

**LINGUAGEM DE PROGRAMAÇÃO NO DESENVOLVIMENTO DE MODELOS DE SIMULAÇÃO PARA SISTEMAS INTEGRADOS DE PRODUÇÃO AGROPECUÁRIA**

**PROGRAMMING LANGUAGE IN DEVELOPING SIMULATION MODELS FOR INTEGRATED CROP-LIVESTOCK SYSTEM**

José Carlos de Oliveira Júnior<sup>1\*</sup>, Gustavo Lineu Sartorello<sup>2</sup> e Augusto Hauber Gameiro<sup>2</sup>

<sup>1</sup> Faculdade de Zootecnia e Engenharia de Alimentos da Universidade de São Paulo;

<sup>2</sup> Programa de Pós-Graduação em Nutrição e Produção Animal da Faculdade de Medicina Veterinária e Zootecnia da Universidade de São Paulo

\* autor correspondente: jocajuni@usp.br

## 1. Introdução

Na produção agropecuária as decisões nem sempre são tomadas com base em intuições ou experiência é preciso uma análise mais elaborada do processo (Reijers, 2016). As análises tornaram-se mais elaboradas com a revolução computacional devido a capacidade de processar um maior volume de informações. Os computadores para compreender essas informações e servir de suporte a decisões exigem uma construção de um fluxo lógico e escrita em linguagem própria para executar tais comandos. Por isso, o objetivo deste estudo foi descrever um breve direcionamento das principais abordagens teóricas em linguagem de programação para a elaboração de modelos de simulação em sistemas integrados de produção agropecuária.

## 2. Desenvolvimento

### 2.1 Pensamento computacional

A lógica de programação foi um método desenvolvido para solucionar e compreender problemas utilizando conceitos da Ciência da Computação (Wing, 2006). A partir do conhecimento desse método torna-se possível transformar o computador, que a princípio é uma ferramenta de transferência de informação, em uma ferramenta de formulação de conhecimento (Wall, 2008). Esse método consiste na identificação das etapas para a solução do problema de forma eficiente e rápida e na instrução da máquina para executar os comandos. A máquina utiliza linguagem diferente da humana, então se faz obrigatório a tradução dessa informação de modo que a máquina realize as etapas para solução do problema, essa tradução chama-se programação (Blikstein, 2008).

Várias linguagens de programação foram desenvolvidas, no entanto, algumas delas tiveram mais êxito ao aproximarem os objetos do mundo real com os utilizados na programação surgindo, assim, a programação orientada a objeto. Foi permitido ao programador adicionar características, atributos e/ou detalhes ao que se pretendia com mais agilidade e os objetos puderam também interagir entre si, o que permitiu se envolverem em ações e/ou reações uns aos outros ou característica que podem ser agrupados em classes de semelhantes (Deitel e Deitel, 2003; Claro e Sobral, 2008). A linguagem Java é orientada ao objeto, teve sua primeira versão em 1996 e ganhou popularidade pelo fato de ser utilizada em navegadores da World Wide Web (WWW) e pela

sua simplicidade (Deitel e Deitel, 2003). Por ser uma linguagem que permite descrever as ações e manifestar as características dos objetos tornou-se uma ferramenta ideal para os modelos de simulação.

## 2.2 Modelos de simulação

A simulação é um processo de projetar um modelo computacional de um sistema real e conduzir experimentos para entender seu comportamento e/ou avaliar estratégias para sua operação (Pegden et al., 1995). Assim a simulação busca descrever comportamentos do sistema, construir teorias e hipóteses considerando as observações efetuadas e, usar o modelo para prever futuras alterações, isto é, efeitos produzidos por alterações no sistema ou no método empregado. Os métodos de simulação adequados para as observações do mundo real são os sistemas dinâmicos (SD), a simulação de eventos discretos (SED) e aquelas simulações baseada em agentes (SBA), nos quais cada um envolve o uso de diversas técnicas (Borshchev, 2013).

Modelos SD procuram descrever a dinâmica do sistema e a principal característica são as retroalimentações (*loops*); as variáveis assumem quaisquer valores dentro de um intervalo real (Borshchev, 2013). A SED possibilita que as variáveis de estado mudem somente quando ocorrer um evento em determinado período (Hillier and Lieberman, 2006). Um dos propósitos do SED é que os programadores considerem o sistema que está sendo modelado como um processo, em que uma sequência de operações está sendo desempenhado entre as entidades.

Recursos e entidades presentes em SD e SED, respectivamente, são passivas, isso é completamente diferente do que acontece em SBA (Borshchev, 2013). A individualidade dos agentes permite características diversas como a autonomia para comportamento mais apropriado, a interatividade com outros agentes e o ambiente e a adaptação durante o processo de simulação (Macal, 2016). Nesse método o nível de abstração é dependente do programador, podendo assumir quaisquer níveis.

Outra possibilidade é a mistura dos métodos de simulação propiciando os modelos híbridos de simulação (HS). Essa mistura tem um apelo forte para os programadores em Pesquisa Operacional, pois muito dos problemas do mundo real são complexos e com muitas características diferentes, por isso, raramente o uso de um único método seria o suficiente para propor modelo adequado.

## 2.3 Aplicações

As aplicações dos modelos de simulação são limitadas aos softwares que estão disponíveis no mercado, e vários são eles. No entanto, com capacidade de integrar os vários métodos em um único modelo passa a ser restrito. O AnyLogic é uma ferramenta multimétodo para a criação dos modelos. Esse software foi desenvolvido pela The AnyLogic Company e é líder no mercado de simulações dinâmicas e pioneiro na integração entre métodos de simulação (Gringryev, 2015).

Nas ciências agrárias diversos modelos de simulação têm sido desenvolvidos e cada um deles com os próprios vieses. Portanto, torna-se importante identificar modelos adequados que contemplem a produção agrícola e pecuária para avaliar e propor soluções em relação as condições climáticas, disparidades inter-regionais no clima, qualidade do solo, fatores sazonais ao

longo do tempo, disponibilidade de capital, entre outros (Weintraub e Romero, 2006). Alguns desses fatores são mensuráveis e podem ser quantificados a partir de informações históricas, experiências de produtores ou gestores agrícolas.

### 3. Conclusões

A linguagem de programação permite a conexão entre os sistemas reais e os modelos, tornando assim imprescindível para o programador que busca através dos modelos auxílio nas soluções, viabilizando as tomadas de decisões. Os modelos de simulação permitem o desenvolvimento de diversos cenários, assim auxiliam na análise de tecnologias, mudanças climáticas e dos indicadores. Quanto mais esses modelos tornam-se assertivos, melhor podem representar a execução de experimentos de campo e com isso minimizar o uso de recursos das instituições. Por fim, devido à importância deste assunto é relevante questionar sobre a formação dos profissionais das ciências agrárias quanto ao conhecimento de áreas das Ciências da Computação.

#### Referências bibliográficas

- BLIKSTEIN, P. **O Pensamento Computacional**, 2008. Disponível em: <[http://www.blikstein.com/paulo/documents/online/ol\\_pensamento\\_computacional.html](http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html)>. Acesso em: 10 out. 2019.
- BORSHCHEV, A. **The big book of simulation modeling : multimethod modeling with AnyLogic 6**. [s.l.] AnyLogic North America, 2013.
- BORSHCHEV, A.; FILIPPOV, A. **From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools**. In: INTERNATIONAL CONFERENCE OF THE SYSTEM DYNAMICS SOCIETY, 22.,2004, Oxford, England, **Proceedings...** Oxford: System Dynamics Society, 2004
- CLARO, D. B.; SOBRAL, J. B. M. **Programação em JAVA**. p. 89, 2008. Disponível em: <<http://www.carlosmatos.com.br/images/noticias/1157/arquivo.pdf>>.
- DEITEL, H. M.; DEITEL. P. J. **Java: Como programar**. 4. ed. Porto Alegre-RS: ARTMED, 2003.
- GRINGRYEV, I. **AnyLogic 7 in three days: a quick course in simulation modeling**. Hampton, NJ: AnyLogic North America, 2015
- HILLIER, F.S.; LIEBERMAN, G. L. **Introdução à pesquisa operacional**. 8. Ed. São Paulo: McGraw-Hill, 2006.
- MACAL, C. M. Everything you need to know about agent-based modelling and simulation. **Journal of Simulation**, v. 10, n. 2, p. 144–156, 02 outubro 2020. Disponível em: <<https://www.tandfonline.com/doi/full/10.1057/jos.2016.7>>.
- PEGDEN, C.D., Shannon, R.E., Sadowski, R.P., 1995. **Introduction to simulation using SIMAN**. McGraw-Hill, New York
- WALL, T. PAPERT , Seymour. **A máquina das crianças : repensando a escola na era da informática** . Tradução de Sandra Costa . Porto Alegre : Artes Médicas , 2008 . A máquina das crianças , numa escola com / sem futuro. p. 227–231, 2008.
- WEINTRAUB, A.; ROMERO, C. **Operations Research Models and the Management of Agricultural and Forestry Resources: A Review and Comparison**, INFORMS Journal on Applied Analytics, 36, 5, 446-457, 2006.
- WING, J. M. (2006). **“Computational Thinking”**. Communications of the ACM. March, Vol. 49, No. 13.