

## Desenvolvimento de simuladores na agropecuária

Luís Gustavo Barioni  
Adauto Luiz Mancini

### 1 Introdução

Simuladores computacionais são softwares que fazem uso de modelos matemáticos e algoritmos para representar um sistema (NUTARO, 2011). A simulação, por sua intrínseca associação com a modelagem matemática, é frequentemente designada “modelagem e simulação” (M&S). As finalidades da aplicação de M&S são diversas e incluem:

- a) Geração, formalização e avaliação do conhecimento científico, particularmente em sistemas complexos.
- b) Apoio ao processo decisório e à resolução de problemas gerenciais.
- c) Educação e treinamento.
- d) Transferência de tecnologia.

Apesar de suas origens na matemática aplicada e na computação, o estudo e aplicação da M&S tem se tornado fundamental nas ciências sociais, humanas, biológicas e da saúde.

Um dos grandes atrativos de simuladores está em prover um ambiente virtual que permite integrar com representações de sistemas, naturais ou artificiais sem as limitações do mundo real. Na realidade, estamos restritos por espaço e tempo e na instrumentação disponível para impor determinados tratamentos e mensurar respostas. Além disso, altos custos e riscos, questões éticas e ambientais, frequentemente, inviabilizam a experimentação em muitas situações. A utilidade da virtualização é evidenciada pelo funcionamento da nossa própria mente. De acordo com Pezzulo et al. (2013), nosso cérebro constantemente simula nossas interações com o meio antes de desencadear ações conscientes.

Do ponto de vista científico e tecnológico, a M&S também tem sido apontada como instrumento de excelência para estudar sistemas naturais e para apoiar a elaboração de projetos e a gestão de sistemas artificiais, por permitir considerar simultaneamente intrincadas interações entre componentes e processos em um sistema e formalizar a lógica associada à emergência de propriedade e fenômenos (JOHNSON et al., 2013; KEATING, 2008).

Na agropecuária há crescente aplicação de simuladores em diversas áreas, tais como: desenvolvimento e produção de plantas e de animais (JONES et al., 2003; OLTJEN, 1986), apoio à tomada de decisões gerenciais (BARIONI et al., 2010; FREER et al., 1997), dinâmica de pragas, doenças

e contaminantes (ZAGMUTT et al., 2013), avaliações de impacto ambiental (ITTERSUM et al., 2008), dinâmica de uso da terra (VERBURG et al., 2004), manejo da água (ASCOUGH et al., 2012) e avaliação de tecnologias em geral (WOODWARD et al., 2008).

É grande a diversidade de ferramentas e paradigmas para implementação de simuladores na agropecuária, o que reflete a busca por um balanço entre facilidade de aprendizado e uso, padronização, flexibilidade e desempenho, e de sua adequação às características da equipe de desenvolvimento (em fase de elaboração)<sup>1</sup>. Com a profusão de iniciativas de desenvolvimento de simuladores na agropecuária, algumas das maiores empresas de pesquisa na área têm alocado equipes dedicadas a seus próprios arcabouços e infraestrutura para simulação (e.g., o OMS (DAVID, 2013), desenvolvido pelo Departamento de Agricultura dos Estados Unidos (USDA); o APSIM (MCCOWN et al., 1996), desenvolvido pelo Commonwealth Scientific and Industrial Research Organization (CSIRO); e os arcabouços VLE (QUESNEL et al., 2007) e Record (BERGEZ et al., 2013), desenvolvidos pelo Institut National de la Recherche Agronomique (Inra).

Nesse contexto, este capítulo apresenta conceitos e experiências relacionados ao desenvolvimento de simuladores na agropecuária, com foco nos trabalhos em andamento na Embrapa Informática Agropecuária, Unidade da Empresa Brasileira de Pesquisa Agropecuária (Embrapa), e também uma visão de futuro sobre esses trabalhos.

## 2 Conceitos básicos

Nutaro (2011) distingue três componentes básicos em softwares de simulação computacional:

- 1) O modelo matemático.
- 2) O simulador.
- 3) A interface de entrada e saída.

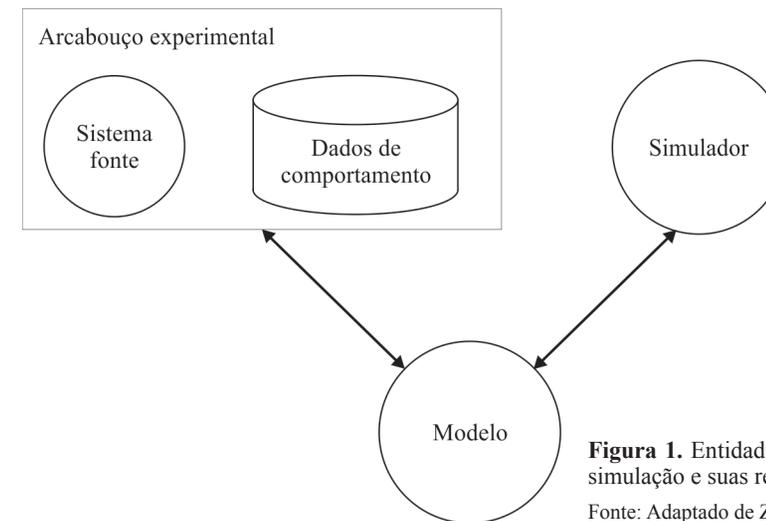
O modelo matemático é uma representação, utilizando linguagem e conceitos matemáticos, das relações e interações relevantes do sistema para o propósito para o qual o modelo se destina. Zeigler et al. (2000) utiliza o termo sistema-fonte para definir o sistema real que estamos modelando. O processo de desenvolvimento do modelo matemático a partir do sistema-fonte é normalmente denominado modelagem ou modelação.

O processo de modelagem se inicia pelo julgamento, de certo modo arbitrário ou empírico, de quais são a fronteira e os aspectos relevantes do sistema-fonte a serem reproduzidos. A fronteira do sistema define as variáveis, processos e inter-relações endógenos ao modelo. A fronteira é também o ponto de partida para identificarmos variáveis exógenas que são fatores determinantes do comportamento do sistema, também denominados em inglês *drivers*, os quais deverão fazer parte das entradas para a simulação, mas não serão afetados por ela.

O modelo conceitual é usado, sequencialmente, para definir, a partir do conhecimento teórico existente, como será representado o estado do sistema e quais os processos e fatores determinan-

tes (*drivers*) de sua dinâmica. Durante a modelagem conceitual deve haver, todavia, a escolha de um paradigma para a modelagem, que irá guiar os passos seguintes do processo de modelagem e de codificação do simulador. Vários paradigmas podem ser utilizados. Na agricultura há uma tendência para a aplicação do paradigma de dinâmica de sistemas contínuos, por meio de sistemas deterministas de equações diferenciais ou de diferenças. Entretanto, outros paradigmas, como simulação orientada a eventos discretos (BARIONI et al., 1999), simulações baseadas em indivíduos, simulação estocástica, e simulação-otimização são também utilizados. A escolha de um paradigma e a área de origem dos profissionais envolvidos no trabalho de modelagem e simulação direciona fortemente a metodologia de modelagem e codificação e os artefatos utilizados. O passo seguinte é a formalização matemática e algorítmica a partir do modelo conceitual e da codificação. Esses passos serão tratados mais detalhadamente ao longo do capítulo.

Zeigler et al. (2000) fazem uma útil distinção entre modelos e simuladores (Figura 1). Eles conceituam modelos como especificações (conjunto de instruções) e simuladores como os programas responsáveis por gerar o comportamento especificado em um computador. Modelos matemáticos podem, portanto, ser armazenados como arquivos contendo informações estáticas e independentes do programa que os executa. Um modelo pode, por essa abordagem, ser executado por mais de um simulador e o simulador pode executar diferentes modelos.



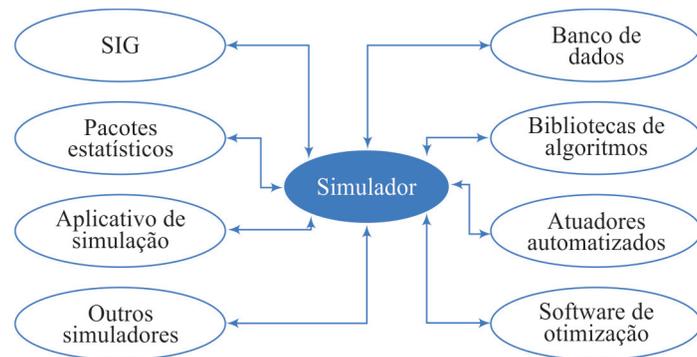
**Figura 1.** Entidades básicas em modelagem e simulação e suas relações.

Fonte: Adaptado de Zeigler et al. (2000).

Dados obtidos a partir do sistema-fonte são essenciais no processo de modelagem. Zeigler et al. (2000) denomina a estrutura envolvida na geração de dados sobre o sistema-fonte de arcabouço experimental. Dados observados são utilizados em diversas fases do processo, particularmente na definição da forma funcional das relações entre as variáveis, na determinação do valor de parâmetros (parametrização), na avaliação do modelo e em processos adaptativos envolvendo a contínua calibração do modelo a partir de dados observados. Dessa forma é equivocado pensar que M&S substitua ou enfraqueça o trabalho relacionado ao arcabouço experimental. Pelo contrário, espera-se que o arcabouço experimental seja ainda mais demandado e que essa demanda seja mais objetiva em função de lacunas de conhecimento identificadas no processo de representação do sistema para simulá-los.

<sup>1</sup> Comparison between three different approaches to implement a system dynamic model: an assessment by a multi-disciplinary team, de autoria de M. C. Freua, L. G. Barioni, R. Vilamiu, F. R. T. A. Dias a ser editado pela Embrapa Informática Agropecuária, 2014.

Além do modelo e do simulador, outros componentes de software ainda são necessários para que um software de simulação seja utilizado. Esses outros componentes estão relacionados ao pré-processamento, armazenamento e comunicação de dados de entrada e saída, bem como interfaces entre o simulador e o usuário. Existe grande variedade nas interfaces para simuladores (Figura 2).



**Figura 2.** Exemplos de possíveis interfaces de um simulador computacional com outros componentes em uma aplicação de simulação e de outros aplicativos em uma solução computacional para simulação de sistemas.

Muitos simuladores utilizam interfaces tradicionais, do tipo desktop, tais como a dos softwares DSSAT (JONES et al., 2003), Grazplan (FREER et al., 1997) e Invernada (BARIONI et al., 2010). Em outras oportunidades são utilizadas interfaces com sistemas de informação geográfica (PRIYA; SHIBASAKI, 2001), softwares estatísticos (BEN TOUHAMI et al., 2013), gerenciadores de processamento paralelo, sistemas de *workflow* (LUDÄSCHER, 2006), entre outros.

Em razão de simuladores serem mais precisos à medida que aumenta o número de iterações nos algoritmos de integração numérica, esses são frequentemente ávidos por capacidade de processamento. Frequentemente a aplicação de simuladores demanda computação de alto desempenho, envolvendo processamento paralelo em muitos processadores arranjados em *clusters*, grids ou nuvens.

Há ainda pouca exploração, na agricultura, de uso de interfaces avançadas como as encontradas em outras áreas de aplicação, tais como jogos educativos ou simuladores com atuadores mecânicos. Além disso, um simulador pode ser cliente de outro simulador na forma de pipeline. A Embrapa informática Agropecuária também desenvolve um protótipo de jogo digital baseado em um simulador de sistemas de produção de bovinos (em fase de elaboração)<sup>2</sup>.

### 3 Métodos para o desenvolvimento de simuladores

Em muitas situações é possível que já exista um simulador adequado para o problema que se deseja. Adotar um simulador existente gera enorme economia e o uso de um determinado simulador

por várias equipes e em várias situações tende a contribuir para sua robustez e confiabilidade. É necessário, entretanto, verificar a adequação de um simulador ao problema e dados disponíveis. Algumas das principais verificações são:

1. O simulador possui as variáveis de saída (resultados) necessárias para a solução almejada?
2. Os dados requeridos para executar o simulador, i.e., os valores das variáveis de entrada, existem ou podem ser gerados com o custo e em tempo adequados?
3. A acurácia do simulador é satisfatória para o propósito do trabalho ou pode ser melhorada sem alterações estruturais?
4. A infraestrutura computacional disponível é suficiente para os resultados no tempo exigido?

Sempre que a resposta a uma dessas questões for negativa, justifica-se o trabalho de modelagem e desenvolvimento de um novo simulador.

O desenvolvimento de simuladores na agricultura, sob visão holística, envolve diversos passos ou subprocessos, que partem da estruturação do arcabouço experimental para a geração de dados, informação e conhecimento até o desenvolvimento das funcionalidades acessórias do software e, eventualmente, da aquisição de hardware necessário à simulação.

Podemos distinguir os trabalhos envolvendo simuladores na agropecuária em, pelo menos, cinco níveis com crescente complexidade:

1. **Aplicação:** O simulador é utilizado na sua forma original tendo como mudança apenas os dados de entrada para uma determinada situação.
2. **Avaliação:** O simulador é utilizado na sua forma original, mas os resultados são comparados a dados observados visando avaliar se o comportamento é adequado em determinadas situações.
3. **Calibração:** O simulador é utilizado na sua forma original, mas os parâmetros de seu modelo matemático são modificados de forma a gerar resultados os mais próximos possíveis das observações.
4. **Modelagem matemática:** O modelo do simulador é alterado estruturalmente nas suas formas funcionais, processos e componentes.
5. **Desenvolvimento de softwares de simulação:** Além do modelo, outros componentes do software de simulação são desenvolvidos desde sua concepção ou significativamente alterados.

O desenvolvimento de simuladores para a agropecuária é raro no Brasil, particularmente para aqueles que se mostram promissores para uso fora do ambiente do grupo de pesquisa no qual foi gerado. Assim, o mais frequente é termos trabalhos de pesquisa no Brasil visando ao aperfeiçoamento de simuladores desenvolvidos no exterior, com processos de M&S sendo desenvolvidos por parceiros internacionais. Embora se possa valer do conhecimento e do trabalho feito por equipes internacionais, essa dependência no processo de modelagem e codificação não pode ser considerada ideal. Essa estratégia é particularmente indesejável quando os simuladores se prestam à avaliação de temas sensíveis em nível nacional, tais como segurança alimentar e mitigação das emissões de gases de efeito estufa e nos casos nos quais se veja valor comercial na aplicação do simulador envolvendo, portanto, questões de propriedade intelectual e royalties.

Em virtude da complexidade, projetos de M&S em níveis mais abrangentes exigem eficiente gestão da equipe multidisciplinar e, normalmente, envolvem custos elevados, grande esforço e

<sup>2</sup> Desenvolvimento de um jogo educativo baseado em um simulador de sistema de produção de bovinos de corte (Resumos), de autoria de Jair Bortolucci Junior e Adauto Mancini a ser editado pela Embrapa Informática Agropecuária, 2014.

tempo. O reúso de produtos de trabalhos anteriores, tanto de modelagem quanto de geração e de organização de dados, torna-se muito importante para viabilizar muitos projetos de desenvolvimento. Simuladores agropecuários que se destacam no contexto internacional são tipicamente originários de atividades multidisciplinares, envolvendo profissionais de diferentes áreas, incluindo ciências agrárias, física, matemática, química e computação. Além disso, projetos bem-sucedidos e de longa vida útil, normalmente têm por característica eficiente feedback entre a geração de dados em campo e processos de M&S.

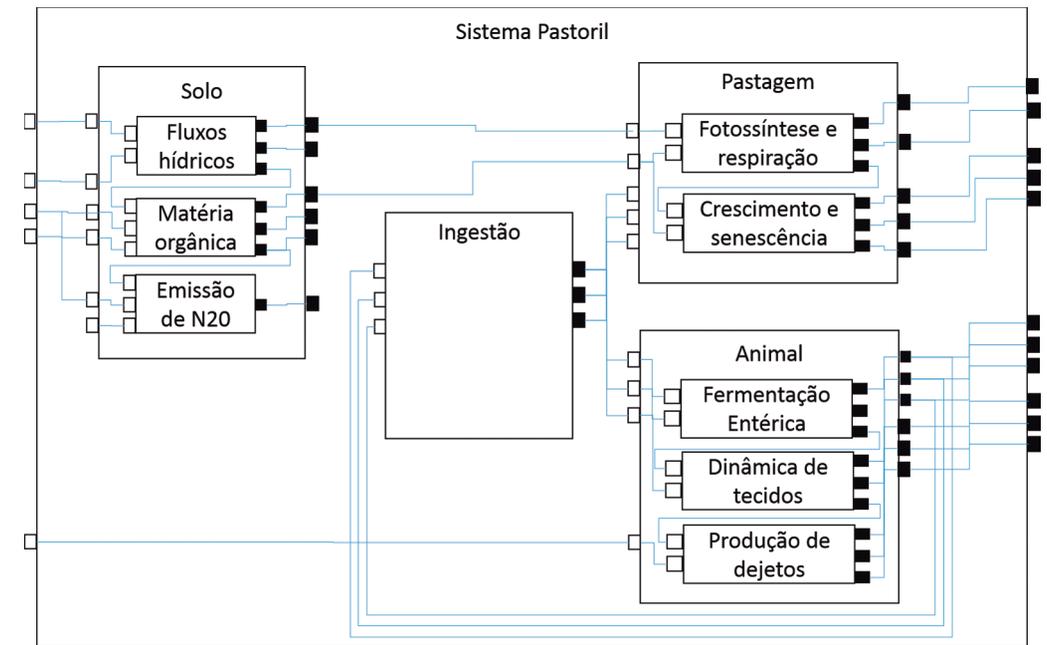
O desenvolvimento científico e tecnológico tem permitido a execução de simulações cada vez mais elaboradas. A obtenção de dados em larga escala, com aumento de resolução da grade dos dados de entrada e precisão dos valores obtidos, seja para processamento posterior ou em tempo real por meio de redes de sensores, associada ao contínuo aumento da capacidade de processamento dos computadores, está ampliando as possibilidades de monitoramento, reprodução e compreensão dos fenômenos observados. Os elementos dos sistemas de interesse podem variar em agregação, desde partículas subatômicas a galáxias. O uso de simuladores para estudar a relação entre processos e fenômenos, em diferentes níveis de agregação, para prever o comportamento de um sistema tem sido crescente, como também o número de níveis envolvidos.

Na maior parte dos casos, um ou mais modelos existem para um determinado processo a ser simulado. Uma vez que cada modelo de processo possa ser desenvolvido inicialmente de forma independente, é bastante conveniente que o modelo matemático de um simulador seja desenvolvido a partir do acoplamento de modelos componentes, facilitando a manutenção e o reúso de modelos componentes. Ademais, algumas abordagens e ferramentas de implementação de simuladores, particularmente a simulação orientada a objetos, permitem que os modelos componentes sejam agregados em estruturas hierárquicas (ZEIGLER et al., 2000), em analogia à agregação e decomposição de sistemas (WU; DAVID, 2002). Esse acoplamento de modelos desenvolvidos por diferentes equipes reimplementados para um determinado software também é realizado em importantes projetos de desenvolvimento de simuladores.

Na Figura 3, pode-se observar um exemplo de acoplamento de modelos componentes em uma estrutura hierárquica. Um modelo componente possui portas de entrada (representadas por pequenos quadrados vazios) e portas de saída (representadas por pequenos quadrados preenchidos em preto) que podem ser utilizadas para estabelecer conexões (linhas associando portas) com outros modelos. Modelos componentes podem abrigar outros componentes em tantos níveis quanto necessário. Portas de modelos componentes também podem ser associadas com portas no nível superior, permitindo que informações sejam transferidas entre níveis hierárquicos.

Olhando por essa perspectiva, Rotmans (2009) afirma que o maior desafio está, na atualidade, na combinação ou acoplamento dos modelos de forma a gerar soluções para as diferentes demandas por simulação. Seguindo filosofia similar, na Embrapa Informática Agropecuária tem-se trabalhado no desenvolvimento de componentes de software que facilitem a especificação de modelos componentes, a implementação do simulador, a execução de simulações e os processos de avaliação e calibração dos modelos.

O arcabouço (ou *framework*, do termo em inglês) de simulação MaCSim é o componente de software central para implementação de simuladores do Laboratório de Matemática Computacional da Embrapa Informática Agropecuária. É um arcabouço de simulação para implementação que



**Figura 3.** Exemplo do acoplamento em estruturas hierárquicas. Detalhes sobre portas, conexões e hierarquias são apresentados no corpo do texto.

permite o desenvolvimento de simuladores orientados a objetos e sua compilação como bibliotecas de vínculo dinâmico (MANCINI et al., 2013).

O paradigma de simulação orientada a objetos usado nessa ferramenta está alinhado com a maioria dos arcabouços de simulação utilizados atualmente como, por exemplo, JDEVS (FILIPPI; BISGAMBIGLIA, 2003); VLE (QUESNEL et al., 2007) e SimPy<sup>3</sup>. A linguagem escolhida foi C++, que oferece ampla versatilidade e eficiência do código e que possui compiladores para praticamente todos os sistemas operacionais atuais, além de ser suportada por diversos ambientes integrados de desenvolvimento gratuitos como Eclipse, NetBeans, Code::Blocks e Qt.

Os principais requisitos para a construção do arcabouço foram modularidade, suporte à hierarquia na estruturação de modelos, padronização de código e reúso de modelos componentes. Outros requisitos levantados pela equipe de desenvolvimento foram:

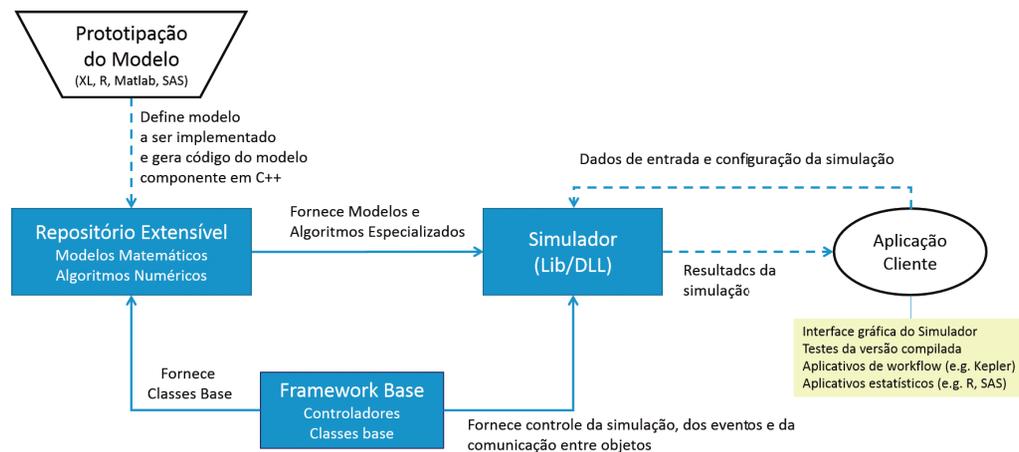
- Suporte à simulação contínua e híbrida.
- Especificação textual padronizada dos modelos para facilitar a interação com especialistas de domínio não programadores.
- Código de baixa complexidade para facilitar seu entendimento por alunos de graduação.
- Separação do código de controle da simulação do código para desenvolvimento de modelos.
- Interface simples para facilitar a comunicação entre a aplicação cliente e o simulador.
- Alto desempenho computacional via geração de arquivo binário executável.

<sup>3</sup> Disponível em: <[www.sourceforge.net/projects/simpy](http://www.sourceforge.net/projects/simpy)>.

A concepção do projeto do arcabouço de software permitiu o desenvolvimento independente dos modelos, de forma a gerar um repositório de modelos que possam ser acoplados conforme a necessidade de uma solução de simulação.

Além disso, o simulador é independente da aplicação cliente (por exemplo, uma interface gráfica específica do usuário ou um aplicativo genérico, como R). Isto é possibilitado compilando um simulador produzido no arcabouço (controle da simulação e o modelo específico do usuário) como uma biblioteca de vínculo dinâmico, que pode ser chamada pela aplicação cliente (Figura 4). Isto é útil porque:

- Um mesmo simulador pode ser usado por diferentes aplicações.
- O simulador e a aplicação cliente podem ser desenvolvidos por times diferentes usando possivelmente diferentes linguagens de programação.
- O simulador pode ser usado por desenvolvedores de software sem necessidade de conhecimento profundo sobre o código do simulador.



**Figura 4.** Fluxograma do processo de desenvolvimento e aplicação de simuladores desenvolvidos com as ferramentas desenvolvidas na suíte MaCSim.

O mesmo simulador pode ser usado para diferentes propósitos, atendendo aplicações clientes distintas (executadas independentemente) e dados da simulação podem ser armazenados de diferentes modos. Por exemplo, um simulador de uma fazenda pode ser usado para uma simulação determinística e pontual em um sistema de suporte à decisão, mas também para uma simulação em grade para uma área geográfica mais abrangente e cenários de mudança climática de longo prazo.

A separação do simulador dos outros componentes de software pode resultar em trabalho mais eficiente e especializado conduzido paralelamente por equipes dedicadas. Neste contexto, componentes de software, que não sejam o simulador propriamente, podem ser desenvolvidos por instituições parceiras. Um protocolo de comunicação padrão, definido pelo arcabouço, facilita o desenvolvimento e minimiza erros, uma vez que chamadas entre os componentes da aplicação cliente e o simulador não precisam ser redesenhadas para cada uso específico das aplicações clientes. A produtividade também pode ser aumentada, porque cada aplicação poderá usar o

mesmo protocolo com o simulador, de modo que desenvolvedores precisarão aprendê-lo apenas uma vez e não será necessário conhecimento detalhado sobre o código do simulador. Estimação de parâmetros, avaliação de modelo, simulações estocásticas, também são beneficiadas por se ter o simulador compilado como uma biblioteca e usado por uma aplicação cliente.

O arcabouço foi concebido para que a aplicação cliente se encarregue do armazenamento de dados. Esta abordagem difere daquela descrita por Bolte (1998), em que o armazenamento de dados é feito pelo software simulador. O desacoplamento dos componentes simuladores do serviço de armazenamento de dados provê flexibilidade, pois a aplicação cliente pode armazenar as saídas do modo mais adequado para atender sua demanda de análise (memória, arquivo texto ou banco de dados). Também melhora a eficiência porque evita armazenamento desnecessário dos dados de cada iteração gravando-os em disco a cada instante, por exemplo.

O MacSim não possui interface gráfica para modelagem ou para a execução da simulação. Assim, para públicos sem treinamento em programação, seu uso para modelagem e simulação depende de ferramentas para prototipação e execução dos modelos desenvolvidos nesse framework.

Para facilitar a prototipação de modelos, desenvolveu-se uma planilha eletrônica, programada em Visual Basic para Aplicativos, que permite a especificação de modelos componentes (SILVA; BARIONI, 2013). A planilha suporta modelos estáticos e sistemas de equações diferenciais ordinárias. Essa ferramenta permite especificar, documentar, simular e gerar código dos modelos em C++ para o *framework* MaCSim e em linguagem R para prototipação mais avançada.

Conforme ilustrado na Figura 4, o *framework* de simulação necessita de uma interface para execução de simulações, estudo de cenários, calibração de modelos, entre outras funcionalidades. A Embrapa Informática Agropecuária desenvolveu, então, uma interface em linguagem R, denominada MaCSimR, para execução das bibliotecas de vínculo dinâmico contendo os simuladores. O MaCSimR permite:

- Acoplar os modelos, incluindo sua estrutura hierárquica, a partir de modelos componentes disponíveis na biblioteca do simulador.
- Carregar dados de arquivos em disco, de forma semi-automática ou utilizando funcionalidades nativas do R ou de suas extensões.
- Executar simulações.
- Recuperar trajetórias das variáveis de interesse (de entrada ou de saída) fornecendo somente o nome da variável.
- Plotar variáveis de interesse fornecendo seu nome.
- Executar experimentos com o modelo, alterando as variáveis de entrada.
- Utilizar algoritmos disponíveis em R para análises estatísticas, calibração e otimização em conjunto com simuladores compilados utilizando o MaCSim.

Experiências recentes com essas ferramentas em projetos de pesquisa da Embrapa têm demonstrado um grande potencial delas. Um exemplo de *script* no MaCSimR é apresentado na Figura 5.

```

source("MacSimR Simulation v2.R") # Runs the MacSimR script which should be in the working dir
Pecus <- Simulation(Simulator="DLLtoR.dll", Name = "Pecus")
ReadStdVal(GetInputs(Pecus), filename = "Input.txt", hasheader =F)
ReadColData(GetInputs(Pecus), "Clima_Piracicaba.csv", modelvar = "Tmin", filevar = "Tmin")
ReadColData(GetInputs(Pecus), "Clima_Piracicaba.csv", modelvar = "Tmed", filevar = "Tmed")
ReadColData(GetInputs(Pecus), "Clima_Piracicaba.csv", modelvar = "Rain", filevar = "Chuva")

Run(Pecus, |
  t0 = 0,
  tf = 365,
  dt = 1,
  outtimes = seq(0, 360, by = 10),
  InitialDate = Sys.Date(),
  IntegrationMethod = "RK4",
  VarsToSave = c("Tmin", "Rain", "IME", "CH4"),
  AutoSave = T
)

```

Figura 5. Exemplo de *script* do MacSimR para execução do modelo e plotagem de dados.

## 4 Perspectivas

Determinar as possíveis trajetórias de desenvolvimento e aplicação de simuladores na agropecuária brasileira para o futuro pode ser feito a partir de diferentes fontes de informação, dentre elas:

- Os simuladores, arcabouços de simulação e as formas de geração e organização de dados para M&S na agropecuária, já desenvolvidos ou em desenvolvimento, bem como as metodologias de desenvolvimento empregadas.
- A aplicação de simulação computacional em outras indústrias em que a simulação vem sendo aplicada há mais tempo, como na engenharia mecânica e mecatrônica, na climatologia, na neurociência e na indústria de entretenimento.
- As projeções de evolução de áreas relacionadas aos principais gargalos para o desenvolvimento de simuladores, tais como a formação de equipes multidisciplinares, a organização de dados, o desenvolvimento de software e a infraestrutura de hardware para suportar as simulações.

Temas como segurança alimentar, mitigação e adaptação às mudanças climáticas e comércio internacional têm sido os maiores demandantes de simuladores mais complexos. Nesses temas há tendência de trabalhar-se com maior número de níveis de agregação em simuladores na agropecuária, tornando frequente a busca por conexões entre observações na escala de parcela, processos biofísicos e da produção na escala da fazenda e consequências nos âmbitos regional, nacional ou global. Além disso, esses simuladores trabalham com abrangência temporal mais ampla e buscam prever os impactos de intervenções, tais como políticas públicas. Assim esses temas têm requerido simulações com abrangência de espaço e tempo muito mais amplas, gerando demanda por processamento milhares e, por vezes, milhões de vezes maiores que aquela demanda para a simulação de processos em simuladores de sistemas produtivos do início do milênio.

Em resposta às novas demandas e à expansão da aplicação da simulação na agropecuária, há atualmente forte tendência para o desenvolvimento de simuladores a partir de ferramentas que permitam alto desempenho computacional e reúso de modelos na forma de “soluções de simulação” a partir do acoplamento de modelos componentes (BERGEZ et al., 2013).

Embora simulações continuem sendo feitas a partir de populações, o uso de sensores que coletam informações fenotípicas em tempo real, aliados à parametrização de modelos com informação genômica, tem trazido à tona a possibilidade de simulação baseada em indivíduos e do uso de simuladores diretamente no melhoramento genético e na gestão de sistemas de produção de plantas e animais.

Em outras indústrias, aplicações importantes da simulação incluem o controle otimizado de sistemas, em intervenção automática por meios eletromecânicos e no treinamento de pessoas, áreas ainda incipientes na agropecuária. Além disso, em outras áreas há integração mais consistente com outras técnicas computacionais, tais como aquelas ligadas à inteligência computacional, otimização (i.e. simulação-otimização) e robótica. Observa-se que tanto a agricultura quanto a zootecnia de precisão caminham no sentido das demais indústrias mais intimamente ligadas à engenharia.

O uso de simuladores para treinamento de pessoas também parece ser bastante promissor nessa área que tem sido negligenciada na agropecuária. O treinamento por meio de jogos nos quais simuladores realistas são utilizados de forma iterativa pode promover experiência importante para gerenciar problemas de sistemas produtivos em situações de risco climático e de preços. Além disso, tal abordagem poderá também facilitar o ensino pela possibilidade de representar conceitos relacionados aos feedbacks que ocorrem em sistemas produtivos.

O desenvolvimento de simuladores agropecuários no Brasil parece ainda sofrer pela presença de vários gargalos, particularmente associados à falta de integração entre equipes de modelagem matemática, desenvolvimento de software e especialistas de domínio com pesquisa de campo em rede para o desenvolvimento de simuladores. Essa dificuldade na integração das equipes esbarra, via de regra, na falta de formação e treinamento de profissionais de ciências agrárias e pela carência de profissionais de ciências exatas com interesse em aplicações agropecuárias em detrimento de outras indústrias.

A Embrapa Informática Agropecuária tem investido, recentemente, na geração de ferramentas visando ao uso por profissionais com formação em ciências agrárias. O desenvolvimento dessas ferramentas tem ocorrido em conjunto com treinamento desses profissionais e a formação de redes de pesquisa. Tal iniciativa almeja dar suporte aos novos desafios de simulação na agricultura, por meio de uma visão holística e integrada, que parte da geração do dado até a solução de um problema de pesquisa auxiliado pelo desenvolvimento de um simulador.

## 5 Referências

- ASCOUGH, J. C.; DAVID, O.; KRAUSE, P.; HEATHMAN, G. C.; KRALISCH, S.; LAROSE, M.; AHUJA, L. R.; KIPKA, H. Development and application of a modular watershed-scale hydrologic model using the object modeling system: runoff response evaluation. *Transactions of the ASABE*, v. 55, n. 1, p. 117-135, 2012.
- BARIONI, G. L.; DAKE, C. K. G.; PARKER, W. J. Optimizing rotational grazing in sheep management systems. *Environment International*, New York, v. 25, n. 6-7, p. 819-825, Sept./Oct. 1999.
- BARIONI, L. G.; SILVA, O. D. D. da; OLIVEIRA, F. S. de. **Embrapa Invernada - Módulo de simulação de crescimento e composição de bovinos**. Versão 0.9. Campinas: Embrapa Informática Agropecuária, 2010. 1 CD-ROM.

BEN TOUHAMI, H.; LARDY, R.; KLUMPP, K.; BELLOCCHI, G. Bayesian calibration of the Pasture Simulation model (PaSim) to simulate emissions from long-term European grassland sites: a case study at Laqueuille (France). In: GREENHOUSE GASES AND ANIMAL AGRICULTURE CONFERENCE, 5., 2013, Dublin. **Proceedings...** Dublin: Cambridge University, 2013. p. 580.

BERGEZ, J. E.; CHABRIER, P.; GARY, C.; JEUFFROY, M. N.; MAKOWSKI, D.; QUESNEL, G.; RAMAT, E.; RAYNAL, H.; ROUSSE, N.; WALLACH, D.; DEBAEKE, P.; DURAND, P.; DURU, M.; DURY, J.; FAVERDIN, P.; GASCUEL-ODOUX, C.; GARCIA, F. An open platform to build, evaluate and simulate integrated models of farming and agro-ecosystems. **Environmental modelling & software**, Oxford, v. 39, p. 39-49, Jan. 2013.

BOLTE, J. Object-oriented programming for decision systems. In: PEART, R. M.; CURRY, R. B. (Ed.). **Agricultural systems modeling and simulation**. New York: Marcel Dekker, 1998. p. 629-650.

DAVID, O.; ASCOUGH, J. C.; LLOYD, W.; GREEN, T. R.; ROJAS, K. W.; LEAVESLEY, G. H.; AHUJA, L. R. A software engineering perspective on environmental modeling framework design: the object modeling system. **Environmental modelling & software**, v. 39, p. 201-213, Jan. 2013. DOI: 10.1016/j.envsoft.2012.03.006.

FILIPPI, J. B.; BISGAMBIGLIA, P. JDEVS: an implementation of a DEVS based formal framework for environmental modelling. **Environmental Modelling & Software**, Oxford, v. 19, n. 3, p. 261-274, Mar. 2003.

FREER, M.; MOORE, A. D.; DONNELLY, J. R. Grazplan: decision support systems for Australian grazing enterprises -II. The animal biology model for feed intake, production and reproduction and the GrazFeed DSS. **Agricultural Systems**, Essex, v. 54, n. 1, p. 77-126, May 1997. DOI: 10.1016/S0308-521X(96)00045-5.

ITTERSUM, M. K. van; EWERT, F.; HECKELEI, T.; WERY, J.; OLSSON, J. A.; ANDERSEN, E.; BEZLEPKINA, I.; BROUWER, F.; DONATELLI, M.; FLICHTMAN, G.; OLSSON, L.; RIZZOLI, A. E.; WAL, T. van der; WIEN, J. E.; WOLF, J. Integrated assessment of agricultural systems – a component-based framework for the European Union (SEAMLESS). **Agricultural Systems**, Essex, v. 96, n. 1-3, p. 150-165, Mar. 2008. DOI: 10.1016/j.agry.2007.07.009.

JOHNSON, J. J.; TOLK, A.; SOUSA-POZA, A. A theory of emergence and entropy in systems of systems. **Procedia Computer Science**, Amsterdam, v. 20, p. 283-289, 2013. DOI: 10.1016/j.procs.2013.09.274.

JONES, J. W.; HOOGENBOOM, G.; PORTER, C. H.; BOOTE, K. J.; BATCHELOR, W. D.; HUNT, L. A.; WILKENS, P. W.; SINGH, U.; GIJSMAN, A. J.; RITCHIE, J. T. The DSSAT Cropping System Model. **European Journal of Agronomy**, Amsterdam, v. 18, n. 3-4, Jan. p. 235-265, 2003. DOI: 10.1016/S1161-0301(02)00107-7.

KEATING, C. B. Emergence in System of Systems In: Jamshidi, M. (Ed.) **System of systems engineering: innovations for the 21st century**. Hoboken: J. Wiley, 2008. p. 169-190. (Wiley series in systems engineering and management).

LUDÄSCHER, B.; ALTINTAS, I.; BERKLEY, C.; HIGGINS, D.; JAEGER-FRANK, E.; JONES, M.; LEE, E.; TAO, J.; ZHAO, Y. Scientific workflow management and the kepler system. special issue: workflow in grid systems. **Concurrency and Computation: Practice & Experience**, Chichester, v. 18, n 10, p.1039-1065, Aug. 2006. DOI: 10.1002/cpe.v18:10.

MANCINI, A. L.; BARIONI, L. G.; SANTOS, E. H. dos; DIAS, F. R. T.; SANTOS, J. W. dos; ABREU, L. L. B. de; TININI, L. V. S. **Arcabouço para desenvolvimento de simuladores de sistemas dinâmicos contínuos e hierárquicos**. Campinas: Embrapa Informática Agropecuária, 2013. 19 p. (Embrapa Informática Agropecuária. Boletim de pesquisa e desenvolvimento, 34). Disponível em: <<http://www.infoteca.cnptia.embrapa.br/bitstream/doc/981039/1/BolPesq34cnptia.pdf>>. Acesso em: 20 set. 2014.

MCCOWN, R. L.; HAMMER, G. L.; HARGREAVES, J. N. G.; HOLZWORTH, D. P.; FREEBAIRN, D. M. APSIM: a novel software system for model development, model testing and simulation in agricultural systems research. **Agricultural Systems**, Essex, v. 50, n. 3, p. 255-271, 1996. DOI: 10.1016/0308-521X(94)00055-V.

NUTARO, J. J. **Building software for simulation: theory and algorithms, with applications in C++**. Hoboken: J. Wiley, 2011. 347 p.

OLTJEN, J. W.; BYWATER, A. C.; BALDWIN, R. L.; GARRETT, W. N. Development of a dynamic model of beef cattle growth and composition. **Journal of Animal Science**, Champaign, v. 62, n 1, p. 86-97, Jan. 1986.

PEZZULO, G.; CANDIDI, M.; DINDO, D.; BARCA, L. Action simulation in human brain: twelve questions. **New Ideas in Psychology**, Oxford, v. 31, n. 3, p. 270-290, Dec. 2013. DOI: 10.1016/j.newideapsych.2013.01.004.

PRIYA, S.; SHIBASAKI, R. National spatial crop yield simulation using GIS-based crop production model. **Ecological Modelling**, Amsterdam, v. 136, n. 2-3, p. 113-129, Jan. 2001. DOI: 10.1016/S0304-3800(00)00364-1.

QUESNEL, G.; DUBOZ, R.; RAMAT, E.; TRAORÉ, M. K. VLE: a multimodeling and simulation environments. In: SUMMER COMPUTER SIMULATION CONFERENCE, 2007, San Diego. **Proceedings...** San Diego: Society for Computer Simulation International, 2007. p. 367-374. SCSC'07.

ROTMANS, J. Three decades of integrated assessment: the way forward. Introductory Keynote. In: INTEGRATED ASSESSMENT OF AGRICULTURE AND SUSTAINABLE DEVELOPMENT CONFERENCE, 2009, Wageningen. **Setting the agenda for science and policy: proceedings**. Wageningen: Wageningen University, 2009. p. 1-35. Paper. AgSAP 2009.

SILVA, R. D. R. da; BARIONI, L. G. **Prototipação de modelos e geração automática de código-fonte em planilha eletrônica para o framework de modelagem MaCSim**. Campinas: Embrapa Informática Agropecuária, 2013. 13 p. (Embrapa Informática Agropecuária. Boletim de pesquisa e desenvolvimento, 35).

VERBURG, P. H.; SCHOT, P. P.; DIJST, M. J.; VELDKAMP, A. Land use change modelling: current practice and research priorities. **GeoJournal**, Dordrecht, v. 61, n. 4, p. 309-324, 2004. DOI: 10.1007/s10708-004-4946-y.

WOODWARD, S. J. R.; ROMERA, A. J.; BESKOW, W. B.; LOVATT, S. J. Better simulation modelling to support farming systems innovation: review and synthesis. **New Zealand Journal of Agricultural Research**, Wellington, v. 51, n. 3, p. 235-252, 2008. DOI: 10.1080/00288230809510452.

WU, J.; DAVID, J. L. A spatially explicit hierarchical approach to modeling complex ecological systems: theory and applications. **Ecological Modelling**, Amsterdam, v. 153, n. 1-2, p. 7-26, July 2002. DOI: 10.1016/S0304-3800(01)00499-9.

ZAGMUTT, F. J.; SEMPIER, S. H.; HANSON, T. R. Disease spread models to estimate highly uncertain emerging diseases losses for animal agriculture insurance policies: an application to the US farm-raised catfish industry. **Risk Analysis**, New York, v. 33, n. 10, p. 1924-1937, Oct. 2013. DOI: 10.1111/risa.12038.

ZEIGLER, B. P.; PRAEHOFER, H.; KIM, T. G. 2nd ed. **Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems**. San Diego: Academic Press, 2000. 510 p. ill.