

Modelo Baseado em Equações *versus* Modelo Baseado em Agentes: uma abordagem usando sistema predador-presa

Diego A. Porcellis¹, Carlos Bertin¹, Marcilene Moraes¹

¹Programa de Pós-Graduação em Modelagem Computacional, PPGMC – FURG
Universidade Federal do Rio Grande

{diegoporcellis, marcilenemoraes}@furg.br, carlosbertin@gmail.com

Abstract. *This paper presents a comparative study between a equation-based model and the agent-based model, where we simulate the predator-prey system of Lotka-Volterra and the other using often techniques used in social modeling, to represent the behavior of agents in a given environment. In both approaches highlights the advantages and disadvantages of each, and finally we present a comparison between the results.*

Resumo. *Este artigo realiza um estudo comparativo entre um modelo baseado em equações e o modelo baseado em agentes, no qual um simula o sistema predador-presa de Lotka-Volterra e o outro utiliza técnicas frequentemente adotadas na modelagem social, para representar o comportamento dos agentes em um determinado ambiente. Em ambas as abordagens, evidencia-se as vantagens e desvantagens de cada um, e por fim é apresentado um comparativo entre os resultados encontrados.*

1. Introdução

Métodos de simulação tem se apresentado eficientes no auxílio da tomada de decisões, no planejamento de médio a longo prazo e em situações que envolvem custos e riscos elevados. O uso do computador em simulações é de grande valia, pois possibilita visualizar o comportamento de sistemas sem a necessidade de realização efetiva das modificações ou a construção, por vezes, de protótipos caros [REBONATTO, 2000].

Neste sentido, surge o conceito de Sistemas Multiagentes (SMA), sub-área da Inteligência Artificial Distribuída (IAD), no qual permite estudar “o comportamento de um conjunto de agentes autônomos, eventualmente com características diferentes, evoluindo em um ambiente comum. Estes agentes podem interagir uns com os outros, com o objetivo de realizar suas tarefas de modo cooperativo, compartilhando informações, evitando conflitos e coordenando a execução de atividades” [Alvares e Sichman *apud* ADAMATTI 2011]. A integração entre tecnologias de agentes e métodos de simulação, permite analisar o cenário em estudo de uma forma mais realista.

O modelo baseado em agentes, que simula um sistema predador-presa tem por objetivo demonstrar a cooperação em sistemas multiagentes (entre os predadores) e, por outro lado, a atuação independente da presa. Assim, é possível explorar o comportamento social dos agentes no meio em que estão inseridos. Os agentes predadores devem identificar a presa, para então iniciar a sua perseguição e, conseqüentemente, a sua captura [MOMM 2012].

Para representar a interação entre presa e predadores, os estudiosos Lotka e Volterra elaboraram independentemente, o sistema predador-presa baseado em equações, chamado de modelo Lotka-Volterra. Trata-se de um par de equações diferenciais, não lineares e de primeira ordem, utilizadas para descrever dinâmicas de sistemas biológicos, mais especificamente quando duas espécies interagem: uma como presa e outra como predadora. A lógica e a teoria matemática sugerem que quando as presas são numerosas, seus predadores aumentam em número, reduzindo assim a população de presas, que por sua vez faz com que o número de predadores decline. A população de presas eventualmente se recupera, iniciando um novo ciclo [NADELHOFFER, 2005].

Por meio de um efetivo estudo do modelo de Lotka-Volterra, um modelo baseado em equações (MBE), pode-se extrair características comportamentais do sistema predador-presa. Com o enfoque do trabalho voltado para o comportamento social do modelo, criou-se um modelo baseado em agentes, adotando técnicas comumente utilizadas na modelagem social. No MBE a simulação é implementada através de um método matemático, utilizando equações diferenciais inter-relacionadas, tornando o modelo mais abstrato, com agentes homogêneos, apresentando resultado global, a longo prazo. Já no modelo baseado em agentes (MBA), os agentes são heterogêneos, com comportamento individualizado, permitindo assim retratar melhor a vida real.

Com intuito de simular a dinâmica interativa dos agentes, utilizou-se o ambiente de programação Netlogo. Uma de suas principais características é que ele realiza simulações voltados para fenômenos naturais e sociais.

O objetivo deste estudo é discutir as vantagens e desvantagens dos modelos MBE e MBA, utilizando o ambiente de programação Netlogo e tendo como tema o modelo de simulação predador-presa. O artigo está dividido em seções a seção 2 conceitua e distingue o MBE do MBA, a seção 3 define o modelo de Lotka-Volterra, a seção 4 demonstra a ferramenta NetLogo, a seção 5 aborda sobre a implementação realizada, na seção 6 são feitas as discussões sobre os resultados encontrados e o trabalho é concluído na seção 7.

2. MBE e MBA

O MBE e o MBA são duas técnicas de modelagem de sistemas sociais difundidas. A MBA opõe-se às abordagens tradicionais de simulação que geralmente são realizadas através das MBE, onde são construídas a partir de conjuntos de equações diferenciais inter-relacionadas [TANG, PARSONS e SKLAR 2006].

O MBA como técnica de modelagem de sistemas sociais está sendo estimulada pelo crescimento dos sistemas computacionais e é cada vez mais aplicada a sistemas anteriormente modelados com equações diferenciais [RAHMANDAD e STERMAN 2004]. Neste modelo, pode-se inserir um comportamento difuso para cada agente no sistema [TANG, PARSONS e SKLAR 2006]. Já os MBE são mais abstratos que MBA, pois eles representam o sistema como um todo através de uma média resumida da possibilidade de ação de determinado agente [COSTA, JEANNES e CAVA 2008].

No MBE, existem leis globais que se aplicam a todos os membros do estudo tornando os agentes homogêneos, tratados como iguais, assim eles podem não ser tão realistas devido a essa homogeneidade, no entanto, ele tem como vantagem o fato de

poder ser manuseado analiticamente. Entretanto nos últimos anos, os modelos vêm se tornando “mais realistas” e complexos tornando-se difícil de manusear analiticamente [NGUYEN et al 2012].

O MBE retrata bem os resultados a longo prazo sem a necessidade de realizar experimentos simulados [NGUYEN et al 2012]. Porém, fica difícil de visualizar individualmente o comportamento dos agentes e também o comportamento destes dentro de um ambiente. Com o MBA consegue-se visualizar os agentes individualmente, as interações entre eles e também simular interações entre agentes e o ambiente onde estes estão inseridos. Devido a isso o MBA é considerado mais flexível que o MBE [TANG, PARSONS e SKLAR 2006].

Então pode-se dizer que a escolha pelo uso do MBA ocorre por quatro principais motivos [TANG, PARSONS e SKLAR 2006]:

- (I) É uma maneira natural de descrever sistemas compostos e interação entre entidades;
- (II) São flexíveis;
- (III) Facilidade de capturar fenômenos emergentes;
- (IV) Fornecem maior nível de detalhes úteis em relação a simulação.

Outra vantagem que pode-se citar no uso do MBA é sua simplicidade na criação, facilidade de entendimento e, portanto, é mais acessível a modificações e ampliações tornando assim o modelo cada vez mais realista e detalhista [TANG, PARSONS e SKLAR 2006]. Alguns autores argumentam que o MBE e MBA são mais produtivos quando juntos em vez de serem consideradas como incompatíveis [COSTA, JEANNES e CAVA 2008].

3. Modelo de Lotka-Volterra

O estudo de dinâmicas populacionais é essencial para o entendimento do que ocorre nos ecossistemas. Populações de duas espécies diferentes podem se relacionar de diversas maneiras. Essas relações podem ser consideradas harmônicas, quando há benefício por pelo menos uma das espécies, ou podem ser desarmônicas, quando há prejuízo para uma ou ambas as espécies [ODUM, 1988]. A predação é um exemplo de relação que resulta em efeitos negativos no crescimento e sobrevivência de uma população e em um efeito positivo na outra.

Uma das ferramentas mais importantes na compreensão destas relações é a modelagem matemática. Para fazê-la, é necessário identificar características fundamentais do sistema a ser estudado, de maneira a obter um conjunto de regras matemáticas simples o suficiente para que se possa extrair informações úteis. Dependendo da natureza do problema, estas regras podem ser instituídas de diversas formas, através de equações diferenciais, equações algébricas, equações de diferenças, etc.

Alfred Lotka (1925) e Vito Volterra (1926) foram pioneiros em estudar interações entre presas e predadores. O modelo de Lotka-Volterra, mesmo com sua simplicidade, é muito representativo. Através de duas equações diferenciais ele consegue descrever o comportamento de dois tipos diferentes de populações, a presa e o predador.

Como citado anteriormente, o modelo Lotka-Volterra consiste em um par de equações diferenciais, e que frequentemente são utilizadas para representar a dinâmica populacional de sistemas biológicos. Segundo Malaquias e Mizukoski apud SOUZA (2007), o modelo de Lotka-Volterra é dado pelas equações (1) e (2):

$$\begin{cases} \frac{dx}{dt} = ax - cxy & (1) \text{ (Corresponde a dinâmica da presa)} \\ \frac{dy}{dt} = -bx + dxy & (2) \text{ (Corresponde a dinâmica do predador)} \end{cases}$$

onde:

(I) x é a densidade de presas;

(II) y é a densidade de predadores;

(III) a é a taxa de nascimento das presas;

(IV) b é a taxa mortalidade dos predadores;

(V) c é a taxa de mortalidade das presas devido a relação com os predadores, esta também é denominada de coeficiente de predação;

(VI) d é a taxa de reprodução dos predadores por unidade de presas consumidas;

(VII) t representa o crescimento das duas populações ao longo do tempo.

Analisando as equações (1) e (2), é notável que cada encontro entre presas e predadores tende a inibir o crescimento da população de presas, pois representa sua captura, o que sugere o aumento de sua taxa de mortalidade, e ainda resulta no aumento do número de predadores, já que sua reprodução está relacionada ao consumo.

4. NetLogo

A ferramenta escolhida para simular MBA foi o NetLogo (Figura 1), que é uma linguagem de programação agregada a um ambiente de desenvolvimento multiagente utilizada para simulações de fenômenos naturais e sociais. O NetLogo tem como vantagem os códigos que são simples e legíveis além de proporcionar uma interface intuitiva onde, pode-se inserir *sliders*, *plotters*, botões de ação e de escolha para controlar variáveis e parâmetros. Ainda pode-se visualizar graficamente os agentes interagindo entre si e com o ambiente.

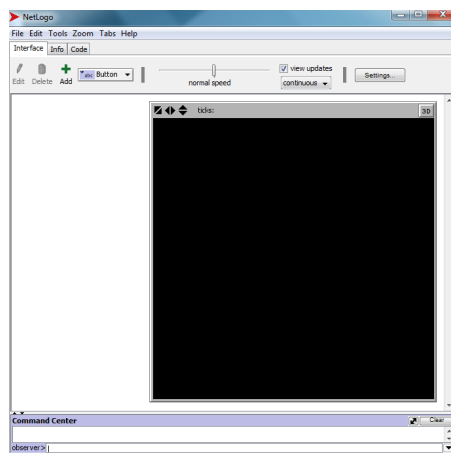


Figura 1. Interface do NetLogo

O ambiente do NetLogo é dividido em três abas (Figura 1): *Interface*, *Info* e *Code* cada aba tem suas próprias particularidades. A aba responsável pela criação da parte gráfica é a *Interface*. Nessa aba pode-se criar todo o ambiente de simulação.

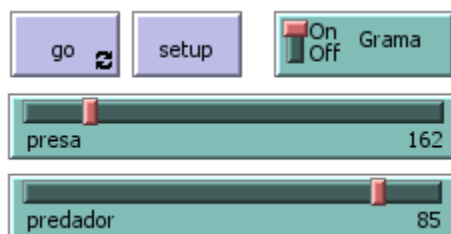


Figura 2. Exemplos de botões (*buttons*), chaves (*switches*), parâmetros (*sliders*).

Os itens que podem ser inseridos na interface são:

- (I) *Button*: (Figura 2) responsável pelas ações dentro de nossa simulação;
- (II) *Slider*: (Figura 2) usados para definir valores de parâmetros e variáveis do sistema;
- (III) *Switche*: (Figura 2) são chaves de liga e desliga;
- (IV) *Monitor*: (Figura 3) usado para monitorar valores textualmente em tempo de simulação;
- (V) *Plots*: (Figura 3) usado para visualizar o estado de uma variável ao longo do tempo gerando gráficos de evolução.

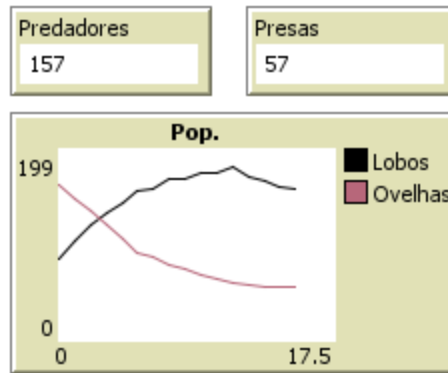


Figura 3. Exemplos de monitor (*monitor*), gráficos (*plot*).

Após a criação da interface, deve-se codificar a simulação. Esta etapa é realizada através de linguagem de programação. Os códigos são determinantes, pois são eles que ditam como cada agente irá se comportar no ambiente. Essa etapa é realizada na aba Code (Figura 4).

```

File Edit Tools Zoom Tabs Help
Interface Info Code
Find... Check Procedures - Indent automatically

breed [lobo lobo]
breed [ovelha ovelha]
lobo-own [ovelhas-in-cone]

to setup
  ca
  ask patches [set pcolor green
  if Grama [
    if (pxcor > 0) and (pycor > 0) [ set pcolor lime ]
  ]

  ask n-of presa patches[
    sprout 1[
      ask ovelhas [set shape "sheep"]
      set breed ovelhas
      set size 4
      rt random 360
      fd random 5
      set color red
    ]
  ]
]

```

Figura 4. Exemplos de codificação no NetLogo.

A aba *Info* é responsável pela documentação do modelo criado. Nela podem ser inseridas as rotinas executadas no modelo, a finalidade, explicações sobre como usá-lo, futuras ampliações, entre outros.

5. Implementação

O modelo realizado, usando a linguagem NetLogo, será um MBA que simula as regras do modelo de Lotka-Volterra que é um MBE. Porém, o método de resolução desse modelo difere do método de resolução aplicado ao modelo de Lotka-Volterra. Enquanto no modelo de Lotka-Volterra é aplicado um método matemático como o método de Euler, o modelo criado se dá através da iteração entre os agentes ao longo do tempo, dentro de um ambiente pré-determinado.

Essa simulação busca representar o modelo de Lotka-Volterra pois, como visto anteriormente as equações de Lotka-Volterra são apenas em função do tempo sem nenhuma relação com espaço. E o NetLogo usa agentes como pontos dentro de um

espaço para representar o seu comportamento. Logo, o espaço é parte integrante da simulação e tem um papel importante para o resultado final.

A simulação tem como divisão temporal o *tick*, que é um intervalo de tempo dentro do ambiente de desenvolvimento NetLogo. A cada *tick*, os agentes irão realizar suas ações. Essas ações são predeterminadas no momento do desenvolvimento do código.

5.1. Agentes

Os agentes da simulação são divididos em duas classes: as presas e os predadores. Cada um deles terá sua atribuição no sistema conforme exposto abaixo:

5.1.1. Predador

Os predadores realizam basicamente duas funções: `caçarPresas` e `morrePredador`.

Na função de `caçarPresas`, o predador busca uma presa dentro de seu campo de visão e ao encontrá-la, ele começa a perseguição até estar sobre a presa, o que ainda não garante sua captura, pois o predador realiza um sorteio randômico de dois valores e se esses valores estiverem dentro do intervalo escolhido nas variáveis $k3$ (taxa de predação) e $k4$ (eficiência de predação), ele a captura e gera um novo predador em *patch* aleatório.

A morte dos predadores acontece ao chamar a função `morrePredador` que realiza o sorteio de um número randômico e se esse valor estiver dentro dos valores estipulados na interface na variável $k2$ (taxa de mortalidade de predadores), ocorre a morte do agente atual.

5.1.2. Presa

A presa realiza duas funções a `nascePresas` e a `movePresas`.

Na função `nascePresas` é feito um sorteio de um número randômico e se esse número estiver dentro do intervalo escolhido na variável $k1$ (taxa de nascimento de presas) é criada uma nova presa dentro de nosso sistema num *patch* aleatório.

Já na função `movePresas` é escolhido randomicamente uma direção qualquer e a presa anda um passo nessa direção.

5.2. Interface

A interface da simulação (Figura 5) é onde pode-se selecionar os parâmetros iniciais da simulação, como o número de presas iniciais, número de predadores iniciais, a taxa de nascimento de presas, a taxa de mortalidade dos predadores, a taxa de predação de lobos e a taxa de eficiência de predação. Além da seleção de parâmetros iniciais, também é possível visualizar o gráfico da população de presas e predadores, a quantidade textual atual de presas e predadores e uma interface gráfica mostrando o comportamento dos agentes dentro do ambiente.

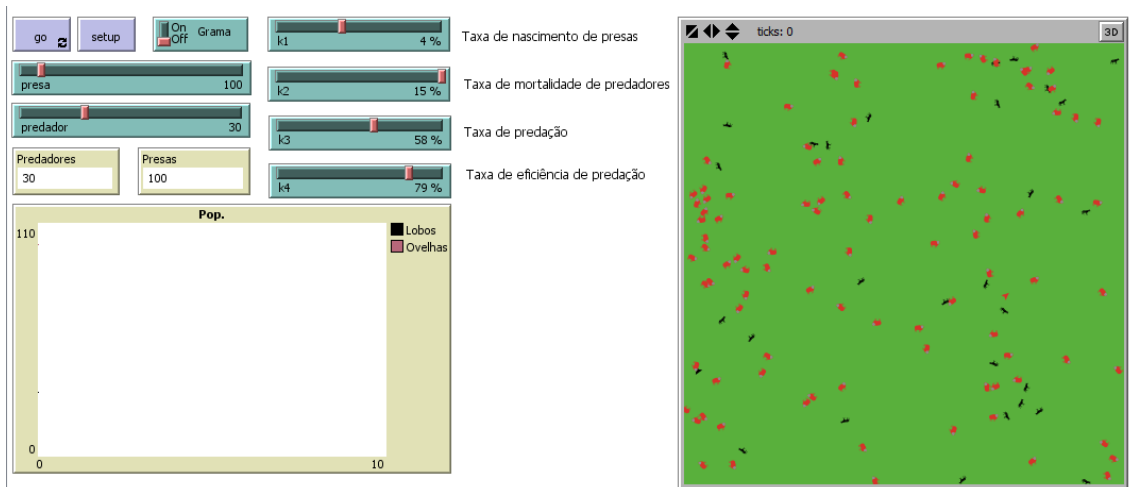


Figura 5. Interface do modelo Predador-Presa com a opção de grama desligada.

5.3 Patch

O *Patch* é cada área do terreno ocupado ou não por agentes, nesta simulação tem-se um ambiente de 201x201 *patches*. Inicialmente todos os *patches* são verdes e não interferem no comportamento dos agentes (Figura 5). Num segundo momento criam-se *patches* diferentes e assim modifica-se o comportamento dos agentes quando estiverem dentro destes *patches*. Essa modificação ocorreu em alguns *patches* que tinham uma cor mais escura e foi considerado que, dentro desses *patches*, os predadores teriam uma visão prejudicada como se fosse, por exemplo, uma vegetação fechada (Figura 6).

Esta funcionalidade foi inserida através da chave *Grama*. Quando a chave estiver ligada alguns dos *patches* serão mais escuros e modificarão o comportamento da caça dos predadores.

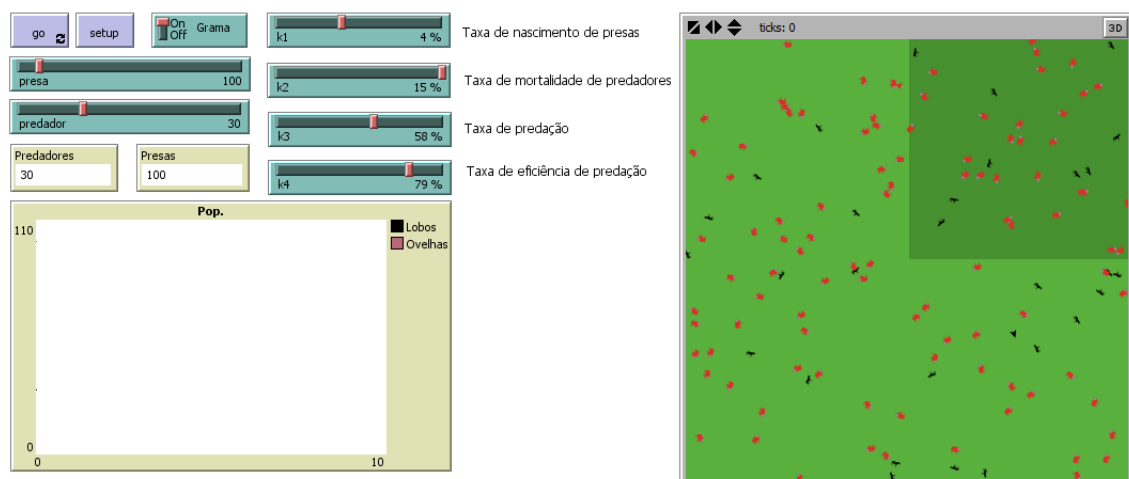


Figura 6. Interface de simulação Predador-Presa com a opção de grama ligada.

6. Resultado e Discussões

Os resultados obtidos através das simulações foram comparados com os resultados do sistema dinâmico obtidos através do método de Euler. Na figura 7 tem-se o resultado da simulação usando o MBA criado com NetLogo sem a opção *Grama* ligada. Enquanto que na figura 8 tem-se o resultado do MBE usando o método de Euler para a resolução do sistema de equações de Lotka-Volterra formado pelas equações (1) (2).

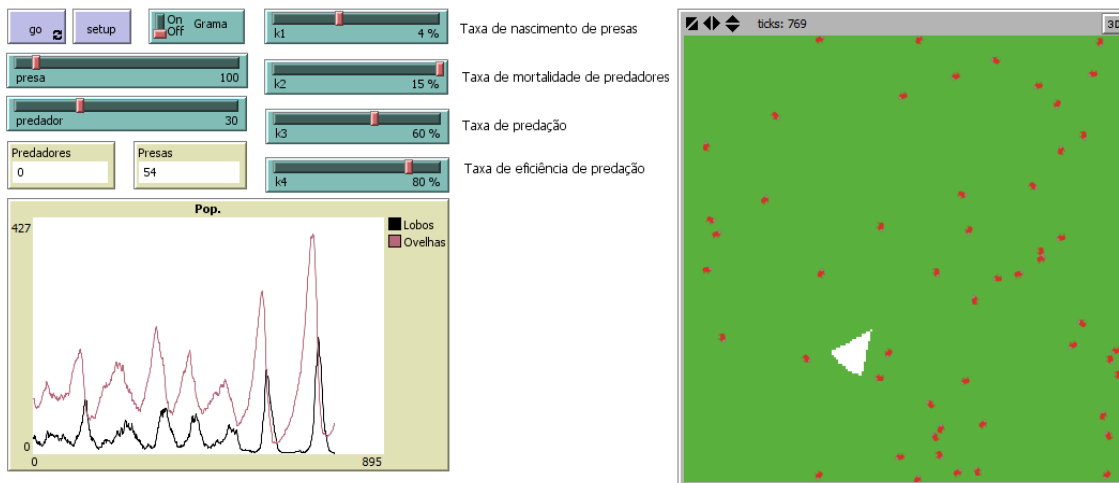


Figura 7. Simulação sem a utilização de ambiente diferenciado.

É importante verificar que o comportamento de ambas as simulações seguiu um padrão em relação às linhas de presas e predadores (na simulação lobos e ovelhas) ao longo do tempo. Apesar das pequenas mudanças ainda existe uma sinuosidade diferenciada que pode ter sido influenciada pela não previsibilidade de encontros e movimentos de presas e predadores dentro do ambiente delimitado que não é considerado quando trabalhamos com equações.

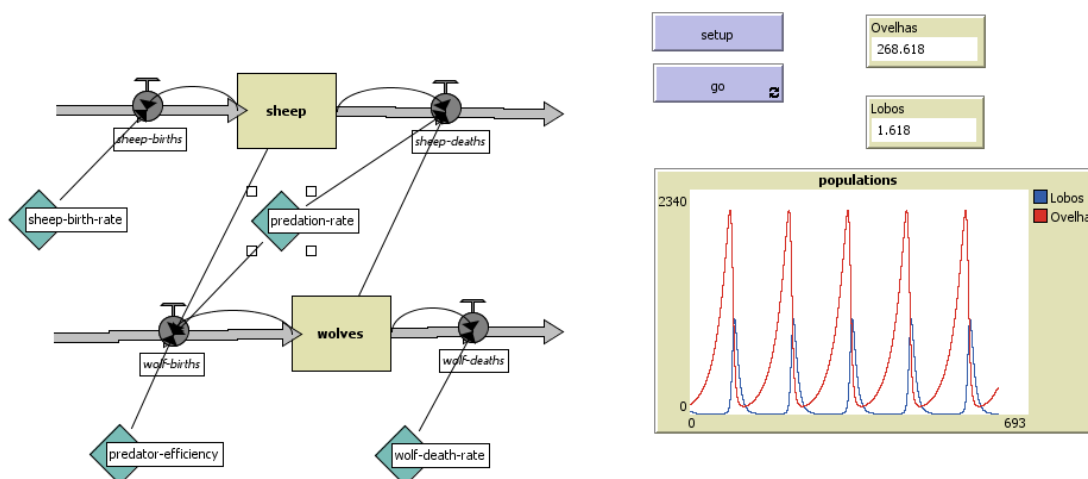


Figura 8. Simulação usando método de Euler.

Na simulação realizada com um ambiente diferenciado, onde uma porção do ambiente com vegetação fechada é evidenciado (Figura 9), pode-se ver também a semelhança entre os gráficos da simulação e pelo método de Euler, apesar de os resultados encontrados seguirem sempre um padrão no MBA, tem-se algumas modificações relacionadas ao ambiente.

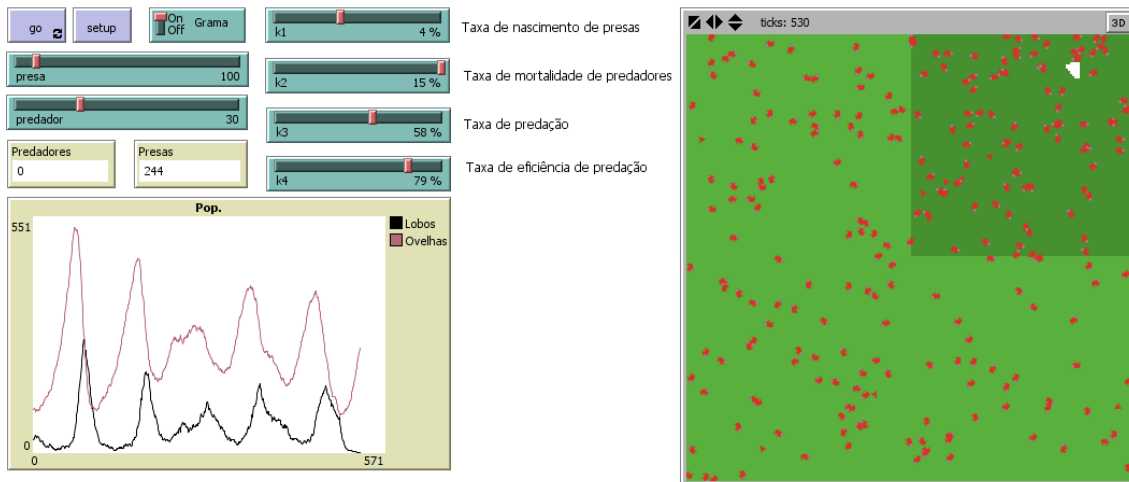


Figura 9. Simulação com a utilização de ambiente diferenciado.

Dentro das simulações baseadas em agentes consegue-se visualizar com mais perfeição alguns detalhes úteis. Um desses detalhes, vistos nos exemplos é que quando tem-se uma área onde a visão do predador é reduzida e a concentração de presas nesse ambiente aumenta (Figura 10) consegue-se verificar detalhes que não são possíveis quando utiliza-se MBE na simulação.

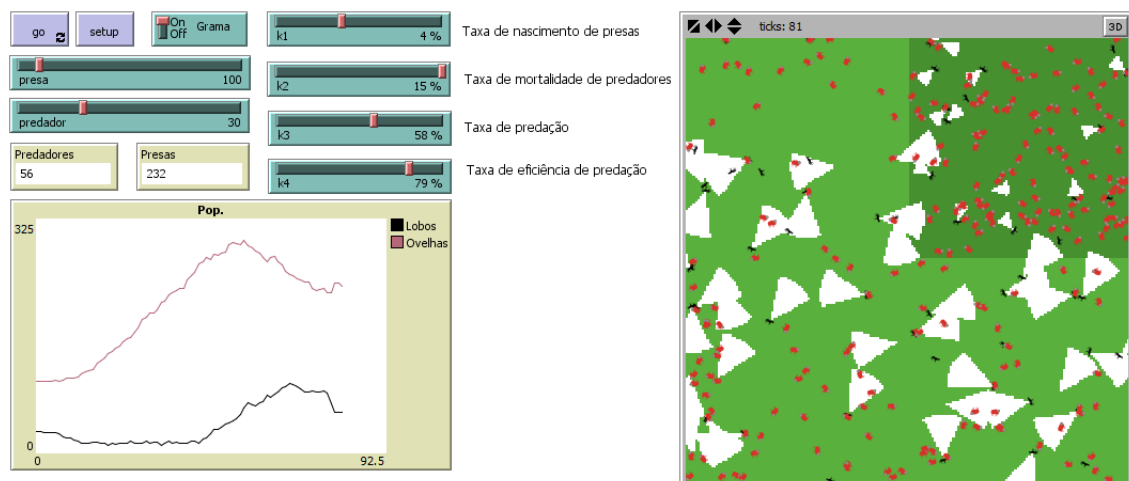


Figura 10. Simulação com a utilização de ambiente diferenciado.

Com esse exemplos, verifica-se que o MBA tem vantagens quando necessita-se de detalhes sobre a simulação, quando é preciso que a modelagem evolua com o tempo e inserção de novas variáveis e novos detalhes. Porém, nota-se que o resultado gráfico das simulações acaba sempre seguindo um mesmo padrão, o que nos leva a concluir que

o MBE consegue criar o comportamento geral da simulação, sem a necessidade da criação do modelo e inserção detalhada do funcionamento.

7. Conclusão

Ao avaliar o modelo predador-presa a partir de uma visão analítica (equações) e do modelo baseado em agentes, conseguiu-se observar o comportamento resultante das interações entre eles, foi possível identificar de forma clara que as oscilações dependiam dos parâmetros e o quanto isso influenciava no modelo. Constatou-se que o modelo de simulação multiagente é bastante simples pois, sua criação é realizada através da visualização do ambiente e descrição dos comportamentos dos agentes envolvidos. Por outro lado, os de equações são mais complexos devido ao grande estudo matemático envolvido.

Também é possível observar que o MBE gera uma resposta mais abstrata, devido aos seus resultados serem para um cenário mais geral e utilizado para qualquer situação predador-presa em qualquer ambiente, já no MBA foi possível inserir mais detalhes da simulação, como o caso da vegetação mais fechada, a limitação da visão do predador e a forma de caça. Mesmo com essas diferenças os resultados encontrados foram muito semelhantes.

Dessa forma acreditasse que a escolha pelos modelos sejam baseados em equações ou em agentes depende do quão detalhado deseja-se o resultado final da simulação. Por fim, ressalta-se que é possível que as duas formas trabalhem associadas na busca por um resultado geral do MBE aliado ao detalhamento do MBA.

Referências

- Adamatti, Diana. (2011) Simulação Baseada em Multiagentes como Ferramenta em Estudos Interdisciplinares. *RENOTE*, v. 9, n. 1.
- Costa, Antônio CR; Jeannes, Fernanda M.; Cava, Ulisses A. (2010) Equation-based models as formal specifications of agent-based models for social simulation: preliminary issues. In: *Social Simulation (BWSS), 2010 Second Brazilian Workshop on*. IEEE, p. 119-126.
- Momm, Deise R. C. (2012) Desenvolvimento de Robótica Inteligente em Ambiente Virtual para Múltiplos Robôs, Relatório Anuais, Departamento de Engenharia Elétrica, Puc-Rio, Rio de Janeiro.
- Nadelhoffer, K. J. (2005) Predation and Parasitism. <http://www.globalchange.umich.edu/globalchange1/current/lectures/predation/predation.html>, 2005. Acesso em: 15/10/2014.
- Nguyen, N. D., Taillandier, P., Drogoul, A., & Auster, P. (2012). Inferring equation-based models from agent-based models: a case study in competition dynamics. In *Principles and Practice of Multi-Agent Systems*(pp. 413-427). Springer Berlin Heidelberg.
- Odum, Eugene P. (1988) *Ecologia*. Rio de Janeiro: Guanabara.

- Parunak, H. Van Dyke; Savit, Robert; Riolo, Rick L. (1998) Agent-based modeling vs. equation-based modeling: A case study and users' guide. In: Multi-agent systems and agent-based simulation. Springer Berlin Heidelberg. p. 10-25.
- Rahmandad, Hazhir; Sterman, John. (2008) Heterogeneity and network structure in the dynamics of diffusion: Comparing agent-based and differential equation models. Management Science, v. 54, n. 5, p. 998-1014.
- Rebonatto, M. T. (2000) Simulação paralela de eventos discretos com uso de memória compartilhada distribuída. Master's thesis, Programa de Pós-Graduação em Computação, UFRGS, Porto Alegre, 2000. Mestrado em Ciência da Computação.
- Souza, L.A. (2007) Sustentabilidade da pesca através da inclusão do homem em modelos predador presa: Um estudo de caso no Lago Preto. Dissertação (Doutorado em Ciências Biológicas). Programa de Pós-graduação e Biologia Tropical e Recursos Naturais do convênio INPA/UFAM.
- Tang, Yuqing. Parsons, Simon. Skalar, Elizabeth. (2006) Modeling human education data: From equation-based modeling to agent-based modeling Proceedings of the Workshop on Multiagent-based Simulation, Hakodate, Japan.