
NetPlay – uma ferramenta para construção de modelos de simulação baseado em multiagente

Livia Lopes Azevedo¹, Crediné Silva de Menezes^{2,3}

¹UniAraguaia – Universidade Federal do Mato Grosso (UFMT)

²Centro Tecnológico – PPGEE – Universidade Federal do Espírito Santo (UFES)

³Programa de Pós-Graduação em Informática na Educação (PGIE) – UFRGS

livia@cpd.ufmt.br, credine@inf.ufes.br

Abstract. *The study of complex phenomena is made considerably easier through multiagent simulations. Several environments are available for this purpose, and they allow the construction and simulation of models. Since these tools were developed for professionals, they are not designed for the lay computer user. In this paper we present NetPlay, a tool which was projected for educational use, to enable and motivate the user to create simulation models, cooperatively, using the viewpoint of a multiagent modeling. The objects thus constructed can be reused with the aim to decrease the effort of the programming. The use of NetPlay makes it possible to visualize the behavior resulting from interactions that have been defined for the agents that compose the system, thereby facilitating any analyses, and providing elements for new conjectures on the models, while, at the same time, supporting the decision-making process for reviewing these models.*

Resumo. *O estudo de fenômenos complexos são fortemente facilitados através da simulação multiagente. Com esta finalidade existem disponíveis vários ambientes que permitem a construção e simulação de modelos. Sendo desenvolvida para uso de profissionais estas ferramentas não se preocupam com usuários leigos em computação. Neste trabalho apresentamos o NetPlay, uma ferramenta que foi projetada com objetivos didáticos 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. Usando o NetPlay é possível visualizar o comportamento resultante das interações definidas para os agentes que compõem o sistema, facilitando a análise, dando elementos a novas conjecturas sobre o modelo e apoiando a tomada de decisão para revisão desses modelos.*

1. Introdução

A simulação por computador tem se mostrado um excelente instrumento de apoio à aprendizagem, particularmente útil na exploração de sistemas complexos, nos quais há interdependência e, até mesmo, dependência entre uma multiplicidade de fatores e agentes. Através da simulação os estudantes constroem sistemas controlando as variáveis que o descrevem e observando as modificações em seu comportamento, diretamente e indiretamente, usando ferramentas de apoio à percepção. Com a simulação por computador, podemos controlar a velocidade dos fenômenos, acelerando

e desacelerando, e também podemos instrumentar o modelo através da inclusão e exclusão de variáveis e alterações de parâmetros, visando uma melhor compreensão do objeto de estudo.

Mais recentemente tem-se usado a simulação baseada em agentes (“*agent-based modeling*”). Esse tipo de simulação é caracterizado pela utilização de uma grande quantidade de agentes interagindo uns com os outros, com pouca ou nenhuma coordenação centralizada. 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. Esta abordagem tem se mostrado bastante promissora, notadamente na compreensão de sistemas complexos.

O uso desta abordagem requer ambientes computacionais que possibilitem descrição e execução dos sistemas em estudo. Existem vários ambientes de programação especificamente projetados para modelagem baseada em agentes, entre eles: StarLogoⁱ, Swarmⁱⁱ, RePastⁱⁱⁱ, AgentSheets^{iv}, MASON^v e NetLogo^{vi}. Esses ambientes estão revolucionando a prática científica pela possibilidade de explorar as conexões entre o comportamento de indivíduos no micro-nível e modelos em macro-nível que emergem das interações de muitos indivíduos. Da mesma forma, as perspectivas de trabalhar com sistemas complexos mediante o uso de simulações multiagente estão ganhando importância para os educadores em todos os níveis de ensino, que estão voltando sua atenção para essas novas ferramentas, que têm influenciado mudanças nas salas de aula (Tisue e Wilensky 2004). Sendo o desenvolvimento desses ambientes um campo de estudo bastante efervescente.

Um dos aspectos que ainda requer estudos e desenvolvimentos é a de construção dos modelos, segundo a ótica de um estudante (observador de sistemas) que seja leigo em programação. Em geral, a definição de famílias de agentes, suas propriedades relevantes, as interações interfamiliares e intrafamiliares, requer a escrita de código em alguma linguagem de programação. Além disso, a observação do funcionamento desses sistemas ainda carece de melhores ferramentas de acompanhamento e análise. Para explorar este aspecto estamos desenvolvendo o **NetPlay** – uma ferramenta para construção de modelos de simulação baseado em multiagente. A ferramenta **NetPlay** foi projetada 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.

Este artigo está organizado da seguinte forma: a sessão 2 brevemente apresenta o contexto da modelagem baseada em agentes, requisitos para ambientes de modelagem multiagente, trabalhos correlatos e a justificativa para a criação de uma nova ferramenta. A sessão 3 descreve a ferramenta NetPlay. Um modelo desenvolvido nesta ferramenta é apresentado na sessão 4. Na sessão 5 apresentamos a utilização da ferramenta seguida das considerações finais e referências bibliográficas.

2. Modelagem baseada em agentes

Em geral, um modelo baseado em agentes consiste nas interações entre os agentes e entre estes e o ambiente. É importante ressaltar que mesmo num modelo simples é

possível perceber o comportamento complexo que emerge das interações, sendo estas informações valiosas sobre a dinâmica do sistema. Isso torna possível explorar as conexões entre o comportamento no micro-nível de indivíduos e modelos em macro-nível que emergem das interações de muitos indivíduos.

Através da modelagem baseada em agentes é possível observar como os agentes individuais (pessoas, bactérias, insetos, nações, ou organizações) interagem entre si e com seu ambiente; é possível modelar situações nas quais os indivíduos têm comportamentos diferentes e complexos, podendo levar em conta tanto as propriedades quantitativas (parâmetros numéricos) quanto qualitativas (comportamento individual) do sistema representado no modelo. Esta descrição não revela a complexidade do comportamento individual, mas sim as características espaço temporal que emergem do coletivo, assumindo assim que interações entre simples indivíduos podem produzir um comportamento coletivo altamente estruturado e inteligente. A simulação no computador é então usada para descobrir propriedades do modelo e assim, ganhar entendimento dentro de um processo dinâmico, o que seria muito difícil de modelar com técnicas matemáticas padrões (Epstein, et al., 1996) (Axelrod, 1996).

Atualmente o campo de modelagem baseada em agente tem expandido além de suas fronteiras e incorporando outros grupos de pessoas que trabalham numa gama de disciplinas incluindo ciências da computação, inteligência artificial, economia, sociologia, física, biologia, antropologia, entre outros.

2.1 Requisitos para ambientes de simulação multiagente

O trabalho com a simulação multiagente requer que seja possível: a) descrever o sistema a ser estudado; b) a simulação computacional do sistema e c) facilidades para o monitoramento e a análise desses comportamentos. Para definir uma sociedade de agentes é necessário dispor de elementos que descrevam essas sociedades. Entre esses elementos deve ser considerado:

- **definição do agente**, isto é, os elementos que descrevem as características do agente. Nesta categoria destaque deve ser dada a forma (aparência do agente); o *tamanho*; a *quantidade* – essencial para analisar o comportamento do modelo, pois se quisermos simular um comportamento de uma população, é interessante que tenhamos muitos agentes interagindo entre si; a *posição*; a *cor* – usado para diferenciar os elementos componentes do modelo e melhorar a visualização;
- **definição de comportamento**, alguns elementos necessários para definir o comportamento dos agentes são: movimento – forma como o agente percorre o ambiente; percepção – capacidade de o agente perceber outros agentes e ou objetos no ambiente; colisão – momento em que dois ou mais agente tende a ocupar a mesma posição no ambiente (variante da percepção);
- **definição do ambiente**, ou seja, o local onde o agente está posicionado e no qual desenvolve seu comportamento. Em sociedades artificiais, normalmente usa-se uma grade (malha) sobre o plano cartesiano;
- **identificação dos agentes e definição de variáveis** – a identificação do agente é usada para o estabelecimento das regras de comportamento. Através da identificação é possível estabelecer o controle e formação de subconjuntos (famílias) e classes de agentes.

Para descrição de modelos é importante estabelecer parâmetros relacionados aos agentes e ao modelo, denominadas de variáveis. As variáveis podem ser de diferentes tipos: lógicas, globais, locais e vinculadas aos agentes. Além das características descritas acima, para definição dos agentes e ambientes, destacamos a necessidade de algumas ferramentas conceituais para facilitar o acompanhamento do modelo e possibilitar a extração de dados comportamentais e análise (gráficos, contadores, análise dos resultados, etc.). Ferramentas com esse teor facilitam a aprendizagem

2.2 Ferramentas para modelagem baseada em agente – trabalhos correlatos

Existem vários ambientes de programação especificamente projetados para modelagem baseada em agentes, destacamos: StarLogo, Swarm, Repast, MASON, AgentSheets, e NetLogo. Em (Azevedo, et al., 2005) é apresentada uma descrição detalhada sobre cada um desses ambientes.

A maioria dos ambientes de modelagem baseado em agentes segue o paradigma de “*framework* e biblioteca”. O *framework* – conjunto de conceitos para projetar e descrever a modelagem baseada em agentes; a biblioteca – conjunto de implementações de software para fornecer as ferramentas de simulação. O ambiente pioneiro foi o Swarm, cujas bibliotecas foram escritas em Objective-C e posteriormente disponibilizadas o Java Swarm. O Repast surgiu como uma implementação Java do Swarm, mas divergiu deste em relação às opções oferecidas, tanto de linguagem quanto de ferramentas. Recentemente o MASON esta sendo desenvolvido como uma nova biblioteca de simulação multiagente escrito em Java. Esses ambientes têm difundido pelo fato de fornecerem uma padronização nos projetos de software e ferramentas para construção de modelos de simulação multiagente, sem limitar a complexidade ou tipo do modelo. Entretanto, apresentam algumas limitações: dificuldade de uso, devido à necessidade aprofundada em linguagem de programação; ferramentas insuficientes para construir modelos, principalmente as relacionadas à representação do espaço; ferramenta insuficiente para executar e observar o experimento da simulação; além da falta de ferramenta para documentar o modelo.

Os ambientes descendentes da família Logo: StarLogo e NetLogo seguiram um caminho completamente diferente. Sendo sua primeira proposta fornecer um ambiente de alto nível que possibilitasse, desde estudantes do ensino médio até pesquisadores, construir e aprender através da modelagem baseada em agentes. O StarLogo desenvolvido parte em Java e parte numa linguagem chamada YoYo, que é uma variante do Logo construída sobre Java para permitir programas híbridos e também para poder disponibilizar aplicações na web, via *applet*. Em janeiro de 2006, foi lançado o StarLogo TNG, a nova geração do StarLogo. O ambiente continua com a mesma filosofia, porém apresenta uma programação exclusivamente gráfica, nos moldes de blocos de encaixe. O ambiente NetLogo é escrito em Java e pode ser executado em várias plataformas (Linux, Windows, Mac). O NetLogo utiliza como linguagem de programação uma variação do Logo para apoiar execução simultânea e modelagem multiagentes. Entretanto, o NetLogo disponibiliza ferramentas bem sofisticadas (Shapes Editor, BehaviorSpace, System Dynamics, Hubnet, interfaces gráficas, etc). Os dois ambientes disponibilizam uma ótima interface gráfica e barra de ferramentas para controle do modelo (botões para acionamento e definição de variáveis, *plots* para construção de gráficos, monitores, caixa de mensagens, etc).

AgentSheets é um ambiente comercial. A ferramenta usa o paradigma da programação visual, isto significa que não há um código baseado em texto, todo o desenvolvimento é feito através de uma interface gráfica (arrastando e colando elementos da caixa de ferramentas, etc.). Por ser uma ferramenta comercial, não tivemos acesso a maiores informações sobre seu funcionamento interno. É importante ressaltar que os ambientes examinados continuam sendo desenvolvidos e revisados. Nossa avaliação considera apenas as observações realizadas em: MASON version 10, Repast 3.1, Swarm 2.2, StarLogo 2.2, AgentSheets 2.5 e NetLogo 3.0. Algumas de nossas observações podem ter sido superadas nas versões atuais.

2.3 Por que desenvolver um novo Ambiente?

Embora existam muitas ferramentas para descrever modelos de simulação baseado em agentes, para muitos usuários, o desenvolvimento de tais modelos ainda permanece um obstáculo. Segundo (Ropella, et al., 2002), (Minar, et al., 1996), (Railsback, et al., 2006), este problema resulta em grande parte pela ausência de treinamento e habilidade no uso de software por pesquisadores (usuários) de muitos campos que usam a modelagem baseada em agente (por exemplo: biólogos, ecólogos, economistas, cientistas políticos, sociólogos, entre outros). Todas as ferramentas citadas, exceto o StarLogo TNG e AgentSheets, utilizam linguagem de programação baseada em texto para descrever o modelo.

Entendemos que, além da falta de treinamento dos usuários, os ambientes ainda podem proporcionar metáforas e ferramentas mais convenientes ao trabalho dos resolvidores de problemas. Fazendo um apanhado sobre cada uma das ferramentas citadas podemos argumentar que os ambientes Swarm, Repast e MASSON, apresentam algumas limitações: dificuldade de uso, devido à necessidade aprofundada em linguagem de programação (Objective C, Java, Python, C#); ferramentas insuficientes para construir modelos, principalmente as relacionadas à representação do espaço; ferramenta insuficiente para executar e observar o experimento da simulação; além da falta de ferramenta para documentar o modelo. Com relação aos ambientes StarLogo e NetLogo, ambos usam uma linguagem de programação própria derivadas do Logo. Apesar da aparente facilidade da linguagem, à medida que os modelos vão ficando mais sofisticados é percebida uma exigência no domínio do programa e uma capacidade de programação que estão obviamente além do que se pode esperar de alunos até mesmo do ensino médio e usuários pouco familiarizados com programação. Apesar da facilidade de programação visual do AgentSheets, a ferramenta se torna frágil à medida que os modelos ficam mais elaborados. As principais dificuldades apresentada por esta ferramenta são: dificuldade na identificação do posicionamento do agente sobre a grade; os agentes só movimentam em células adjacentes, não é possível saltos; como os agentes vivem sobre as grades, estes são exatamente do tamanho da grade; a ferramenta foi idealizada para ser executada principalmente em Mac, a versão para Windows é limitada; e principalmente, os agentes não têm identificação e nem é possível classificá-los numa família para estabelecer regras de comportamento específicas.

3. NetPlay – Concepção

O NetPlay está sendo construído como componente de um ambiente na web para apoio à resolução cooperativa de problemas, o AProSiMA (Azevedo, Menezes, 2006), cuja proposta é disponibilizar, em um único local, ferramentas para: recuperação de informação (documentos, especialistas); comunicação e discussão de idéias; e criação

de modelos de simulação, para viabilizar a resolução cooperativa de problemas. A função do NetPlay neste ambiente é o de construção de modelos de simulação baseado em simulação multiagente. Com o objetivo de acessibilidade a usuários leigos em programação e tecnologia multiagente, propomos um novo modelo de programação para a ferramenta NetPlay. Na seqüência apresentamos detalhes da ferramenta.

Uma característica incorporada ao NetPlay é o dispositivo para análise gráfica que possibilita analisar em detalhes os dados apresentados durante o processo de execução. Com este dispositivo o usuário pode, além de inspecionar todos os pontos do gráfico, solicitar a montagem de novos gráficos destacando os pontos de interseção entre todas as variáveis plotadas; os pontos de coordenadas máximas e mínimos; os pontos principais de cada variável, isto é, destacado cada uma das variáveis e as coordenadas relativas às mesmas; o que facilita a análise do modelo pelo usuário.

3.1 Uso para web

O objetivo de implementar a ferramenta com características para web se deu, 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. Assim sendo, a troca de informações entre os usuários e a participação cooperativa se torna muito mais dinâmica. Todos os modelos construídos ficam disponíveis para que outros usuários possam consultar e reaproveitar para o desenvolvimento de novos modelos ou traçar estratégias diferentes – reuso.

3.2 Descrição dos Modelos

A descrição de um modelo de simulação multiagente no NetPlay pode ser comparada a uma peça teatral. Usando essa metáfora destacamos os seguintes elementos: a) o diretor – é o sujeito que caracteriza o cenário e os personagens e também orquestra toda a peça. Neste trabalho é representado pelo usuário; b) o cenário – é definido pelos elementos que compõem o palco, local onde a peça se desenvolve; c) os atores – são os personagens na peça do teatro que agem de acordo com o seu script.

Neste modelo o diretor (usuário) seleciona os atores e os objetos para compor o cenário. Após essa seleção, o diretor passa a descrever o comportamento de cada um dos atores, definindo como deve ser a sua atuação na peça, e posicionando os objetos no palco. Feito isso, passa a composição das regras de interação entre os componentes da peça. O diretor faz uso de um conjunto de regras descentralizadas, isto é, cada ator tem o seu próprio script (conjunto de composições). As regras definem o comportamento dos atores (condição) diante de determinada situação (ação). No palco ocorrem as interações entre os personagens (ator x ator) e com os elementos que compõem o cenário (ator x cenário) segundo os seus scripts. A interação entre os vários atores e os elementos que compõem o cenário faz emergir o comportamento descrito no modelo (peça teatral).

De modo geral, a maioria dos usuários sabe descrever claramente as características, o comportamento e a forma de interação dos atores (agentes), porém não sabem programar de que modo essas características e as interações possam ocorrer. Assim, no NetPlay o usuário descreve o modelo usando elementos desta metáfora, escolhendo os comportamentos típicos para compor cada papel. O ambiente de

simulação passa então a ser visto como uma metáfora do teatro de representação, onde o diretor (usuário) assiste o desenvolvimento das interações e estuda o modelo simulado que representa um retrato do mundo real, podendo interferir e tomar decisões sobre as possibilidades e ocorrências.

3.3 Ambiente de desenvolvimento

O ambiente de desenvolvimento é onde o modelador relaciona os elementos do modelo, isto é, onde ele define os componentes do modelo. Fazendo analogia a programação orientada a objetos temos: 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 estrutura do ambiente pode ser vista na Figura 1. 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 possibilita a construção de modelos de simulação multiagente, além de possibilitar a visualização de modelos já criados pelo usuário em momentos anteriores. Toda a ferramenta foi desenvolvida na linguagem de programação Java.

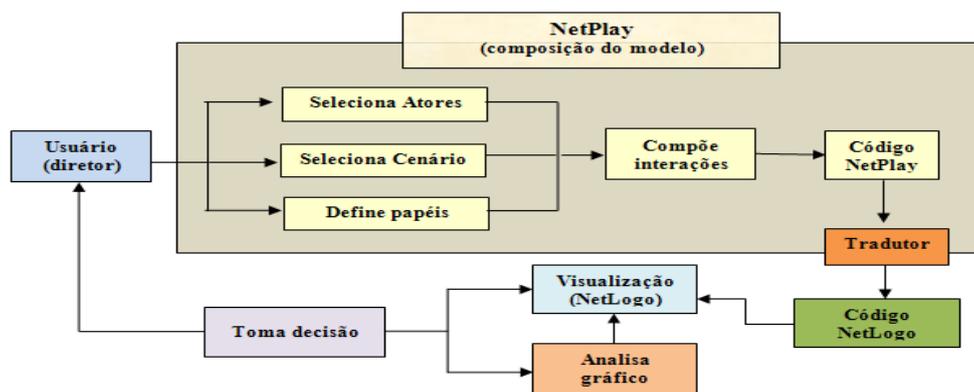


Figura 1 – Estrutura interna NetPlay

3.4 Realização dos Modelos

A realização do NetPlay requer uma máquina abstrata onde seja possível encenar as peças concebidas por seus usuários. Para agilizar a implantação do sistema estamos traduzindo os modelos desenvolvidos em NetPlay para programas em NetLogo, Figura 1. O tradutor é o conjunto de bibliotecas que decodifica as características definidas na linguagem de programação visual e a transforma em uma linguagem de programação baseada em texto. A estratégia tem por finalidade experimentar o modelo antes da concepção de uma máquina abstrata específica. Assim, a tradução do modelo poderia ser feita para qualquer linguagem, optamos pelo NetLogo por economia de tempo em implementação desnecessária de interfaces, uma vez que o ambiente NetLogo já trás embutido nele toda essa estrutura. Desta forma, para visualizar o funcionamento do sistema, têm-se duas opções: a) acompanhar através do ambiente NetLogo, usando o código gerado pelo NetPlay ou, b) utilizando um *browser* para visualizar o *applet* do modelo. A ferramenta NetPlay esta disponível em <http://200.137.66.20/AProSiMa/>.

3.5 Descrição de uso do NetPlay

Para ilustrar o uso do NetPlay descrevemos o modelo “vírus” [Wilenky, 1998], que retrata a transmissão e perpetuação de um vírus numa população humana. Os detalhes da modelagem estão disponíveis em http://www.inf.ufes.br/~liviaufmt/modelo/modelo_virus.htm.

De forma resumida, o desenvolvimento de uma epidemia pode ser explicado, conhecendo todos os fatores que o influenciam, com base nos fenômenos epidemiológicos de contágio viral. Neste caso, os fenômenos de maior importância são a transmissão por contágio direto pessoa a pessoa e a transmissão através de ar contaminado em espaços confinados

3.5.1 Definição dos agentes

O NetPlay possibilita a criação de três tipos de agentes (atores, segundo a metáfora): móveis, estacionários e borda. Como o próprio nome sugere, os agentes móveis têm a capacidade de percorrer o ambiente que é definido pelos agentes estacionários, isto é, os que compõem o ambiente de interação. O agente borda é utilizado para delimitar uma região no ambiente. Para este modelo, os agentes são pessoas, aqui caracterizados de doentes (vermelho) e saudáveis (verde). O diretor seleciona os atores e entrega o script do comportamento a ser executado. A Figura 2 representa a criação dos agentes no NetPlay e a definição de seu comportamento, neste caso andar aleatoriamente pelo ambiente (palco). O diretor define a quantidade de atores que iniciarão a peça.



Figura 2 - Criação de agentes no NetPlay

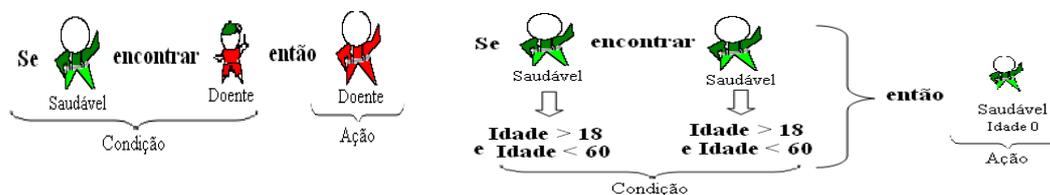
3.5.2 Definição de variáveis

As variáveis neste modelo estão relacionadas aos papéis que os agentes desempenharão. Elas são utilizadas para guardar valores e auxiliar na simulação, por exemplo: estabelecer a chance de o agente ser infectado ou não, tornar-se imune, computar a idade, avaliar a chance de reprodução, entre outras. Através das variáveis o diretor pode alterar o comportamento dos atores e conseqüentemente da simulação.

3.5.3 Regras de comportamento

O comportamento dos agentes que definem a simulação são obtidos através de regras lógicas *se condição ... então ação*. As **composições**, que descrevem o comportamento dos agentes, são compostas em duas etapas: a primeira determinando as situações que representam as condições e posteriormente as situações que representam as ações decorrente dessa. A Figura 3 (a) representa a seguinte regra de comportamento: *Se um agente **saudável** encontrar (significa perceber outro agente à distância 0) um agente **doente**, então ele se contamina e também fica **doente***. O diretor tem a sua disposição uma série de combinações possíveis para compor os relacionamentos (percepção, posicionamento, cor, forma, relacionado às variáveis, reprodução, morte, etc). Cada uma dessas opções remete a uma construção, seja ela selecionando os elementos (por exemplo, a percepção indica que dois agentes devem se encontrar a uma determinada distância) ou compondo uma expressão. Para as composições que necessitam de mais de uma condição é utilizado os operadores lógicos (e) e (ou). A Figura 3 (b) indica se dois agentes saudáveis se encontrarem e a idade de ambos estiver no intervalo ente 18 anos e 60 anos, eles geram um descendente. Uma mesma condição pode levar a mais de uma

ação. Da mesma forma, há ações que não necessitam satisfazer nenhuma condição, por exemplo: incrementar tempo.



- a) Composição de contaminação b) Composição reprodução (duas condições uma ação)

Figura 3 - Composições (regras de comportamento)

3.5.4 Ferramentas de monitoramento

São oferecidas duas ferramentas para monitoramento do modelo: monitores e gráficos. Através dos monitores é possível inspecionar a variação dos agentes no decorrer da simulação, geralmente através de uma contagem. Paralelamente ao palco de visualização do movimento dos agentes, o comportamento do modelo pode ser observado através da plotagem num gráfico (linha, barra ou pontos) das variáveis que controlam o comportamento dos agentes. A Figura 4 representa a análise do diretor ao ver o desenvolvimento dos atores no palco. O diretor acompanha o desenrolar das ações através dos monitores e do gráfico.



Figura 4 - Visualização do modelo

3.5.5 Ferramenta de análise gráfica

Em geral, durante a execução de uma simulação ocorrem muitos ciclos. Devido à quantidade de pontos gerados e plotados no gráfico uma visualização mais pontual torna-se difícil de precisar. Desta forma, incorporamos a ferramenta de construção de modelos de simulação um dispositivo de análise gráfica que permite avaliar precisamente: os pontos de interseção entre todas as variáveis plotadas; os pontos de coordenadas máximas e mínimas; os pontos principais de cada variável, isto é, destacado cada uma das variáveis e as coordenadas relativas às mesmas; para uma visão mais geral, todos os pontos são apresentados simultaneamente num mesmo quadro. Outro elemento de destaque do dispositivo de análise gráfica é a possibilidade de exportar os dados em forma de planilha para que possa ser lido em outras ferramentas (*nome_arquivo.csv*). Esta facilidade permite que os dados sejam trabalhados isoladamente e novos gráficos possam ser construídos de acordo com o interesse do usuário.

4. Utilização da ferramenta NetPlay

Foi ofertada, no primeiro semestre de 2007, a disciplina “Simulação baseada em sistemas multiagentes – aplicações em educação“, tendo três objetivos principais: a) iniciar um processo de sensibilização para o estudo de sistemas descentralizados, mostrando que é possível e simples modelar situações na quais os indivíduos têm

comportamentos diferentes e complexos; b) testar a ferramenta para construção de modelos de simulação – NetPlay; c) fazer um comparativo entre as ferramentas NetLogo e NetPlay. Participaram da disciplina 8 alunos. A disciplina teve duração de 45 horas, sendo um encontro semanal com duração de 3 horas, num total de 15 encontros.

A disciplina fundamentou-se na participação cooperativa. A metodologia adotada priorizou: leitura e discussão sobre sistemas multiagente e sistemas complexos; a exploração de modelos de simulação multiagente e posterior discussão sobre os dados observados; intensivamente na construção de novos modelos de simulação multiagente, utilizando as ferramentas disponíveis; apresentação dos modelos implementados.

Com relação à aprendizagem de sistemas complexos pudemos perceber o interesse dos alunos com relação à nova forma de ver os fatos, que até então passavam despercebidos. Por exemplo, ao analisar o modelo presa x predador, usando a abordagem analítica (equações diferenciais) e a de simulação multiagente, para explorar o comportamento resultante das interações entre os agentes, os alunos puderam perceber mais nitidamente que as oscilações dependiam criticamente dos parâmetros e como essas influenciavam o modelo. Apesar do conhecimento matemático que eles tinham, perceberam que a exploração com modelos de simulação é nitidamente diferente e muito mais acessível, enquanto a exploração com equações é muito mais abstrata, pois trabalha com símbolos. Elemento que merece destaque foi o fato de despertar nos alunos as diferentes formas de análise de comportamento do modelo: na forma analítica as relações se dão em termos do comportamento da população no decorrer do tempo; enquanto na forma de simulação multiagente às ações e interações estão mais relacionadas ao comportamento individual dos agentes, que é mais próxima da nossa forma de pensar. Outros aspectos que despertaram o interesse dos alunos foram: a) a possibilidade e facilidade de analisar os modelos com a introdução de muitos fatores aleatórios e várias inter-relações entre as variáveis ou agentes; b) que pequenas alterações nos parâmetros refletiam em grandes alterações no comportamento do modelo; c) perceber as alterações do modelo realizadas pela aleatoriedade do comportamento dos agentes e não pela imposição do que o agente deve fazer; c) estudar sistemas multiagente em situações definidas por relacionamento através de regras simples e não como protocolos e programas elaborados.

Além dos modelos previamente selecionados cada aluno desenvolveu um modelo de seu interesse, representativo de um fenômeno complexo. Foram apresentados 3 modelos relacionados ao comportamento de insetos (abelhas e cupins). Um deles pretendia observar o comportamento interno de uma colméia, através das atividades desenvolvidas em cada ciclo de vida das abelhas. Os outros dois pretendiam observar o comportamento externo dos insetos (abelhas e cupins) com relação à coleta de alimentos e defesa da colônia, respectivamente. Dois modelos tinham objetivos de avaliar o desempenho dos agentes em situações aleatórias: um simulando uma equipe de troca de pneus num *pitstop* de formula um, e o outro avaliando a performance de jogador e goleiro em situação de marcação de pênalti. Para avaliar o comportamento de animais durante a jornada de migração, foi modelado a “marcha dos pingüins”, baseado no filme de mesmo título. Em caráter organizacional foi modelado o movimento de um armazém: recebendo, depositando, organizando e despachando cargas. Para avaliar a influência dos semáforos no tráfego de veículos foi criado um modelo que mapeava a interferência deste em condições de fluxo de trânsito.

Durante todo o processo de construção dos modelos os alunos foram instigados sobre os seus propósitos, tanto pelo mediador como pelos demais alunos. Todo o material produzido neste curso está disponível numa página de construção cooperativa <http://simulacaomultiagente.pbwiki.com/>. Com as discussões e os modelos desenvolvidos pelos alunos constatamos a importância do estudo de sistemas descentralizados através de modelos de simulação para que os alunos possam pensar, não apenas no processo do fenômeno, mas explorando conceitos e idéias que até então eram inacessíveis.

Quanto ao uso do NetPlay foi consenso que a ferramenta tem uma interface muito amigável, sendo adequado principalmente para pessoas que não possuem domínio em programação. Apresenta grande facilidade de representação do modelo através da composição dos elementos e possibilidades de relacionamentos. O fato de ser uma ferramenta para web possibilita maior mobilidade do usuário e o compartilhamento de modelos em tempo real. Quanto ao NetLogo, eles acharam que o ambiente possibilita grande possibilidade para construção de modelos de simulação, entretanto é necessário conhecimento da linguagem e domínio do ambiente para utilizar todos os recursos que ele oferece.

5. Considerações Finais

A ferramenta NetPlay está sendo desenvolvida para fins didáticos, sendo um dos seus objetivos ser de fácil utilização, no intuito de possibilitar e estimular os usuários a criar seus próprios modelos de simulação, na perspectiva do uso da modelagem multiagente, da capacidade de reuso dos objetos projetados. Da forma como a ferramenta NetPlay foi projetada, a concepção do modelo não exige conhecimento algum do modelador sobre o uso de agentes ou linguagem de programação. O que possibilita ao modelador preocupar-se, exclusivamente, com o fenômeno e as ações decorrentes da sua observação e não com linguagem de programação. Visando também diminuir o esforço da programação, NetPlay torna possível aos usuários visualizar o comportamento resultante das interações definidas e fazer conjecturas em relação ao modelo e às tomadas de decisão. Outra característica importante da ferramenta é o fato de ser projetada para web, o que possibilita o seu acesso em qualquer lugar, independente de instalação de software.

Trabalhar sistemas complexos e sistemas multiagente, novidade para alguns alunos, através de modelos de simulação, despertou neles a curiosidade e surpresa ao observar o comportamento emergente dos sistemas, resultante da interação de partes tão simples. Os modelos explorados e as conjecturas levantadas indicam a importância de inserir nos currículos educacionais abordagens nesta direção desde o ensino fundamental.

Como trabalhos futuros temos duas frentes: uma pedagógica, através de seminários e oficinas de modelagem de sistemas complexos usando o NetPlay; uma técnica, aperfeiçoando a ferramenta e, principalmente, desenvolvendo a máquina abstrata para execução dos modelos.

Referências Bibliográficas

Axelrod, R.(1996), *The complexity of cooperation: agent-based models of competition and collaboration*. Princeton, NJ: Princeton Univ. Press.

-
- Azevedo, L.L., Menezes, C.S., Pantarolo, E., Magdalena, B.C. (2005), Ambientes Computacionais Orientados a Agentes para apoio à Aprendizagem Baseada em Simulação, SBIE2005, Juiz de Fora.
- Azevedo, L.L., Andrade, A., Velasco, D., Barbiero, L., Cury, D., Menezes, C. S. (2006), NETPLAY: um ambiente para simulação multiagente, Renote/dez2006
- Azevedo, L. L., Menezes, S. S., (2006), AProSiMA - Um ambiente na web para resolução cooperativa de problemas em simulação multiagente, Renote/dez2006
- Epstein, J. M., & Astell, R. L. (1996), *Growing artificial societies: social science from the bottom up*. Cambridge, MA: MIT Press.
- Minar, N., Burkhart, R., Langton, C., & Askenazi, M. (1996). *The Swarm Simulation System, A Toolkit for Building Multi-Agent Simulations*. Acesso em 2006, <http://citeseer.ist.psu.edu/minar96swarm.html>
- Railsback, S. F., Lytinen, S. L., & Jackson, S. K. (2006), *Agent-based simulation platforms: Review and development recommendations*. Acesso em 2006, <http://www.humboldt.edu/~ecomodel/documents/ABMPlatformReview.pdf>
- Ropella, G. E., Railsback, S. F., & Jackson, S. K. (2002), Software engineering considerations for individual-based models. *Natural resource modeling, vol 15*.
- Tissue, S., Wilensky, U. (2004), NetLogo: Design and Implementation of a Multi-Agent Modeling Environment. Proceedings of Agent 2004, Chicago, <http://ccl.northwestern.edu/papers/netlogo-swarmfest2004.pdf>
- Wilensky, U. (1999), "NetLogo", Center for Connected Learning and Computer-Based Modeling, Evanston, IL. 1999-2006 <http://ccl.northwestern.edu/netlogo>
- Wilensky, U., NetLogo Virus model, (1998). <http://ccl.northwestern.edu/netlogo/models/Virus>. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

ⁱ (<http://education.mit.edu/starlogo/>)

ⁱⁱ (<http://www.swarm.org>)

ⁱⁱⁱ (<http://repast.sourceforge.net/>)

^{iv} (<http://www.agentsheets.com/>)

^v (<http://cs.gmu.edu/~eclab/projects/mason/>)

^{vi} (<http://ccl.northwestern.edu/netlogo/>)