



## NETPLAY: um ambiente para simulação multiagente

Lívia Lopes Azevedo<sup>1,2</sup>,

Alexandre Andrade<sup>2</sup>,

Diego Velasco<sup>2</sup>

Lucio Barbiero<sup>2</sup>

Davidson Cury<sup>2</sup>

Crediné Silva de Menezes<sup>2,3</sup>

<sup>1</sup>Instituto de Ciências e Letras do Médio Araguaia (ICLMA) – UFMT

<sup>2</sup>Centro Tecnológico – PPGEE – Universidade Federal do Espírito Santo (UFES)

<sup>3</sup>Programa de Pós-Graduação em Informática na Educação (PGIE) – UFRGS

[livia@cpd.ufmt.br](mailto:livia@cpd.ufmt.br), [aleandradel@yahoo.com](mailto:aleandradel@yahoo.com),  
[diegocvelasco@gmail.com](mailto:diegocvelasco@gmail.com), [lm\\_barbiero@yahoo.com.br](mailto:lm_barbiero@yahoo.com.br),  
[dede@inf.ufes.br](mailto:dede@inf.ufes.br), [credine@inf.ufes.br](mailto:credine@inf.ufes.br)

### Resumo.

*Neste trabalho apresentamos o ambiente NetPlay, desenvolvido para a criação de modelos de simulação baseada em multiagente voltados para o estudo de sistemas complexos. O NetPlay foi desenvolvido para fins didáticos, sendo um dos seus objetivos a facilidade de utilização, no intuito de possibilitar e estimular os aprendizes a criar seus próprios modelos de simulação.*

**Palavras-chave:** Ambiente de modelagem, sistemas complexos, simulação multiagente.

### Abstract.

*In this work we present the NetPlay environment, developed for the creation of models of simulation based in multi agent address for the study of complex systems. The NetPlay was developed with didactic objective, being one of its goals the easiness of use, in the intention of to make possible and to stimulate the students to create their own models of simulation.*

**Keywords:** Environment of modeling; complex systems; multiagent simulation.

### Introdução

Um modelo de simulação, de modo geral, consiste em reproduzir artificialmente uma situação ou fenômeno natural. Os modelos de simulação se baseiam na idéia de que os programas exibem o comportamento do sistema real e desta forma, ao transportar um modelo do mundo real para um artificial, na qual hipóteses particulares podem ser exploradas, possibilita o processo de resolução de problema como um ambiente de aprendizagem. A simulação pode ser usada para realização de experimentos de maneira análoga à que realizamos em um laboratório real.

A modelagem baseada em agentes é um tipo de simulação, caracterizada pela existência de muitos agentes interagindo uns com os outros, com pouca ou nenhuma coordenação central. Através dela, é possível observar como os agentes individuais (pessoas, bactérias, insetos, nações, ou organizações) interagem entre si e com seu ambiente e, a partir daí, ter um melhor entendimento sobre os comportamentos dos

sistemas – sociedades artificiais. É necessário dispor de ambientes computacionais que possibilitem a criação dessas simulações, que, normalmente, envolvem fenômenos complexos e modelos descentralizados.

Complexidade é um tema associado às matérias tradicionais do currículo educacional, apesar de raramente ser estudado nas disciplinas até o ensino médio, possivelmente pelo seu caráter não-linear. Muitos estudos têm mostrado que tanto no público em geral quanto as disciplinas de ciências não é fácil encontrar bons pensadores sobre sistemas compostos de agentes autônomos que, para realizar suas finalidades, precisam interagir com os outros e com o ambiente (Resnick, 1991). O sistema escolar atual, em seus vários níveis, ainda repete uma prática tradicional de ensino que enfatiza a memorização de fórmulas, esquemas, classificação e teorias estabelecidas. Essa classificação pode levar os alunos a aprender os nomes dos ossos do corpo humano, os nomes e formas de diferentes folhas e plantas, ou filo no reino animal, mas isto é muito pouco para o domínio dos fenômenos naturais e sociais. Mesmo na universidade, onde os conteúdos se ampliam para incluir fenômenos não visto nos ciclos anteriores, o processo de memorização e classificação continua essencialmente o mesmo.

Este quadro contrasta nitidamente com o aquele que emerge da proposta apresentada nos PCN's – Parâmetros Curriculares Nacionais (1998). Nessa proposta os alunos devem ser participantes ativos do processo de aprendizagem. Ao invés de aceitar as informações como prontas e acabadas, eles devem ser levados a questionar os processos, admitindo como teorias provisórias que constantemente são reavaliadas e reconstruídas levando em conta o diálogo entre teoria e evidência. Esta avaliação envolve frequentemente raciocínio por uma gama de níveis – o processo investigativo.

Tentando estreitar esta distância e buscando um uso efetivo das tecnologias e acreditando que através da exploração de modelos computacionais os aprendizes passam a ter uma atitude mais efetiva na construção do seu próprio conhecimento, estamos desenvolvendo o NetPlay, um ambiente para construção de modelos suportado por simulação multiagente, apropriado para o trabalho escolar, no âmbito dos sistemas complexos.

A construção do modelo de simulação pode ser usada em diferentes momentos do trabalho em sala de aula. Após a apresentação de um conteúdo, servindo de idéia desencadeadora, tanto para lançamento de questões que desafiem o conhecimento dos alunos como para identificação de conceitos que ali estão presentes. Como elemento interdisciplinar, numa abordagem baseada em projetos, entre outros. Um aspecto a ser considerado é a participação cooperativa entre grupos de alunos para o desenvolvimento do modelo, cujo processo de desenvolvimento deve oferecer liberdade para o lançamento de idéias, levantamento de dúvidas e sugestões. As hipóteses levantadas deverão ser testadas na simulação, produzindo novos dados e/ou adequação do modelo. A participação do professor neste processo é a de mediador, o objetivo é despertar no aluno a curiosidade sobre os dados e o desejo de buscar a compreensão dos mesmos.

Este artigo faz a apresentação de um ambiente que visa atender essas necessidades. A sua organização está na seguinte forma: o domínio do problema é apresentado no tópico 2, seguido com o tópico que descreve sobre “simulação multiagente”. No tópico 4 apresentamos alguns ambientes de simulação multiagente existentes na literatura. O tópico 5 apresenta a concepção da ferramenta, descrevendo suas características. A apresentação da implementação e sua utilização é realizada no tópico 6, seguido das considerações e referências bibliográficas.

## 2. Sistemas complexos

Complexidade é o termo empregado para o estudo de sistemas nos quais fenômenos e comportamentos surgem de interações de componentes simples. Sussman (2000) catalogou definições sobre complexidade de vários pensadores em diferentes áreas. Apesar das diferentes áreas, existe um sentido que permeiam todas elas. Sistemas complexos podem ser identificados ao exibir organização sem um princípio organizacional centralizado e também pela forma que eles podem ou não ser analisados. Ao decompor um sistema e analisar suas subpartes, os resultados, não necessariamente, dão indício sobre o comportamento do todo. A complexidade estuda o comportamento conjunto de milhares ou milhões de agentes trocando informações e compondo o que se chama de “sistema adaptativo complexo” (Ruthen 1993). Um sistema adaptativo complexo consiste de muitos elementos – aqui chamados de agentes, cada um interagindo com outros agentes, tendo as mesmas ou diferentes estratégias, e com o seu próprio ambiente. A escola convencional reforça a hipótese dos sistemas lineares formando assim adultos com dificuldades para compreender os fenômenos naturais e sociais, posto que, estes, em sua maioria são dificilmente explicáveis com o pensamento centralizado (Resnick, 1991).

São exemplos de sistemas complexos: uma colônia de formigas, mercado financeiro, uma cidade, bando de pássaros voando, reações do sistema imunológico, congestionamento de trânsito ou o funcionamento do cérebro humano. Em todos esses sistemas não há controle centralizado e os modelos são determinados pelas interações locais entre os componentes descentralizados (Resnick, 1991). Esses exemplos mostram que a coletividade possui características que não podem ser definidas pelos componentes individuais, mas esses são essenciais para o comportamento bem adaptado que tais sistemas apresentam. Nestes exemplos, todos são sistemas de muitos agentes, cada um interagindo com seus vizinhos e, mais importante, adaptando-se às mudanças (Ruthen, 1993).

A teoria da complexidade é um novo campo da ciência que estuda até que ponto as partes de um sistema se separam do comportamento coletivo, e como o sistema interage com seu ambiente. Sistemas complexos são difíceis de entender porque as causas e os efeitos não estão relacionados de forma direta, ou seja, existem correlações entre eventos distantes no tempo e no espaço. Analisar os modelos à luz da complexidade pode fornecer uma importante oportunidade para examinar problemas relacionados às ciências sociais e cognitivas de modo mais preditivo.

### **3. Simulação multiagente e Sistemas Complexos**

A área de sistemas multiagentes (SMA) estuda o comportamento de um grupo organizado de agentes autônomos que cooperam entre si para realizar atividades de resolução de problemas que estão além das suas capacidades individuais. A modelagem baseada em agentes tem-se mostrado uma interessante abordagem para investigar os sistemas complexos, pois segundo (Drogoul, Ferber, 1992) a simulação multiagentes consiste, geralmente, em reproduzir artificialmente um fenômeno natural. Construir modelos a partir de um agente permite as pessoas assistirem o comportamento de vários agentes (pessoas, bactérias, insetos, nações, ou organizações), que, seguindo regras simples, interagem entre si e com seu ambiente. Isto permite observar de que forma os padrões do sistema, como o fluxo de pedestres e congestionamento de trânsito, surgem independente dos comportamentos individuais e, a partir daí, ganhar um melhor entendimento sobre os comportamentos emergentes.

Utilizando a simulação por computador acreditamos fazer uso efetivo das tecnologias, através da exploração de modelos computacionais, de modo a levar os

aprendizes a uma atitude proativa na construção de seu conhecimento. Dentre as várias técnicas de simulação destacamos a modelagem baseada em agentes (“*agent-based modeling*”). Esse tipo de simulação é caracterizado pela existência de muitos agentes interagindo uns com os outros, com pouca ou nenhuma coordenação centralizada. A propriedade emergente do modelo baseado em agentes é que ela vem do baixo para cima (“*botton-up*”) e não do sentido de cima pra baixo (“*top-down*”). Nessa perspectiva é que vamos trabalhar com modelos de simulação multiagentes.

#### 4. Ambientes de simulação multiagente

Existem vários ambientes de programação especificamente projetados para modelagem baseada em agentes<sup>1</sup>, destacamos: StarLogo, Swarm, Repast, AgentSheets, NetLogo e SIMULA. Convém citar também o ambiente Klik&Play, que apesar de ter sido desenvolvido para a criação de jogos apresenta características suficientes para a modelagem multiagente, ainda que, as implementações conhecidas só permitam uma quantidade reduzida de agentes. Em (Azevedo et. al., 2005) está apresentado uma descrição detalhada sobre cada um desses ambientes. Esses ambientes de modelagem multiagentes estão revolucionando a prática científica, até então utilizada para entender sistemas complexos do Universo (Abrahamson, Wilensky, 2005; Blikstein, Wilensky, 2005).

##### 4.1 Considerações sobre os ambientes

Cada um dos ambientes citados segue abordagens bem distintas para descrição de modelos de simulação. Dentre os ambientes citados os que oferecem maior gama de possibilidades e algoritmos são Swarm (Swarm Development Group. 1999) e o Repast (Collier, Howe, North, 2003), que por sua vez emprega todos os conceitos do Swarm. Porém seu uso depende de muito conhecimento em programação. O NetLogo (Wilensky, 1999) é um ambiente programável de fácil apropriação, pois sua linguagem é um dialeto Logo estendido para suportar agentes. Os ambientes AgentSheets (Repenning, 1993) e Simula (Frozza, 1999), com suas características próprias, utilizam uma linguagem mais voltada para a programação visual. O Klik&Play (Clickteam.com) faz uso de uma linguagem orientada a eventos.

Independente do processo idealizado por cada um dos ambientes, para definir uma sociedade de agentes é necessário dispor de elementos que descrevam essas sociedades, entre eles: definição de agente, definição de comportamento, definição do ambiente, definição de variáveis. Além disso, os ambientes devem oferecer ferramentas conceituas (gráfico (*plot*), monitores, botões de acionamento, caixa de texto, *slider* – para controle de parâmetros, exportar dados pra área externa) para facilitar o acompanhamento do modelo e possibilitar a extração de dados comportamentais e análise. Uma característica importante para a prática da aprendizagem cooperativa é a possibilidade do ambiente ser disponível para uso na web.

Os ambientes citados não apresentam simultaneamente as características relacionadas e, praticamente, todos exigem conhecimento e o uso de linguagem de programação específica. Em nosso entendimento esses dois fatos contribuem para uma

---

<sup>1</sup> StarLogo (<http://education.mit.edu/starlogo/>), Swarm (<http://www.swarm.org>), Repast ([repast.sourceforge.net/](http://repast.sourceforge.net/)), AgentSheets (<http://www.agentsheets.com/>), NetLogo (<http://ccl.northwestern.edu/netlogo/>), SIMULA (<http://simula.sourceforge.net/>), Klik&Play (<http://www.clickteam.com>)



baixa difusão do estudo de sistemas descentralizados. O que pretendemos aqui é validar a possibilidade de ter um ambiente para uso na web de descrição de problemas, através da modelagem de simulação multiagentes, com o alcance daqueles baseados em uma linguagem de programação e com a facilidade de programação oferecida por aqueles que disponibilizam uma interface mais amigável ao usuário. Em outras palavras, pretendemos oferecer um sistema que reúna as principais características relevantes dos sistemas estudados e que se integre a um ambiente de resolução cooperativa de problemas, baseado na web.

## **5. NetPlay – Concepção**

O NetPlay é parte integrante do ambiente AProSiMA, um ambiente para resolução cooperativa de problemas, baseado na Simulação Multiagente, ora em desenvolvimento como tese de doutoramento de um dos autores e a implementação do NetPlay foi realizada como trabalho de conclusão de graduação de três dos autores. O ambiente NetPlay foi projetado no intuito de possibilitar e estimular os usuários a criar cooperativamente modelos de simulação, na perspectiva do uso da modelagem multiagente, da capacidade de reuso dos objetos projetados e visando diminuir o esforço da programação, para que estes possam visualizar o comportamento resultante das interações definidas e fazer conjecturas em relação ao modelo e tomada de decisão.

### **5.1. Uso para web**

O objetivo de implementar a ferramenta com características para web se dá, em primeiro lugar, pela facilidade de uso e acesso por meio de navegadores disponíveis nas máquinas dos usuários, dispensando a instalação de novos programas. Segundo, porque o mesmo será acoplado a um ambiente maior de resolução cooperativa de problemas, que dispõe de um conjunto de outras ferramentas, sendo estas também projetadas para web. Outros elementos que sugerem a necessidade de uso na web estão diretamente relacionados ao ambiente ao qual o NetPlay será incorporado, e não destacaremos aqui.

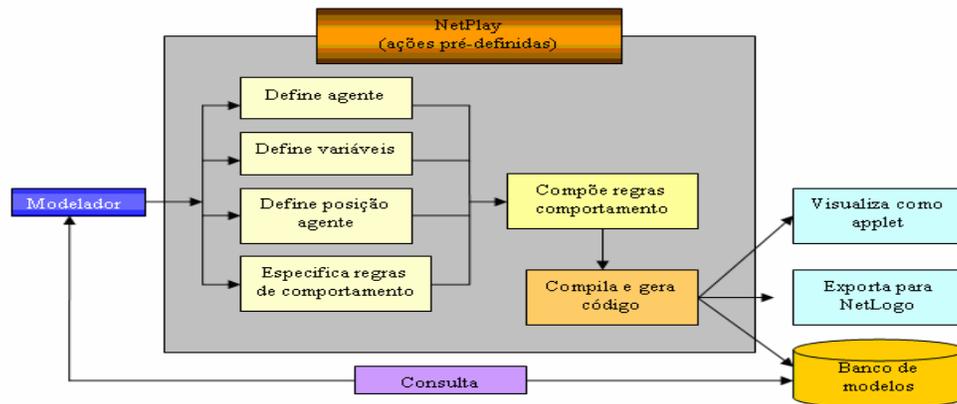
### **5.2. A linguagem e o código**

A linguagem adotada pela ferramenta segue o paradigma visual, sendo a programação realizada através da composição de regras simples de comportamento, manipulada através de figuras ilustrativas e caixas de diálogo. Na ferramenta NetPlay não se visualiza código em forma de texto. O modelador define os elementos do modelo de simulação multiagente através da seleção de objetos e atributos pré-definidos, preenchendo caixas de diálogo, e compondo regras de comportamento e interação desses elementos através do preenchimento de relações entre eles. Cada objeto definido tem uma biblioteca de primitivas para compor os procedimentos que governam o comportamento e as interações dos agentes. Para visualizar o funcionamento do sistema, têm-se duas opções: a) acompanhar através de um ambiente NetLogo (Wilensky, 1999), usando o programa gerado pelo NetPlay ou, b) utilizando uma applet produzida pelo NetPlay.

### **5.3. Ambiente de desenvolvimento**

O ambiente de desenvolvimento é onde o modelador escreve o programa, isto é, onde ele define os componentes do modelo: agentes (que são as classes) as regras de

comportamento (que são os métodos) e as características de todos os agentes (que são os atributos). A arquitetura do ambiente pode ser vista na **Figura 1**.



**Figura 1 - Arquitetura Netplay**

Como afirmado em 4.1 para definir uma sociedade de agentes é necessário dispor de elementos que descrevam essas sociedades. Em sua arquitetura o NetPlay disponibiliza esses elementos. Num primeiro momento são estabelecidos: definição de agentes (cor, forma, tamanho, quantidade, identificação), definição de variáveis (referentes aos agentes ou específicas ao fenômeno), posicionamento dos agentes no ambiente e estabelecimento das regras de comportamento (tipo de movimento, ações). Na seqüência, definido o comportamento individual dos agentes, passa-se a fase de composição das regras que definem as interações entre agentes e entre agente e ambiente. O passo seguinte é a geração automática do código que possibilita a visualização do modelo de simulação, através de um applet ou no ambiente Netlogo (Wilensky, 1999), através da exportação do código gerado. Todos os modelos são armazenados no banco de dados do ambiente.

O ambiente trabalha com um modelo de caixas de diálogo para definição de seus componentes. Por se tratar de uma ferramenta que trabalha na web, sua arquitetura segue as definições dos sistemas web tradicional, baseada na abordagem cliente/servidor. O NetPlay é um conjunto de páginas dinâmicas que possibilitará a construção de modelos de simulação multiagentes, permitindo o usuário criar, excluir, alterar conteúdo, e exportar para o servidor de execução, além de possibilitar a visualização de modelos já criados pelo usuário em momentos anteriores. Toda a ferramenta foi desenvolvida em Java.

## 6. NetPlay – Funcionamento

A entrada no ambiente é executada depois de feito login, caso o usuário ainda não possua um, este poderá ser realizado imediatamente após cadastro no ambiente, conforme as telas apresentadas na **Figura 2**. Depois de logar no ambiente (tela a), uma vez cadastrado (tela b), é apresentado um menu de ações (tela c), podendo o usuário: incluir, alterar, consultar ou excluir um modelo existente; ou consultar versões existentes dos modelos. Neste caso é oferecida uma ferramenta de pesquisa. Uma vez solicitado à criação de um novo modelo, é apresentada ao usuário os elementos necessários para a sua composição (**Figura 3**): cadastrar agente, cadastrar variável, cadastrar as condições relacionadas ao comportamento do agente e a composição dessas, que descreverão o comportamento emergente do modelo.



Figura 2 -Telas de entrada do NetPlay

Como ferramentas auxiliares são apresentadas às opções de: cadastrar monitor, caixa de texto e plot (gráfico). Para iniciar a construção do modelo é condição necessária definir um nome e uma descrição para sua funcionalidade.



Figura 3 - Menu de Ações, à esquerda, e inserir um novo modelo, à direita

O ambiente permite três tipos de agentes: **móvel** – são os agentes ativos no ambiente; **estacionário**, podendo este ser de dois tipos: **aleatório** – que caracteriza a base do ambiente, e **borda** – para delimitar uma área específica para atuação dos demais agentes, Figura 4.



Figura 4 - Inclusão de agente com seus atributos

Após definir o tipo de agente é necessário definir sua categoria por um nome. Os atributos que caracterizam um agente são: cor, tamanho, forma (para agente móvel), quantidade (para os agentes móveis é definido através de um valor inteiro; para o estacionário aleatório, através de porcentagem de dispersão sobre a tela de visualização,

quando se tratar de distribuição aleatória); posição, podendo está ser aleatória ou específica. No caso da distribuição aleatória o ambiente se encarrega de distribuir randomicamente os agentes sobre o ambiente, no caso de posição específica é apresentado uma representação definida do ambiente, através de coordenadas cartesianas, onde o modelador pode posicionar o local desejado para disposição do agente (móvel ou estacionário). No caso de agente de borda, o modelador poderá selecionar o lado do quadrado (variando de 1 a 17 unidades). Esse quadrado inicia no centro da tela e vai sendo incrementado em uma unidade até seu limite máximo, que é a fronteira da tela, quando o modelo é visualizado na interface gráfica do NetLogo (Wilensky, 1999).

Três tipos de variáveis podem ser associados ao modelo: **Lógica** – representação para o valor verdadeiro/falso da variável. **Slider** – variável global que são acessíveis pelo agente. Ambas são usadas nos modelos para possibilitar uma mudança rápida do valor da variável sem a necessidade de gravar o modelo a toda ação. Na interface gráfica a primeira assume a representação de *Switch* e a segunda pelo mesmo nome - Slider. Elas são úteis para avaliar o comportamento do modelo diante desta alteração. **Valor** – é uma variável que admite valor específico. As variáveis podem estar ou não relacionadas aos agentes. Uma vez selecionado a opção de “variável associada a agente”, são apresentados todos os agentes já definidos no modelo para que possa ser realizada a associação. Cada um dos tipos de variáveis tem seus atributos específicos a serem preenchido. Detalhes Figura 5.



Figura 5 - Inclusão de variáveis tipo lógica

Cada agente do modelo tem um comportamento bem definido com relação aos demais e com relação à definição do próprio ambiente. Atribuímos três opções para definir o comportamento, que são definidos por regras baseado no princípio do “*SE condição ... ENTÃO ação*”. E essas podem ser compostas em relação: ao **posicionamento** dos agentes; à **percepção** de algum outro agente (móvel ou estacionário); ou através de **comparação de variável**. Todas as condições definidas ficaram à disposição do modelador para o momento da composição das regras que regem o comportamento do modelo. Essa composição pode ser realizada através dos conectivos lógicos (“e”/“ou”). Cada composição é uma nova condição que resultará em uma nova ação e realizada nos mesmos moldes de uma condição simples.

Inclusão de Condições e Ações

Composição: atropelamento Operador Lógico: e

Condições: Clique no botão abaixo para incluir uma Condição. Nova Condição

Ações: Clique no botão abaixo para incluir uma Ação. Nova Ação

b) "Se" Condição

Escolha o Tipo de Condição

Percepção, Posicionamento, Cor do Agente, Valor de Variável, Comparação de Variáveis

c) "Então" Ação

Selecione o tipo de Ação a ser criada:

Criar Elemento, Destruir Elemento, Alterar Cor, Alterar Forma, Incrementar Variável, Recuperar Valor de Variável, Alterar Posição, Alterar Movimento

Figura 6 - Define uma composição (acima) , Seleciona condição (à esquerda) e uma ação correspondente (à direita)

A Figura 6 apresenta as janelas para: a) composição das regras e operador lógico para a composição, inclusão de "condição" e de "ação"; b) opções de "condição": percepção do agente para outro agente ou objeto no ambiente; posicionamento do agente no ambiente; alteração de cor do agente; alteração do valor de variável e comparação de variáveis; c) opções de "ação", mediante satisfação da condição: gerar novo agente; excluir um agente; alterar cor do agente; alterar movimento do agente; alterar forma do agente; alterar posição do agente; incrementar valor na variável ou recuperar valor de variável.

Além de possibilitar a definição dos agentes e as regras que caracterizam o modelo, o NetPlay oferece ainda a possibilidade de criar ferramentas conceituais para análise do modelo: **monitores** – apresenta o valor de qualquer expressão, que pode ser uma variável ou uma expressão mais completa. Sua atualização se dá automaticamente, em intervalo de segundos. **Caixa de texto** – usada para oferecer informação referente ao modelo ou algum outro elemento apresentado na interface de visualização. O teor das informações não se altera com a execução do modelo. **Gráfico (plot)** – essa ferramenta gera o gráfico, em tempo real, dos dados do modelo, Figura 7. Os elementos que definem cada uma dessas ferramentas é compatível com aqueles descrito no NetLogo (Wilensky, 1999).

Após definido o modelo, o NetPlay gera o código referente às características descritas na linguagem, dialeto Logo, utilizada no ambiente NetLogo (Wilensky, 1999), local onde o modelo será visualizado. A Figura 8, apresenta um fragmento de código gerado pelo NetPlay, à esquerda e, à direita, a interface padrão do NetLogo. Disposta sobre ela as ferramentas possíveis de gerenciamento pelo NetPlay (monitor, plot, caixa texto). Depois de gerado o código, o modelador pode optar em visualizá-lo diretamente no NetLogo, que deverá estar instalado em sua máquina, ou através de um applet, gerenciado pelo sistema.

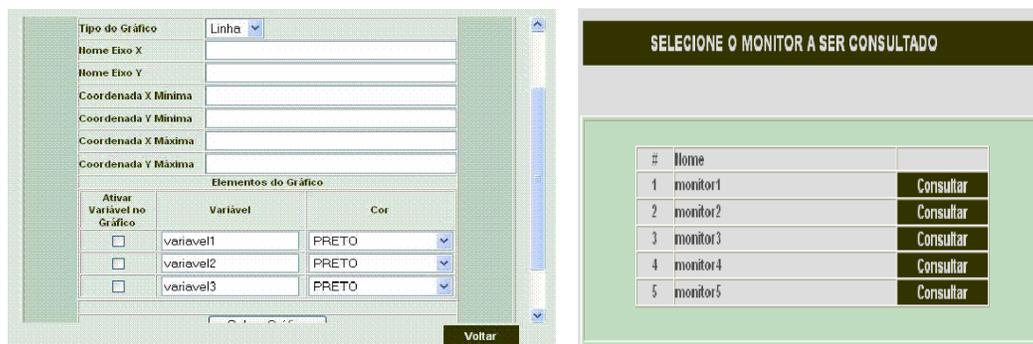


Figura 7 - Inclusão de Gráfico no modelo, à esquerda e, à direita, relação de monitores definido no modelo e visualizados pela opção consulta monitor.

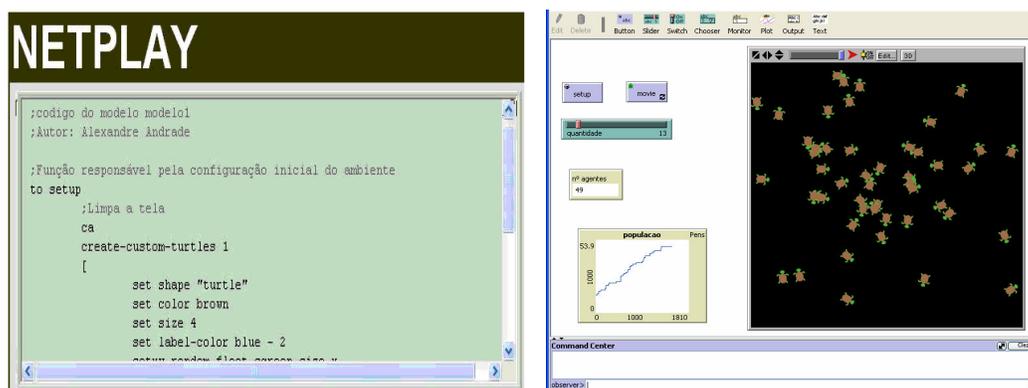


Figura 8 - Código gerado pela ferramenta, à esquerda e, à direita, interface do NetLogo padrão com as ferramentas possíveis de serem gerenciadas pelo NetPlay

## 7. Considerações finais

A ferramenta NetPlay está sendo desenvolvido para fins didáticos, sendo um dos seus objetivos ser de fácil utilização, no intuito de possibilitar e estimular os aprendizes a criar seus próprios modelos de simulação, na perspectiva do uso da modelagem multiagente, da capacidade de reuso dos objetos projetados. Visando também diminuir o esforço da programação, NetPlay tornará possível aos aprendizes visualizar o comportamento resultante das interações definidas e fazer conjecturas em relação ao modelo e às tomadas de decisão.

Da forma como foi projetada a ferramenta NetPlay, a concepção do modelo não exige conhecimento algum do modelador sobre o uso de agentes ou linguagem de programação. Apesar de ainda estar em desenvolvimento, a ferramenta tem se mostrado bastante promissora em seu propósito. As opções disponibilizadas no ambiente tem correspondido à concepção dos modelos com o mesmo grau de eficiência daqueles modelados diretamente no NetLogo (Wilensky, 1999).

O próximo passo da pesquisa é o oferecimento de duas disciplinas, uma na graduação (ciência da computação) e outra na pós-graduação (mestrado em informática) no Departamento de Informática da UFES – “Simulação baseada em Sistemas Multiagentes - Aplicações em Educação”. Nesta disciplina faremos uso do NetPlay como ferramenta para construção de modelos com os objetivos de: a) obter dados que nos permita aprimorar os ambientes; b) iniciar um processo de sensibilização para o estudo de sistemas complexos e c) analisar os modelos e o processo de resolução de problemas, bem como as conseqüências das tomada de decisão. Como trabalhos futuros destacamos o empenho na implementação das ferramentas que compõem o ambiente



AProSiMA, que auxiliarão no processo de gerenciamento de informações para a resolução cooperativa de problemas, baseado na Simulação Multiagente.

### Referências bibliográficas

ABRAHAMSON, D., WILENSKY, U., **ProbLab goes to school: Design, teaching, and learning of probability with multi-agent interactive computer models**, IV Conference of the European Society for Research in Mathematics Education, San Feliu de Guixols, Spain., 2005.

AZEVEDO, L.L., MENEZES, C.S., CORSO, B., PONTAROLO, E., **Ambientes Computacionais Orientados a Agentes para apoio à Aprendizagem Baseada em Simulação**, XVI SBIE, Juiz Fora – MG, 2005.

BLIKSTEIN, P., WILENSKY, U., **Less is More: Agent-Based Simulation as a Powerful Learning Tool in Materials Science**, IV International Joint Conference AAMAS, Utrecht, Holland. (Manuscript in press), 2005.

BRASIL, Secretaria de Educação Fundamental, **Parâmetros curriculares nacionais: ensino médio: bases legais** / Secretaria de Educação Fundamental. – Brasília: MEC/SEF, 1998.

COLLIER, N., HOWE, T., NORTH, M., **Onward and Upward: The Transition to Repast 2.0**, Electronic Proceedings, Pittsburgh, PA USA, 2003

DROGOUL, A., FERBER, J., **Multi-Agent Simulation as a Tool for Modeling Societies: Application to Social Differentiation in Ant Colonies**, Lecture Notes In Computer Science; Vol. 830, Springer-Verlag, London – UK, 1992.

FROZZA, R., ALVARES, L. O. C., **SIMULA - Ambiente de Simulação em Sistemas Multiagentes Reativos**. In: ATIA98 - Workshop sobre Aspectos Teóricos da la Inteligencia Artificial, 1998, Neuquen. ATIA98 - Workshop sobre Aspectos Teóricos da la Inteligencia Artificial, 1998.

REPENNING, Alexander, **Agentsheets: A Tool for Building Domain-Oriented Dynamic, Visual Environments**, Ph.D. Dissertation, University of Colorado at Boulder, Department of Computer Science, 1993

RESNICK, M., **Turtles, termites, and traffic jams: explorations in massively parallel microworlds**, The MIT Press, Massachusetts, USA, 1991.

RUTHEN, R., **Adapting to Complexity**, Journal: Scientific American, 268, 1993.

SUSSMAN, Joseph M., **Ideas on complexity in systems - twenty views**, 2000. <http://web.mit.edu/esd.83/www/notebook/20ViewsComplexity.PDF>

Swarm Development Group. [Swarm. www.swarm.org](http://www.swarm.org). **Swarm Development Group**, Santa Fe, New Mexico. 1999.

WILENSKY, U., **NetLogo**. Center for Connected Learning and Computer-Based Modeling. Northwestern University, 1999, <http://ccl.northwestern.edu/netlogo>.