progress*, <http://www.cababstractsplus.org/google/abstract.asp?AcNo=20043010510>, v. 31, n. 6, p. 1–6, 2003.

YANG, X. B.; DOWLER, W. M.; TSCHANZ, A. T. A simulation model for assessing soybean rust epidemics. *Journal of phytopathology*, Blackwell, Berlin, ALLEMAGNE (1986) (Revue), v. 133, n. 3, p. 187–200, 1991. ISSN 0931-1785.

YOUNG, C. S. et al. Predicting epidemics of yellow rust (*puccinia striiformis*) on the upper canopy of wheat from disease observations on lower leaves. *Plant Pathology*, v. 52, n. 3, p. 338–349, 2003. Disponível em: <<http://www.blackwell-synergy.com/doi/abs/10.1046/j.1365-3059.2003.00848.x>>.

ZADOKS, J. Epipre, a computer-based decision support system for pest and disease control in wheat: its development and implementation in europe. 1989. Disponível em: <<http://library.wur.nl/WebQuery/wurpubs/9400>>.

ZADOKS, J.; SCHEIN, R. *Epidemiology and Plant Disease Management*. [S.l.]: Oxford University Press, New York, 1979.