CORDENONSI, A. Z.; LOY, Alisson Moscato. Desenvolvimento de um Sistema de Simulação para Sistemas Multiagentes Tridimensionais. Revista do Centro de Ciências da Economia e Informática, Bagé - RS, v. 11, n. 1, 2003.

DESENVOLVIMENTO DE UM SISTEMA DE SIMULAÇÃO PARA SISTEMAS MULTIAGENTES TRIDIMENSIONAIS

Alisson Moscato Loy[1], *Andre Zanki Cordenonsi*[2]

RESUMO

O presente trabalho propõe uma ferramenta genérica para simulação de sistemas multiagentes reativos em ambientes tridimensionais, permitindo uma análise e a obtenção de resultados mais próximos da realidade do nosso mundo. Tal ferramenta permite a aplicação de modelos comportamentais amplamente estudados em ambientes bidimensionais, agora num mundo tridimensional. Como validação do modelo, foi escolhido o problema clássico dos Robôs Mineradores.

Palavras-chave: Sistemas Multi-agentes, Sistemas Multiagentes Reativos, Simulação de Sistemas

ABSTRACT

The present paper implements a generic simulation tool for a reactive multiagent system in three-dimensional environments, which are next of the real world. This tool can use agents behaviors previously defined for a two-dimensional environments to test them in a 3D world. To validate the simulation environment, a classical problem was studied: the Mineral Robots.

KeyWords: Multiagent Systems, Reactive Multiagent Systems, Systems Simulation

INTRODUÇÃO

Os Sistemas Multiagentes (SMA) concentram-se no estudo de agentes autônomos em um universo multiagente. Para os SMA, o termo autônomo designa o fato de que os agentes têm uma existência própria, independente de outros agentes. Usualmente, cada agente possui um conjunto de capacidades comportamentais que definem sua competência, um conjunto de objetivos, e a autonomia necessária para utilizar suas capacidades a fim de alcançar seus objetivos. A idéia principal em um sistema multiagente é que um comportamento global inteligente pode ser alcançado a partir do comportamento individual dos agentes. Em um SMA não é necessário que cada agente seja individualmente inteligente para alcançar um comportamento global inteligente.

Um agente é uma entidade computacional com um comportamento autônomo que lhe permite decidir suas próprias ações (ALVARES, 1997). A decisão de qual ação levar a cabo é determinada pelo agente, tendo em consideração as mudanças acontecidas no ambiente em que atua e o desejo de alcançar seus objetivos. Segundo (DEMAZEAU, 1995), pode-se decompor um sistema segundo uma metodologia de Inteligência Artificial Distribuída através dos conceitos definidos a seguir.

Dado um determinado sistema, denomina-se *agente* cada uma de suas entidades ditas ativas. Este conjunto de agentes forma uma *sociedade*. As entidades passivas serão designadas pelo termo *ambiente*. Um agente recebe informações e raciocina sobre o ambiente, sobre outros agentes e decide quais ações deve realizar e quais objetivos deve seguir. Um agente é uma

[1] Centro Universitário Franciscano – UNIFRA, Curso de Sistemas de Informação (aloy@programmer.net)

[2] Centro Universitário Franciscano – UNIFRA, Curso de Sistemas de Informação (andrezc@unifra.br)

entidade ativa, ou seja, capaz de controlar suas ações, diferentemente das noções estáticas tais como módulos, conjuntos de regras e bases de conhecimentos.

Denomina-se interação entre agentes ou entre agente/ambiente uma troca de informações, que pode ser realizada de forma direta (comunicação explícita) ou de modo indireto (emissão de sinais através do ambiente). Um *organização* define todas as restrições aplicadas aos agentes pertencentes a uma determinada sociedade, ou seja, os meios através dos quais o projetista do sistema pode garantir que cada agente desejará e realizará a resolução dos problemas propostos.

SIMULAÇÃO DE SISTEMAS MULTIAGENTES

Apesar de largamente utilizados, sistemas de simulação de sistemas multiagentes (mundos virtuais) ainda não possuem uma definição formal para sua construção. O modelo mais abrangente foi proposto por (LATTAUD e CUENCA, 1998), que visava definir um modelo para a construção de ambientes evolutivos que integrassem todos os componentes destes, provendo um *framework* para a construção de ambientes dinâmicos. Este modelo define cinco classes, que são descritas a seguir.

A classe *ambiente abstrato* é a raiz de toda a hierarquia. Ela possui diversos parâmetros, que são comuns a todas as suas subclasses:

- tamanho do ambiente(S): determina se o ambiente é finito ou infinito;
- conjunto de objetos do ambiente(O): dividido em três sub-conjuntos (agentes, objetos inativos e sub-ambientes);
- conjunto de leis de interação(I): conjunto de regras que regem as interações entre agentes e objetos, agentes e agentes, agentes e ambiente e ambiente e objetos;
- conjunto de leis de evolução(E): determina as regras de evolução do ambiente, além das funções de transição que determinam a passagem de um estágio para outro.

A classe *ambiente situado* é utilizada geralmente em simulações de sistemas multiagentes. Um determinado sistema é dito *situado* se ele for construído num espaço delimitado, numa métrica conhecida, e seus objetos ocuparem uma posição relativa no espaço. Os parâmetros para esta classe são definidos a seguir:

- forma(F): definição das diferentes estruturas existentes (círculos, esferas, quadrados, etc);
- dimensão(D): de 1 a *n* dimensões;
- métrica(M): define a unidade de medida e a escala para a definição das coordenadas dos objetos do ambiente;
- conjunto de propriedades de gerência do espaço(P): define propriedades de visualização do ambiente.

Um *ambiente situado* pode ser definido como *discreto* se ele pode ser dividido em um número finito de partes elementares, denominadas células, onde cada célula possui um número fixo de vizinhos.

Um *ambiente situado* pode ser definido como *contínuo* se, para quaisquer dois pontos, há pelo menos um ponto entre eles. Esta ambiente é complementar ao ambiente *discreto*. Um ambiente *contínuo* pode ser construído utilizando uma métrica real, baseada em sistemas vetoriais reais.

A classe *ambiente não situado* pode ser utilizada para definir um ambiente em sistemas baseados em regras. Um ambiente é dito *não situado* se os conceitos de espaço e posição relativa não forem aplicados ao sistema.

Segundo a classificação proposta por (LATTAUD e CUENCA, 1998), percebe-se que a grande maioria dos sistemas multiagentes existentes podem ser classificados como *Ambientes Situados, Discretos e Bidimensionais*. Exemplos de sistemas com estas características podem ser encontrados em (FROZZA, 1997), (CORDENONSI, 2000), (REYNOLDS, 1997) e (DROGOUL, 1993).

No entanto, quando da passagem destes sistemas para ambientes reais, usualmente aplicados na robótica, ocorrem problemas de adaptação em ambientes tridimensionais (PFEIFER, 1996), pois as simulações realizadas utilizaram grides bidimensionais.

Portanto, o objetivo deste artigo é propor o desenvolvimento de um sistema multiagente reativo tridimensional. Neste ambiente, cada agente poderá ter até vinte e seis vizinhos diferentes, enquanto que os ambientes bidimensionais permitem um máximo de oito vizinhos diferentes. Este aumento na complexidade do ambiente trará novos desafios aos problemas clássicos dos sistemas multiagentes.

MODELAGEM DO SISTEMA

Escopo do Problema

Inicialmente, o sistema será desenvolvido para resolver um problema clássico dos sistemas multiagentes: o problema dos robôs mineradores. Futuramente, pretende-se generalizar o sistema para a resolução de outros tipos de problemas.

O problema dos Robôs Mineradores foi desenvolvido como um modelo para a funcionalidade emergente de um Sistema Multiagente Reativo em (STEELS, 1994). O problema possui as seguintes características: dado um certo ambiente, desconhecido, deve-se encontrar depósitos de minerais e estes devem ser levados a uma base central, também situada neste ambiente. Não existe nenhuma informação sobre as características do ambiente.

Este problema pode ser facilmente modelado para um sistema multiagente reativo, através da representação do ambiente por uma grade bidimensional, onde a posição dos agentes robôs, do depósito de minerais e da base podem ser representadas por coordenadas cartesianas deste ambiente. Os agentes mineral e base são fixos no ambiente, ou seja, suas posições não mudam no decorrer da simulação, enquanto que os agentes robôs são caracterizados por agentes reativos, que movimentam-se livremente pelo ambiente, procurando o mineral e trazendo este à base.

O problema dos Robôs Mineradores tornou-se um problema clássico na Inteligência Artificial Distribuída, sendo ainda largamente estudado. Diversos métodos e algoritmos podem ser encontrados na literatura, além de inúmeras variações do mesmo tema central. Dentre estes podemos citar: utilização de marcas (DROUGOL, 1993), arquitetura de subordinação (BROOKS, 1986) e *Fungus Eater* (PFEIFER, 1996).

O sistema multiagente desenvolvido foi baseado no modelo apresentado em (DROUGOL, 1993). A modelagem deste problema por (DROUGOL, 1993) foi concebida como uma forma para a apresentação e resolução coletiva de problemas.

Implementação do Modelo

O sistema desenvolvido é composto de cinco estruturas - Mundo, Base, Minas, Robôs e Bandeiras (*Flags*).

Cada objeto possui um identificador (ID) único, numérico, hexadecimal, que o qualifica e diferencia no mundo, conforme a tabela abaixo:

	ID	Observações
Mundo	0x0000	Posição vazia
Bandeiras (flags)	0x0001	Incrementada em 1 para cada nova bandeira
Minas	0xAAAA	3 minas (ID + 1, ID + 2 e ID + 3)
Base	0xBBBB	1 base no centro do mundo
Robôs (agentes)	0xFFFF	*n* agentes no sistema (ID + *n* identifica o robô)

Tabela 01 – Relação dos ID dos objetos.

O Mundo é simulado por um vetor de 100x100x100 posições numéricas (inteiros) inicializadas em 0x0000.

Os robôs são entidades de dimensão 1 x 1 x 1, mantidos em um vetor de 1 x *MAXROBO*, onde *MAXROBO* é a quantidade de robôs que o sistema deve ter na simulação. Para cada robô é necessário armazenar a posição (x, y, z), um *flag* que determina se o robô está com carga, um contador para verificar o grau de utilização deste robô e uma variável que armazena o ID do mundo para a posição do robô. Quando o robô muda de posição, ele restaura o ID anterior do mundo para a sua posição original, mantendo assim as informações das bandeiras (*Flags*). A posição inicial dos robôs é aleatória, mudando a cada nova simulação.

As minas são entidades de dimensão 2 x 2 x 2, mantidas em um vetor de 3 posições onde para cada mina é necessário saber sua carga atual. A posição das minas no mundo é constante, por isto não é necessário armazenar sua posição neste vetor, uma vez que o ID da mina no mundo já a identifica.

A base é posicionada no centro do mundo e possui uma dimensão de 2 x 2 x2 unidades no mundo. A base, assim como a mina, é identificada pelo seu ID no mundo.

As bandeiras, ou *flags*, são entidades que serão "colocadas" e/ou "retiradas" pelos agentes robôs para marcar o caminho entre as minas e a base. Estes *flags* podem ser sobrepostos. Assim, a mesma posição do mundo pode conter de 0 a n bandeiras. Desta forma, o número de bandeiras naquela posição do mundo é determinado pela subtração do ID do *flag* do valor armazenado no vetor mundo na posição desejada. O sistema aceita até 43.690 flags sobrepostos.

Os elementos que compõem o ambiente e os agentes são criados e inicializados através de chamadas para funções específicas que obedecem a uma ordem.

Inicialmente, cria-se o mundo e inicializa-se com todas as posições vazias (0x0000). A seguir é criada a base no centro e as minas em posições fixas no mundo. Por último são criados os robôs em posições aleatórias nos espaços vazios que restaram. Varia-se a quantidade de robôs alterando a constante *MAXROBO*.

Cada um destes elementos ocupa sua posição no mundo e o sistema não permite que sejam sobrepostos, excetuando-se as bandeiras que podem existir na mesma posição do agente robô.

A percepção do agente está ajustada para duas unidades de distância em qualquer direção. Assim o agente só consegue detectar as minas e as pistas deixadas por outros agentes quando estão a duas unidades da sua posição atual, conforme exemplificado na figura 01.

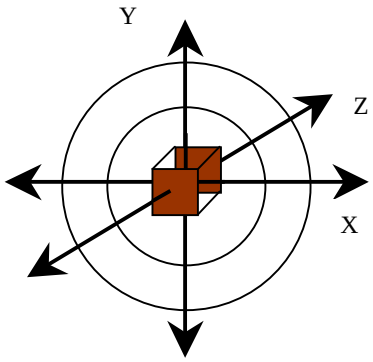

Figura 01 – Percepção do agente.

O grau de percepção do agente pode ser aumentado ou diminuído, de acordo com as características dos agentes e da simulação.

Neste sistema, a posição da base é a única conhecida pelos agentes. Logo, quando o agente está carregado ele dirige-se imediatamente ao centro do mundo – onde está a base – para descarregar.

As funções de percepção e movimentação foram desenvolvidas de maneira a generalizar o máximo possível o comportamento dos agentes. Esta generalização é regra básica, pois no mesmo ambiente estão sendo testados diversos algoritmos para a solução do problema proposto e, em breve, serão aplicados algoritmos genéticos para procurar novas soluções.

DESENVOLVIMENTO DO SISTEMA

O programa desenvolvido para a simulação de multiagentes implementa funções e procedimentos que permitem ao programador criar um mundo virtual tridimensional, definir uma base, minas e agentes robôs que atuarão neste mundo.

Para visualizar a evolução dos agentes no mundo, suas ações e reações, utilizou-se no sistema uma câmera sintética. A técnica da câmera sintética permite representar cenas tridimensionais em superficies bidimensionais. Faz-se analogia com a câmera fotográfica tradicional, onde a fotografia gerada no filme é a representação em duas dimensões do ambiente que está sendo fotografado, o qual está em três dimensões.

Alterando a posição da câmera fotográfica obtem-se diferentes visões do objeto a ser fotografado. Da mesma forma, no sistema desenvolvido, a câmera sintética permite ao observador da simulação posicionar a “lente” em qualquer ângulo ao redor do centro do mundo (*ponto focal*), bem como aproximar e recuar a “lente”, permitindo observar detalhes da simulação. Na figura 02 tem-se o *front-end* do sistema.

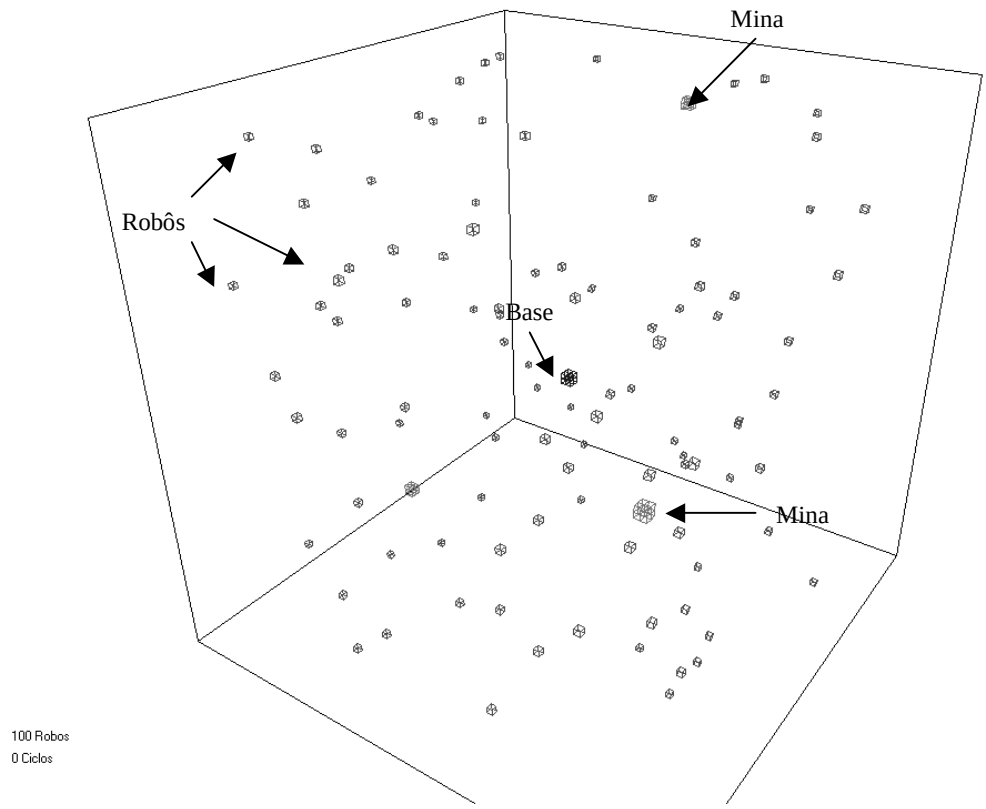

Figura 02 – *Front-end* do sistema de simulação 3D.

No atual estágio de desenvolvimento, é necessário um mínimo de conhecimento de programação na linguagem *object pascal* (*Delphi*) para programar um agente que possa atuar no mundo. Como projeto futuro tem-se a idéia de um interpretador de macro-comandos que possa facilitar a criação de protótipos de agentes, aplicando-se uma linguagem mais natural com recursos gráficos para facilitar esta modelagem.

Mesmo assim, a criação é bastante fácil, pois as principais ações e reações dos agentes já estão definidas no sistema, bem como as rotinas de criação, inicialização, movimento e destruição.

Com estas funções, serão implementados algoritmos originalmente modelados em sistemas bidimensionais, como *Petit-Poucet*, *Dockers* (DROGOUL, 1993) e *Algoritmo Evoluído* (CORDENONSI, 2000), agora em ambiente tridimensional.

CONCLUSÕES

A modelagem de sistemas multiagentes é uma área bastante explorada da IA, principalmente para ambientes bidimensionais. Este sistema propõe uma aproximação destas simulações com a realidade do nosso mundo tridimensional. Espera-se chegar a algoritmos mais funcionais e aplicáveis na solução de problemas no mundo real.

A técnica da câmera sintética permite uma visualização clara por todos os ângulos do sistema, facilitando o acompanhamento da simulação. Além do acompanhamento visual, uma análise dos resultados da simulação pode ser feita a partir de dados estatísticos gerados pelo sistema como: quantidade de agentes envolvidos na simulação, o grau de utilização de cada agente, quantidade de ciclos necessários para o fim da simulação, entre outras.

O sistema encontra-se em desenvolvimento e já é possível a modelagem e a simulação de agentes como os citados anteriormente. Como tarefas futuras tem-se a necessidade de aperfeiçoar o sistema tornando mais simples a criação de agentes por pessoas não conhecedoras de alguma linguagem de programação e a aplicação da técnica de evolução genética para o desenvolvimento de novos agentes.

REFERÊNCIAS BIBLIOGRÁFICAS

ALVARES, L.O.; SICHMAN, J. *Introdução aos Sistemas Multiagentes.* In: JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA, 16.;CONGRESSO DA SBC, 17., 1997, Brasília. Anais... Brasília: SBC,1997. p.1-38.

BROOKS, R. A. *A Robust Layered Control for a Mobile Robot*. IEEE Journal of Robotics and Automation, New York, v. 2, n. 1, Mar.1986.

CORDENONSI, A.Z. *Um Ambiente de Evolução de Comportamentos para Sistemas Multiagentes Reativos.* Porto Alegre: CPGCC da UFRGS, 2000. Dissertação de mestrado.

DEMAZEAU, Y. *From interactions to collective behaviour agent-based system.* In: EUROPEAN CONFERENCE ON COGNITIVE SCIENCE, 1., 1995, St. Malo. Proceedings. France: [s.n], 1995.

DROGOUL, A. *De la Simulation Multi-Agents à la Résolution Collective de Problèmes.* Paris, France: Université Paris VI, 1993.
Thèse de Doctorat.

FERBER, J.; GASSER, L. *Intelligence artificielle distribuée*. In: INTERNATIONAL WORKSHOP ON EXPERT SYSTEMS & THEIR APPLICATIONS, 10., 1991, Avignon. Cours n. 9. France:[s.n], 1991.

FROZZA, R. *SIMULA – Ambiente para Desenvolvimento de Sistemas Multiagentes Reativos.* Porto Alegre: CPGCC da UFRGS, 1997. Dissertação de mestrado.

LATTAUD, C.; CUENCA, C. In: HEUDIN, J.C.(Ed.)*Virtual World*. Berlin: Springler-Verlag, 1998, p.218-228.

PFEIFER, R. *Building "Fungus Eaters": Design Principles of Autonomous Agentes*, In: INTERNATIONAL CONFERENCE ON SIMULATION OF ADAPTIVE BEHAVIOR, 4., 1996. Proceedings... Cambridge: MIT, 1996. p. 3-12.

REYNOLDS, C. W. *Flocks, Herds and Schools: A Distributed Behaviorial Model.* In: SIGGRAPH, 1987. Proceedings.[S.l.:s.n], 1987. p. 25-34.

STEELS, L. *The Artificial life roots of artificial intelligence.* Artificial Life Journal, Cambridge, v.1, n.1, p. 75-110, 1994.