



Pós-Graduação em Ciência da Computação

**"Negociação em Sistemas Multiagentes para
Patrulhamento"**

Por

Talita Rodrigues de Menezes

Dissertação de Mestrado



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE, AGOSTO/2006

Universidade Federal de Pernambuco
Centro de Informática

Talita Rodrigues de Menezes
**Negociação em Sistemas Multiagentes para
Patrulhamento**

*Trabalho apresentado ao Programa de Mestrado
em Ciência da Computação do Centro de
Informática da Universidade Federal de
Pernambuco, como requisito parcial para obtenção
do grau de Mestre em Ciência da Computação.*

Orientadora: Patrícia C. de A. R. Tedesco
Co-orientador: Geber Lisboa Ramalho

Recife,
Agosto 2006

Menezes, Talita Rodrigues de

Negociação em sistemas multiagentes para patrulhamento / Talita Rodrigues de Menezes. – Recife: O autor , 2006.

89 folhas : il., fig., quadros.

Dissertação (mestrado) – Universidade Federal de Pernambuco. CIN. Ciência da Computação, 2006.

Inclui bibliografia.

1. Inteligência artificial. 2. Sistemas Multiagentes - Coordenação. 3. Patrulhamento multiagente. 4. Negociação. I. Título.

006.3

CDD (22.ed.)

CIN2006-025

À minha família e amigos que sempre me apoiaram.

AGRADECIMENTOS

Agradeço a todos os que colaboraram, direta ou indiretamente, para que eu conseguisse concluir este trabalho, bem como para minha formação pessoal, acadêmica e profissional.

Em especial à minha família, aos meus pais, Telmo e Inez Menezes pelo carinho e incentivo constante. Aos meus irmãos Larissa e Telmo por terem agüentado o meu estresse e o meu confisco do computador.

Aos meus professores e orientadores Geber e Patrícia pela orientação, disponibilidade, amizade e, principalmente, pela confiança em mim depositada.

A Eduardo por ter compreendido minha ausências e me apoiado. Por estar sempre disposto a me ajudar. Pelo amor, presença, paciência e disponibilidade.

Muito obrigada a Hugo e Alessandro, pela ajuda e por voltarem ao passado, simulando seus agentes em novos mapas.

Ao Centro de Informática da Universidade Federal de Pernambuco pela estrutura e suporte.

RESUMO

A Patrulha multiagente é uma tarefa onde os agentes devem, coordenadamente, visitar lugares, representados por nós de um grafo, a fim de protegê-los ou supervisioná-los. Para tanto, os agentes devem minimizar a ociosidade do conjunto dos nós do grafo, i.e., minimizar o tempo passado entre uma visita e outra a cada nó. Sistemas multiagente podem ser usados para realizar a tarefa da patrulha em domínios onde vigilância, inspeção ou controle distribuído são requeridos.

Ao distribuírem entre si os nós que formam o grafo do mundo, os agentes negociadores reduzem o caminho que tem que percorrer e o número de nós a patrulhar, tornando mais fácil manter baixa a ociosidade média dos nós do mundo. Leilões formam os mecanismos de negociação usados para dividir os nós do mundo. Neste trabalho, os resultados obtidos pelo novo tipo de abordagem baseada em negociação são comparados aos resultados de trabalhos anteriores.

Além de propor uma nova abordagem de agentes patrulhadores baseada em negociação, este trabalho propõe novos critérios de avaliação de desempenho. (1) Escalabilidade que avalia se a abordagem é capaz de patrulhar mundos ou grafos de todos os tamanhos. (2) Estabilidade mede a variação na ociosidade do grafo. (3) Adaptabilidade *offline* que avalia a capacidade das abordagens de patrulhar em vários mundos de topologias diferentes sem a necessidade de definição de estratégia específica para cada grafo a ser patrulhado, ou seja, avalia a necessidade de pré-processamento das abordagens. (4) Adaptabilidade *online* também avalia a capacidade da abordagem de lidar com modificações do grafo durante a simulação.

Uma avaliação empírica mostrou a eficiência desta abordagem distribuída quando comparada às abordagens anteriores. Os sistemas de agentes apresentados neste trabalho são estáveis, o que será demonstrado pelo baixo desvio padrão obtido na ociosidade dos nós. Além disso, os sistemas são mais adaptáveis e possuem uma maior escalabilidade, já que podem realizar a patrulha em mundos de vários tamanhos e topologias.

Palavras-chave: sistemas multiagentes, coordenação de agentes, patrulha e exploração de terrenos, negociação, grafos.

ABSTRACT

Multi-agent patrolling is a task where the agents must, coordinately, visit places, represented by nodes of a graph, to protect or supervise them. Thus, the agents must minimize the idleness of the set of nodes of the graph, i.e., to minimize the time passed between couple of visits to each node. Multi-agent systems can be used to perform the patrolling task in domains where monitoring, inspection or distributed control is required. In this work, the results obtained for this new negotiation-based approach are compared with the results of previous works.

By splitting the nodes of the world graph, the agents reduce the path they have to walk and their number of nodes to patrol, making it easier to maintain a low average idleness in world nodes. Auctions are the negotiation mechanisms used to split the world nodes.

Besides the evaluation criteria considered by previous works, there are other important measures that had not been considered yet. The criteria considered in this work are: scalability, stability and adaptability. The scalability criterion evaluates if the approach is capable of patrolling worlds or graphs of all sizes. The stability criterion measures the variation in the graph idleness. Adaptability (offline) evaluates the capacity of the approaches to patrol in worlds of different topologies without the need of defining specific strategies for each graph to be patrolled. In other words, it evaluates the necessity of processing before simulating. Adaptability (online), also, evaluates the capacity of the approach to deal with graph changes during the simulation.

Empirical evaluation has shown the effectiveness of this distributed approach, as the results obtained are substantially better than those previously achieved. The agent systems presented in this work are more stable as indicated by the low standard deviation obtained in node idleness. Besides, they are more adaptable and scalable since they can perform patrolling in worlds of all sizes and topology types.

Keywords: multi-agent systems, coordination of agents, computer games, graphs.

SUMÁRIO

<u>INTRODUÇÃO</u>	<u>1</u>	
1.1	INTRODUÇÃO	3
1.2	TRABALHO DESENVOLVIDO	4
1.3	ESTRUTURA DESTA DISSERTAÇÃO.....	5
<u>O PROBLEMA DA PATRULHA.....</u>	<u>8</u>	
2.1	PATRULHAS	9
2.1.1	DESCRÍÇÃO DA TAREFA	10
2.2	CRITÉRIOS DE AVALIAÇÃO	12
2.2.1	CRITÉRIOS INICIALMENTE PROPOSTOS	12
2.2.2	CRITÉRIOS DE AVALIAÇÃO PROPOSTOS NESTE TRABALHO.....	14
2.3	SISTEMAS MULTIAGENTES PARA PATRULHA	15
2.3.1	ABORDAGEM PIONEIRA DE PATRULHA MULTIAGENTE: AGENTES SIMPLES	18
2.3.2	ABORDAGEM BASEADA EM UTILIDADE	20
2.3.3	ABORDAGEM BASEADA EM APRENDIZAGEM POR REFORÇO	21
2.3.4	ESTRATÉGIA DO CICLO ÚNICO	23
2.3.5	ABORDAGEM BASEADA NA TEORIA DOS CAMPOS POTENCIAIS.....	24
2.4	AVALIAÇÃO DAS ABORDAGENS SEGUNDO OS NOVOS CRITÉRIOS	25
2.5	CONCLUSÃO.....	26
<u>NEGOCIAÇÃO</u>	<u>27</u>	
3.1	NEGOCIAÇÃO.....	29
3.2	LEILÕES	30
3.2.1	CARACTERÍSTICAS DOS LEILÕES.....	35
3.2.2	TIPOS MAIS CONHECIDOS DE LEILÕES	37
3.2.3	LEILÕES EM SMA	40
3.3	CONCLUSÃO.....	41
<u>NEGOCIAÇÃO NA PATRULHA.....</u>	<u>43</u>	

4.1	NEGOCIAÇÃO NA PATRULHA	45
4.2	DESCRÍÇÃO DA ABORDAGEM UTILIZADA	46
4.3	NEGOCIAÇÃO.....	47
4.4	COMUNICAÇÃO	48
4.5	CONFIGURAÇÕES DOS AGENTES NEGOCIADORES.....	49
4.5.1	COMPORTAMENTO.....	50
4.5.2	O PROTOCOLO DE LEILÃO	50
4.5.3	ESCOLHA DO NÓ A SER LEILOADO.....	50
4.5.4	ALGORITMO DE PERCURSO.....	53
4.6	TIPOS DE AGENTES	54
4.6.1	AGENTE LEILOEIRO	54
4.6.2	AGENTE LEILOEIRO COOPERATIVO.....	54
4.6.3	AGENTE NEGOCIADOR EM ENCONTROS	55
4.6.4	AGENTE FACILITADOR DE TROCAS	55
4.6.5	AGENTE LEILOEIRO DE DOIS LANCES	56
4.6.6	AGENTE LEILOEIRO FLEXÍVEL	56
4.7	CONCLUSÃO.....	56
<u>EXPERIMENTOS E RESULTADOS.....</u>		<u>59</u>
5.1	INTRODUÇÃO.....	61
5.2	HIPÓTESES.....	61
5.3	IMPLEMENTAÇÃO	62
5.4	CENÁRIOS DE SIMULAÇÃO	62
5.5	ENCONTRANDO O LEILOEIRO DE MELHOR DESEMPENHO.....	65
5.5.1	HIPÓTESE 1 – MELHOR LEILOEIRO.....	65
5.5.2	HIPÓTESE 2 - AGENTES BASEADOS EM MCST OBTÊM MELHORES RESULTADOS.....	67
5.5.3	HIPÓTESE 3 – FILA DE PRIORIDADES MELHORA O DESEMPENHO DOS AGENTES	69
5.5.4	HIPÓTESE 4 – USO DE OCIOSIDADE E DISTÂNCIA É A MELHOR ESTRATÉGIA NA ESCOLHA DE PRÓXIMO NÓ A SER VISITADO	70
5.5.5	HIPÓTESE 5 – USO DE OCIOSIDADE E DISTÂNCIA É A MELHOR ESTRATÉGIA NA ESCOLHA DO NÓ A SER LEILOADO	72
5.5.6	DISCUSSÃO	74
5.6	ANÁLISE COMPARATIVA ENTRE ABORDAGENS.....	74

5.7	CONCLUSÃO.....	78
<u>CONCLUSÃO E TRABALHOS FUTUROS.....</u>		<u>81</u>
6.1	RESULTADOS ALCANÇADOS.....	83
6.2	DIFICULDADES ENCONTRADAS	84
6.3	TRABALHOS FUTUROS	84

LISTA DE ILUSTRAÇÕES

Figura 2.1 – Interface Gráfica do <i>Agent Simulator</i>	11
Figura 2-2– Execução da estratégia de ciclo único por dois agentes. Os agentes percorrem o mesmo caminho, separados por um intervalo de tempo	23
Figura 4-1 – Estrutura das Mensagem Utilizadas Neste Trabalho	49
Figura 4-2 – Algoritmo de Kruskal	51
Figura 5-1 – Mapas utilizados nos experimentos	64
Figura 5-2– Mapas utilizados nos experimentos	64
Figura 5-3– Comparação entre desempenho dos agentes (BA, CBA, TSBA, FBA) em termos de ociosidade média, na simulação com população de 5 agentes	66
Figura 5-4– Comparação entre desempenho dos agentes (BA, CBA, TSBA, FBA) em termos de ociosidade média, na simulação com população de 15 agentes	66
Figura 5-5 – Comparação entre desempenho da Ociosidade média dos agentes (BA, MBA, MMBA) na simulação com população de 5 agentes	68
Figura 5-6 – Comparação entre desempenho da Ociosidade média dos agentes (BA, MBA, MMBA) na simulação com população de 15 agentes	68
Figura 5-7– Comparação do desempenho da Ociosidade média dos agentes (RFBA, PQFBA) na simulação com população de 5 agentes	69
Figura 5-8– Comparação do desempenho da Ociosidade média dos agentes (RFBA, PQFBA) na simulação com população de 15 agentes	70
Figura 5-9– Comparação do desempenho da Ociosidade média dos agentes (LFBA, CFBA, HFBA) na simulação com população de 5 agentes	71
Figura 5-10– Comparação do desempenho da Ociosidade média dos agentes (LFBA, CFBA, HFBA) na simulação com população de 15 agentes	71
Figura 5-11 – Comparação do desempenho da Ociosidade média dos agentes (FBA, IFBA) na simulação com população de 5 agentes	73
Figura 5-12 – Comparação do desempenho da Ociosidade média dos agentes (FBA, IFBA) na simulação com população de 15 agentes	73
Figura 5-13 – Comparação entre desempenho dos agentes (SC, HPCC, GBLA, FBA) quanto a Ociosidade média em simulações com 5 agentes	75
Figura 5-14 – Comparação entre desempenho dos agentes (SC, HPCC, GBLA, FBA) quanto a Ociosidade média em simulações com 15 agentes	75
Figura 5-15 – Variação do desvio padrão da ociosidade média. Estes resultados são a média para todos os mapas	76
Figura 5-16 – Quantidade de ciclos necessária para os agentes FBA e HPCC se estabilizarem. Simulação com cinco agentes no mapa Ilhas	77

Figura 5-17 – Resultados das simulações em mapas grandes. No Mapa de 100 nós, a população era de 10 agentes, no Mapa de 200 nós, a população era de 20 agentes e no Mapa de 500 nós de 50 agentes _____ 78

LISTA DE EQUAÇÕES

Equação 1	12
Equação 2	13
Equação 3	13
Equação 4	13
Equação 5	13
Equação 6	13
Equação 7	51
Equação 8	52
Equação 9	52

LISTA DE QUADROS

Quadro 1 - Resumo das principais características das abordagens propostas _____ 18

Capítulo 1

Introdução

Neste capítulo, serão apresentadas as principais motivações para realização deste trabalho. Serão também mostrados os seus objetivos e a estrutura do documento.

1.1 INTRODUÇÃO

A Patrulha multiagente é uma tarefa onde os agentes devem, coordenadamente, visitar lugares, em intervalos regulares, a fim de proteger ou supervisionar estas locações [1]. Informalmente, uma boa estratégia de patrulhamento deve minimizar o espaço de tempo entre duas visitas ao mesmo lugar ou a todos os lugares.

A patrulha de segurança, foco deste trabalho e doravante chamada apenas patrulha, pode ser útil em domínios que requerem inspeção ou controle distribuído. Por exemplo, agentes patrulhadores podem ser usados para ajudar administradores de redes a prevenir ou detectar rapidamente falhas ou situações específicas em uma Intranet [2]. Estes agentes também podem ser usados para detectar mudanças recentes ou novas páginas para serem indexadas por um motor de busca [3], para detectar objetos ou pessoas em situações perigosas, que podem ser resgatadas por robôs [4], para encontrar ofertas em sites de leilão, e até para simular agentes de saúde visitando, da melhor maneira, uma área para tentar identificar e eliminar possíveis focos de doenças, como dengue ou filariose. Patrulhas também podem ser vistas como missões para conseguir informações ou conduzir operações de combate [5]. Enfim, e em particular, a patrulha é útil em diversos gêneros de jogos digitais (*games*)[6].

O grupo de pesquisa na área de Agentes Inteligentes e Jogos do Centro de Informática (CIn) [7] vem aprofundando pesquisas na área de sistemas multiagentes já há alguns anos. Dentre os problemas de pesquisa investigados, o grupo iniciou um estudo pioneiro do problema da patrulha em 2002 [6].

Os trabalhos anteriores em patrulhamento multiagente desse grupo desenvolveram, inicialmente, um estudo sistemático do problema e das suas aplicações, e diversas estratégias para sua resolução foram propostas, sempre usando, por razões metodológicas, modelos de agentes bastante simples [1, 6]. Nestes mesmos trabalhos, foram definidos critérios de avaliação de desempenho de soluções, e se construiu um simulador para realizar tais avaliações. Em trabalhos posteriores [8, 9], soluções mais sofisticadas dos agentes pioneiros foram propostas e avaliadas, e instâncias mais complexas do problema foram tratadas, com técnicas diferentes de exploração, variando

da aprendizagem de máquina aos agentes baseados em utilidade e à pesquisa de operação. Mais recentemente, foi estudado o problema de patrulhar um terreno real com robôs [10].

Uma comparação entre estes trabalhos mostrou que nenhuma das soluções propostas é claramente a melhor com respeito a todos os critérios para uma boa solução de patrulhamento [8]. Em particular, a maioria das soluções apresentou problemas de escalabilidade, já que consomem muito tempo, memória ou ambos nas simulações em mapas grandes. Há também a questão da flexibilidade, já que algumas abordagens precisam conhecer previamente o mapa que será patrulado e não foram desenvolvidas de forma a se adaptarem a mudanças em tempo de execução.

Motivados por estas constatações, decidimos desenvolver uma abordagem de agentes patrulhadores que fosse, em princípio, capaz de lidar com mapas de quaisquer tamanhos e topologias e capaz de se adaptar a mudanças no ambiente em tempo de execução. Também seria necessário que esta abordagem se mostrasse estável, ou seja, fosse capaz de se espalhar pelo ambiente e não deixar que nenhuma região fosse menos supervisionada que outras com mesma prioridade.

1.2 TRABALHO DESENVOLVIDO

Este trabalho propõe algumas soluções para o problema da patrulha de vigilância baseadas em negociação para a coordenação de movimentação em sistemas multiagentes. Os agentes considerados têm apenas informações parciais sobre o ambiente onde estão situados, e estão dispostos a compartilhar suas informações. Assim sendo, consideramos todos os agentes como confiáveis.

Este trabalho também propõe novos critérios de avaliação de sistemas multiagentes para a tarefa da patrulha. É apresentada uma discussão a respeito de escalabilidade, adaptabilidade e estabilidade.

A patrulha é uma tarefa realizada por agentes autônomos. Assim, um agente não pode forçar outro agente a realizar um serviço, ou modificar seu estado interno. Ou seja, um agente deve tentar convencer o outro a cooperar com ele. Para tanto, eles precisam negociar. Negociação é o processo que permite que grupos de agentes se comuniquem entre si para tentar estabelecer um acordo, mutuamente aceitável, para a solução de um

dado problema. Isto é, este processo tem por objetivo possibilitar que os agentes colaboradores atinjam um consenso na distribuição de tarefas e recursos [11].

Os agentes negociadores propostos não necessitam de ajuste para cada situação, nem requerem treinamento antecipado, que pode gastar períodos de tempo proibitivos em grandes mapas. Além disso, não requerem esquemas centralizados e explícitos de coordenação. Os sistemas multiagentes dos trabalhos anteriores [1, 6, 12, 13] tinham como restrição a falta de comunicação direta entre os agentes, havendo comunicação entre o agente coordenador e os coordenados em [1, 6, 8, 12], e comunicação por *flags* em [13]. Neste trabalho, os agentes se comunicam entre si, sem restrição, e fazem uso da teoria dos atos de fala [14] para se comunicarem durante o patrulhamento. De fato, os sistemas multiagentes negociadores para a patrulha são sistemas adaptáveis que podem supervisionar grafos com regiões e mudanças de prioridade e topologia durante simulações. Como esta abordagem tem por intuito particionar o grafo, há um sentimento que esta abordagem será mais estável do que as anteriores.

Os agentes negociadores são integrados com técnicas de decisão avançadas, leilões complexos e técnicas de *pathfinding*, a fim de se confirmar sua capacidade de formarem sistemas de patrulhamento escaláveis, estáveis e adaptáveis.

1.3 ESTRUTURA DESTA DISSERTAÇÃO

O restante deste documento está organizado como descrito a seguir. No próximo capítulo (2), é descrito o problema da patrulha e é dada uma visão geral do estado da arte de pesquisas sobre a tarefa de patrulhamento e áreas relacionadas.

No capítulo 3, são explicados os principais conceitos de negociação e como estes podem ser aplicados para solução do problema da patrulha.

No capítulo 4, são descritos a metodologia utilizada, a abordagem implementada e os critérios de avaliação propostos neste trabalho.

O capítulo 5 apresenta os resultados obtidos e uma discussão sobre eles.

Por fim, o capítulo 6 apresenta as conclusões e sugestões para trabalhos futuros.

Capítulo 2

O Problema da Patrulha

Neste capítulo, são descritos o problema da patrulha, os esforços feitos em trabalhos anteriores, e as métricas utilizadas na avaliação das soluções propostas.

2.1 PATRULHAS

Patrulhar é, literalmente, “o ato de andar ou viajar por uma área, em intervalos regulares, para protegê-la ou supervisioná-la” [15]. Patrulhas também podem ser vistas como missões para conseguir informações ou conduzir operações de combate [5]. Segundo [5], há cinco tipos de patrulha, descritos a seguir.

- Reconhecimento: este tipo de patrulha se subdivide em reconhecimento de área e de zona. No reconhecimento de área, o objetivo é obter informação de uma locação específica (por exemplo, um prédio) de um inimigo e a área diretamente adjacente a ela. Já o reconhecimento de zona visa obter informações mais abrangentes sobre o inimigo, como análise do terreno e rotas.
- Incursão: o objetivo desta patrulha é atacar uma locação para destruí-la, libertar prisioneiros, ou capturar tropas e equipamentos.
- Emboscada: este tipo de patrulha tem como intuito atacar de surpresa, a partir de um esconderijo, um alvo móvel ou temporariamente parado.
- Detecção: o problema da detecção parte do pressuposto de que no ambiente existe algum alvo (estático ou em movimento) que precisa ser encontrado. A tarefa de patrulhamento consiste em, dados N agentes, detectar o alvo no menor tempo possível [12]. Na literatura, esse problema se chama *pursuit-evasion* [16], no qual os agentes patrulhadores são chamados de *pursuers* e os alvos, de *evaders*.
- Patrulha de segurança ou vigilância: esta patrulha, foco deste trabalho e doravante chamada apenas patrulha, tem como objetivo supervisionar uma área própria, para protegê-la de infiltrações e ataques surpresa.

A execução eficiente da tarefa de patrulhamento é necessária em diversas aplicações práticas como, por exemplo, para execução do trabalho de agentes de saúde no intuito de identificar e tratar novos focos de doenças, sugerir e acompanhar tratamentos de pacientes em comunidades carentes. A patrulha também pode ser utilizada para detecção de falhas ou situações específicas em uma rede de computadores [2], para identificação de objetos ou pessoas que precisam ser resgatados por robôs [4].

As várias formas de patrulha podem ser usadas em jogos de computador, no controle de entidades virtuais que precisam executar tal tarefa como: em jogos de estratégia (e.g., como Civilization IV [17]) e RPGs (e.g., como Diablo [18]). Nesses jogos, a patrulha serve para detectar personagens móveis e novos prédios inimigos, proteger cidades e recursos. Em jogos de combate, a patrulha pode ser utilizada para a detecção de inimigos e descoberta de itens, entre outros.

Tendo visto que, dado um ambiente, a tarefa de patrulhamento consiste em visitar continuamente pontos estratégicos visando minimizar o intervalo de tempo entre visitas sucessivas, veremos agora como os trabalhos pioneiros neste problema [1, 6] definiram esta tarefa, de modo a criar uma generalização que permita a simulação tanto de terrenos e prédios, como também de redes de computadores.

2.1.1 Descrição da Tarefa

O trabalho pioneiro do CIn [7] na área [1, 6] reduziu o problema da patrulha, em qualquer ambiente, ao patrulhamento de grafos com as seguintes características:

- as arestas são estáticas (não há obstáculos móveis no terreno);
- as arestas não têm pesos (em todos os trabalhos posteriores do CIn [7], as arestas passaram a ter pesos representando a distância real entre os nós);
- a velocidade de cada agente é igual a uma unidade de distância por unidade de tempo;
- o patrulhamento é uniforme, ou seja, todos os nós têm a mesma prioridade.

Esta redução do problema da patrulha foi usada para simplificar a tarefa e permitir o aumento gradativo de dificuldade na implementação dos agentes.

Para simplificar, os agentes se movem, com a mesma velocidade, uma unidade de distância (*tile*) por unidade de tempo (ciclo). Caso as arestas não fossem estáticas ou houvesse obstáculos móveis no terreno (simulando quedas de barreiras, acidentes de veículos, entre outros) os algoritmos de *pathfinding* [19] dos agentes seriam mais complexos e o desempenho da simulação cairia, já que não se poderia calcular a priori o caminho completo para achar um nó. Isto teria que ser feito aos poucos, pois o grafo poderia mudar enquanto o agente estivesse se movendo.

Também com o objetivo de simplificar, todos os nós têm a mesma prioridade. Caso houvesse nós de maior prioridade que outros, os agentes deveriam se concentrar vigiando mais estes nós e áreas próximas a eles.

Em outras palavras, dados N agentes e K nós conectados por arestas com peso, o problema da patrulha tratado consiste em atingir um comportamento global que minimize o intervalo de tempo entre visitas aos nós do grafo [1, 6].

Para testar e simular as abordagens implementadas, no trabalho pioneiro foi desenvolvido um simulador, o *Agent Simulator* [6]. A Figura 2.1 mostra a interface gráfica do simulador.



Figura 2.1 – Interface Gráfica do *Agent Simulator*

Este simulador agrupa os agentes e simula a tarefa da patrulha, armazenando dados para posterior análise. Os mapas interpretados por este simulador são descritos em formato proprietário, permitindo ao pesquisador indicar todas as características do ambiente, tais como o tamanho do mapa, os obstáculos, a arquitetura de agentes e a quantidade de agentes, as posições iniciais do agente e o número de ciclos a serem executados.

Nesta seção, a tarefa da patrulha foi definida e foi apresentado um simulador usado para simular as várias abordagens de sistemas multiagentes para patrulhamento. Na próxima seção, serão mostrados os critérios de avaliação propostos nos trabalhos pioneiros na área.

2.2 CRITÉRIOS DE AVALIAÇÃO

Uma das grandes contribuições do trabalho pioneiro em patrulhamento multiagente foi o estabelecimento de critérios de avaliação para comparar diferentes modelos de sistemas multiagentes [1, 6]. Alguns destes critérios de avaliação identificados foram: ociosidade, desvio padrão da ociosidade, pior ociosidade e tempo de exploração.

Este trabalho também propõe novos critérios de avaliação de sistemas multiagentes para a tarefa da patrulha: escalabilidade, adaptabilidade e estabilidade. Os novos critérios serão apresentados na seção 2.2.2.

2.2.1 Critérios Inicialmente Propostos

Considerando que um ciclo é o tempo necessário para o agente andar uma unidade de distância (*tile*) em direção ao nó destino, a *ociosidade instantânea do nó* é o número de ciclos que um nó permanece sem ser visitado por algum agente. Esta ociosidade instantânea é medida a cada ciclo. A *ociosidade instantânea do grafo* é a média das ociosidades de cada nó em um dado ciclo. E ainda, a *ociosidade do grafo*, ou simplesmente *ociosidade*, é a média das *ociosidades instantâneas do grafo* ao fim de uma simulação de vários ciclos. A *pior ociosidade* é o máximo valor da *ociosidade instantânea do grafo* durante toda a simulação. O *tempo de exploração* é a quantidade de ciclos necessária para que todos os nós do grafo tenham sido visitados pelo menos uma vez.

Matematicamente, a *ociosidade instantânea do grafo* no ciclo t O_t , sendo O_{it} a ociosidade instantânea do nó i no ciclo t e $nós$ a quantidade de nós do grafo, é dada por (Equação 1):

$$O_t = \frac{\sum_{i=1}^{nós} O_{it}}{nós}$$

Equação 1

Sendo O a *ociosidade*, O_t a ociosidade instantânea do grafo no ciclo t , e *ciclos* a quantidade de ciclos da simulação, a ociosidade (O) é dada por (Equação 2):

$$O = \frac{\sum_{t=1}^{\text{ciclos}} O_t}{\text{ciclos}}$$

Equação 2

O *desvio padrão da ociosidade* é medido como sendo a raiz quadrada da variância, a qual é dada pela Equação 3, sendo O a ociosidade, O_t a ociosidade instantânea do grafo no ciclo t , e *ciclos* a quantidade de ciclos da simulação.

$$\sigma^2 = \frac{\sum_{t=1}^{\text{ciclos}} (O_t - O)^2}{\text{ciclos} - 1}$$

Equação 3

A *pior ociosidade* pode ser definida pela Equação 4, sendo O_{it} a ociosidade instantânea do nó i no ciclo t , *nós* a quantidade de nós do grafo e *ciclos* a quantidade de ciclos da simulação.

$$\text{piorOciosidade} = \max_{i=1, t=1}^{\text{nós}, \text{ciclos}} \{O_{it}\}$$

Equação 4

O *tempo de exploração* pode ser mostrado matematicamente pela Equação 5.

$$t = \text{tempoExploração} \Leftrightarrow (\forall i : \text{NÓS} \bullet (O_{it} < t)) \wedge (\neg \exists t' \bullet (t' < t) \wedge (\forall i : \text{NÓS} \bullet (O_{it} < t')))$$

Equação 5

A tendência é que os resultados destas medições de desempenho sejam maximizados à medida que o número de agentes cresce, contudo a coordenação entre os agentes não for boa, esta melhoria pode não alcançar os níveis esperados.

Para medir a qualidade da coordenação, a contribuição individual de cada agente foi medida normalizando os quatro critérios (ociosidade, desvio padrão da ociosidade, pior ociosidade, e tempo de exploração), conforme apresentado na Equação 6.

$$\text{valorNormalizado} = \text{valorAbsoluto} \times \frac{\text{quantidade DeAgentes}}{\text{quantidade DeNós}}$$

Equação 6

2.2.2 Critérios de Avaliação Propostos neste Trabalho

Uma das contribuições principais de [6], o estudo pioneiro neste assunto, foi a identificação de critérios de avaliação para a tarefa da patrulha (Seção 2.2.1), contudo outras medidas importantes não foram consideradas ainda. Os critérios propostos neste trabalho são: escalabilidade, estabilidade e adaptabilidade discutidos em detalhes nas próximas seções.

Escalabilidade

Escalabilidade avalia se a abordagem é capaz de patrulhar mapas de quaisquer tamanhos. Trabalhos anteriores [1, 6, 8, 9, 13] haviam sempre simulado abordagens de sistemas multiagentes para a patrulha em mapas cujos grafos possuíam menos de uma centena de nós. Porém, é preciso considerar que os agentes devem ser capazes de patrulhar de forma eficiente (em termos de ociosidade) tanto o interior de um prédio, quanto uma cidade inteira, caso dos agentes sanitários, por exemplo.

Estabilidade

Uma abordagem SMA para patrulhamento é instável quando em determinados momentos ela patrulha bem o terreno e em outros patrulha muito mal, apresentando em alguns momentos ociosidades instantâneas do grafo muito altas. O critério de estabilidade na tarefa da patrulha mede a variação de ociosidade de cada nó no grafo do mundo. Além de considerar a ociosidade média, em algumas situações é necessário garantir que nenhum nó terá ociosidade muito mais alta que os demais para que mudanças ou problemas sejam rapidamente detectados.

Adaptabilidade

Adaptabilidade pode ser medida antes (*offline*) e durante (*online*) o patrulhamento. Este critério avalia a capacidade da abordagem multiagente de patrulhar vários mapas de diferentes topologias sem necessidade de reajuste, ou redefinição de estratégia para cada mapa. Esse critério avalia a flexibilidade das abordagens e pode ser necessário, por exemplo, em jogos com vários níveis. Se os agentes são adaptáveis, basta uma compilação, e eles executarão seu patrulhamento em qualquer estágio do jogo. Ou seja, adaptabilidade *offline* avalia quão imediata é a solução para um novo problema.

Adaptabilidade *online* mede a capacidade da abordagem de se adequar a mudanças do mundo durante a simulação. Este critério, por exemplo, pode ser aplicado em jogos de computador onde possa haver mudanças, como destruição de uma ponte, o que causaria a remoção de uma aresta no grafo que representa o mundo. Ele também pode ser aplicado no cotidiano dos agentes de saúde, caso por aterramento ou alagamento, em época de chuvas, acesso a algumas ruas seja modificado, por exemplo.

Também pode acontecer de certas regiões do grafo se tornarem mais prioritárias, exigindo uma patrulha mais constante, como no caso de agentes de saúde detectarem um novo foco de dengue, que será destruído, mas precisará de vigilância constante neste ponto e em sua vizinhança.

Nesta seção foram apresentados os critérios de avaliação da tarefa da patrulha propostos nos trabalhos anteriores e neste trabalho. A próxima seção apresenta o estado da arte da pesquisa na área de sistemas multiagentes patrulheiros.

2.3 SISTEMAS MULTIAGENTES PARA PATRULHA

Alguns trabalhos relacionados com o tema, embora não voltados para resolver o problema da patrulha, foram encontrados, como o mapeamento de redes, o problema do bar El Farol, *steering behaviors* (ou comportamento de navegação) [2] e *The Art Gallery Theorem* (o teorema da galeria de arte) [20].

No mapeamento de redes, alguns trabalhos em agentes móveis que auxiliam o gerenciamento de rede foram desenvolvidos, por exemplo [21]. Na abordagem adotada por esses trabalhos, o gerenciamento não é mais centralizado no administrador da rede, e sim pulverizado e distribuído através dos agentes que auxiliam na detecção e correção de eventuais problemas. Em particular, a utilização de agentes móveis cooperativos para o mapeamento de redes de computadores se mostrou interessante para o escopo da patrulha pela metodologia adotada. Esse trabalho tirou proveito da mobilidade, modularidade, flexibilidade e adaptabilidade das abordagens multiagente, que podem ser criadas para realizar o mapeamento. Com este mapeamento realizado, decisões sobre rotas nessa rede podem ser tomadas de uma forma mais eficiente e segura.

O problema do “Bar El Farol” [22] é um problema de previsão definido da seguinte forma: N pessoas decidem de forma independente, a cada semana, se elas devem ir ou não a um bar que oferece entretenimento em certa noite. O espaço é limitado, e somente é divertido se o bar não estiver muito cheio. Não existe a possibilidade de se saber, a priori, o número de pessoas que irão para o bar. Não existe a possibilidade de comunicação entre as pessoas. A única informação disponível é o número de pessoas que foram para o bar nas semanas anteriores. Assim, como cada pessoa ou agente deve atuar de maneira a maximizar o divertimento e minimizar as saídas frustradas?

O problema do “Bar El Farol” é similar ao da patrulha, na medida em que a proposta é deixar o número de visitas abaixo de um certo limiar de aceitação. Poderia-se imaginar um bar em cada nó do grafo. No caso da patrulha, o objetivo é deixar a ociosidade no seu valor mínimo. Este problema mostra uma solução interessante, mas apenas para esse tipo de aplicação onde apareçam todas as restrições que foram descritas com o problema, como é o caso da existência de um histórico dos acontecimentos anteriores. Essa abordagem também não leva em consideração os outros parâmetros usados nas abordagens de agentes discutidas nos trabalhos sobre agentes patrulhadores. Alguns deles são: comunicação, monitoramento, planejamento, coordenação.

Uma outra abordagem, baseada em comportamentos, foi encontrada para solucionar os problemas de movimentação ou navegação. Tal abordagem foi batizada pelos pesquisadores dessa área como “comportamentos de navegação” [23-25]. A capacidade de navegação é conseguida, nestes sistemas, através da composição de comportamentos simples. Esses comportamentos são baseados na informação sensorial instantânea, e são canalizados de forma imediata para o sistema motor, responsável pela locomoção. Isso mostra a natureza reativa desses comportamentos. As pesquisas dessa área têm como principal preocupação a criação de novos comportamentos, bem como sua adaptação e utilização nos ambientes mais variados.

Essas técnicas de movimentação mostram ótimos resultados para movimentar agentes em ambientes hostis. Isso se dá graças à reatividade desses agentes, que é muito útil em situações imprevisíveis que exigem uma decisão rápida, sendo esse o principal objetivo dessa área. Tais características podem ser utilizadas para implementar a

movimentação dos agentes para uma patrulha, sendo esta uma solução sem muito recurso, que não dá o devido tratamento para o problema proposto neste trabalho.

O teorema da galeria de arte está relacionado à colocação de câmeras em locais específicos, ou móveis, de modo a impedir o roubo de uma obra de arte, ou a invasão ou fuga de um elemento não autorizado no ambiente. O teorema da galeria de arte diz que, se a galeria é formada por n segmentos de linhas (formando um polígono), então toda a galeria estará protegida se $n/3$ câmeras foram colocadas em seus vértices [16]. Embora seja um problema relacionado à vigilância, essa abordagem também não leva em consideração os outros parâmetros colocados nas abordagens de agentes discutidas nos trabalhos sobre agente patrulhadores como coordenação, por exemplo.

Existem também pesquisas específicas na tarefa de patrulhamento para jogos[26-28], mas estas não avaliavam as estratégias de coordenação possíveis, abordagens de agentes ou organizações de sociedades de agentes. Nestes trabalhos, patrulhar consiste em mover-se entre uma lista de pontos especificados a priori, simulando assim uma patrulha. Existem algumas variações que procuram deixar o comportamento um pouco mais “inteligente”, tal como sortear o ponto inicial depois de percorrer todos os pontos da lista, sortear novamente o primeiro ponto e percorrer, assim por diante. Outra solução encontrada, para o mesmo problema, foi simplesmente deixar as unidades paradas enquanto não acontecia nada de relevante para elas.

Para preencher um pouco desta lacuna em pesquisas na área, o grupo de pesquisa em Inteligência Artificial e Jogos do Centro de Informática (CIn) da Universidade Federal de Pernambuco vem desenvolvendo pesquisas no assunto com trabalhos [1, 6, 8, 12, 13, 29] que procuraram investigar as seguintes questões [1, 6, 8, 12, 13, 29]. (1) Qual abordagem de sistema multiagente (SMA) deveria ser escolhida pelo projetista para uma determinada tarefa de patrulhamento; (2) Que critérios devem ser empregados na avaliação de desempenho de um SMA; (3) Até que ponto parâmetros como tamanho e conectividade do grafo influenciam o desempenho do sistema multiagente patrulheiro.

A seguir serão mostrados alguns trabalhos realizados para criar soluções eficientes de agentes patrulhadores.

2.3.1 Abordagem Pioneira de Patrulha Multiagente: Agentes Simples

Com o intuito de determinar que abordagens de sistemas multiagentes deveriam ser consideradas na realização da tarefa de patrulhar, bem como desenvolver um estudo sistemático do problema e das suas aplicações, diversas estratégias para resolução do problema da patrulha foram propostas no trabalho pioneiro nesta área [1, 6] sempre usando, por razões metodológicas, modelos de agentes bastante simples. Quatro parâmetros básicos foram explorados: presença de objetivo, comunicação, coordenação central e escolha do próximo nó. A idéia era partir de modelos simples e aumentar a complexidade aos poucos para evitar experimentações imaturas [1, 6]. Por esta razão, só foram consideradas abordagens homogêneas, onde os agentes patrulhadores são todos iguais. Nas três últimas abordagens existe um coordenador e vários patrulhadores, mas o coordenador não executa patrulhamento. As abordagens investigadas nestes trabalhos são apresentadas no Quadro 2.1.

Quadro 2.1 - Resumo das principais características das abordagens propostas

Nome	Tipo Básico	Comunicação	Escolha do próximo nó	Coordenação
<i>Random Reactive</i>	Reativo	Nenhuma	Randômica	Emergente
<i>Conscientious Reactive</i>			Ociosidade individual	
<i>Reactive with Flags</i>		Flags	Ociosidade coletiva	
<i>Conscientious Cognitive</i>	Cognitivo	Nenhuma	Ociosidade individual	Central
<i>Blackboard Cognitive</i>		Blackboard	Ociosidade coletiva	
<i>Random Coordinated</i>		Mensagens	Randômica	
<i>Cognitives Coordinated</i>			Ociosidade coletiva	

O primeiro parâmetro considerado foi a presença ou não de objetivos em cada agente. Esta é a principal diferença entre agentes reativos e agentes cognitivos: enquanto agentes reativos agem apenas baseados em suas percepções correntes, agentes cognitivos têm objetivos. No caso da patrulha, os agentes reativos estão limitados a perceber apenas os nós adjacentes, enquanto agentes cognitivos percebem todo o grafo, podendo escolher qualquer nó do grafo como objetivo e usar técnicas de *pathfinding* [30] para alcançá-lo.

Outro parâmetro importante quando executando uma tarefa coletivamente é a comunicação. Agentes podem se comunicar de três formas básicas: (1) através de sinais, que são modificações no ambiente, como marcas ou rastros (isto é, ao passar por um determinado nó, o agente pode deixar certas *flags* com informações para os próximos agentes, como por exemplo, seu identificador e o instante de tempo em que visitou o nó e deixou o flag); (2) *Blackboard* [19], que é uma estrutura de dados compartilhada por todos os agentes (nesse trabalho os agentes publicam no *blackboard* o nó onde se encontram, e os outros agentes podem verificar se o seu nó destino já foi visitado); e (3) troca de mensagens direta. Mas nesse trabalho [1], a comunicação através da troca de mensagens só era possível entre coordenador e agentes, (quando existia um coordenador que informa a cada agente que nó deve ser visitado).

Um aspecto também muito importante em coordenação em sistemas de movimentação multiagente é o uso de um coordenador central, (que nesse trabalho escolhe o próximo nó-objetivo de cada agente cognitivo), ou de um mecanismo descentralizado, onde a coordenação emerge da interação entre agentes. A vantagem da presença de um coordenador central é que este pode distribuir seus agentes de modo a não haver possibilidade de vários agentes resolverem se dirigir ao mesmo nó. Por outro lado, essa centralização cria um gargalo no sistema, e se o coordenador, por algum motivo, deixar de funcionar, todo o sistema estará comprometido.

Finalmente, o último parâmetro considerado foi a escolha do próximo nó. A forma como essa decisão é tomada é importantíssima para o bom desempenho do sistema, já que más escolhas podem fazer com que os mesmos nós sempre sejam escolhidos, enquanto outros podem permanecer ociosos por toda a simulação. O critério de escolha

pode ser randômico ou baseado na ociosidade dos nós, onde os agentes tenderão a visitar os nós com maior ociosidade.

Naturalmente, dois aspectos influenciam diretamente esta tomada de decisão. O primeiro é o campo de visão do agente, que pode ser local ou global, nesse trabalho pioneiro [1] o campo de visão local permite apenas que os agentes tomem conhecimento da ociosidade dos nós vizinhos (separados apenas por uma aresta) ao que se encontra, enquanto o campo de visão global, permite que os agentes tenham conhecimento da ociosidade de todos os nós do grafo.

O segundo aspecto, que se aplica apenas para escolhas baseadas em heurística, é o fato do agente poder ou não saber o que os outros agentes têm feito. De acordo com as possibilidades de comunicação, por exemplo, um agente ter a capacidade de checar se o nó para o qual se dirige foi visitado. As várias possibilidades de abordagens multiagente, nesse trabalho pioneiro, se deram pelas combinações destes quatro parâmetros.

2.3.2 Abordagem Baseada em Utilidade

A função de utilidade de um agente, usada quando existem múltiplos objetivos possivelmente conflitantes, mapeia um estado (ou seqüência de estados) em um número, que descreverá o grau de contentamento deste agente com o estado, ou seqüência de estados, analisado [19]. Quanto maior a utilidade melhor. Então, a tarefa do agente baseado em utilidade é escolher o estado que maximize a utilidade.

As heurísticas utilizadas pelos agentes pioneiros eram bastante simples: apenas a ociosidade do nó objetivo era utilizada, e o algoritmo de escolha do menor caminho para atingir o objetivo considerava que as arestas tinham o mesmo comprimento, não levando em conta as ociosidades dos nós que fazem parte do próprio caminho percorrido até se atingir o nó escolhido.

Sendo assim, os trabalhos [8, 12] aprimoraram os agentes pioneiros com heurísticas mais sofisticadas para definir qual nó será visitado, e para escolher o melhor caminho até este nó. A heurística definida foi:

- A escolha do nó objetivo, que era feita baseada apenas na ociosidade (o nó mais ocioso era escolhido), passou a utilizar uma função de utilidade que considera também a distância entre o nó em que o agente se encontra e o nó destino;

- A escolha do percurso até o nó escolhido, que considerava apenas distância (logo optava pelo menor caminho), agora considera uma função de utilidade que procura um caminho cujos nós por onde o agente passa sejam os mais ociosos, em compromisso com a minimização do comprimento total do caminho (que reflete no tempo para atingir o objetivo).

Dentre os agentes baseados em utilidade, destaca-se o *Heuristic Pathfinder Cognitive Coordinated* (HPCC), uma evolução do *Cognitive Coordinated*, tendo obtido o melhor desempenho dentre os agentes propostos por [8, 12]. Este agente exige uma coordenação central, que determina o próximo nó que cada um dos agentes patrulheiros deve visitar.

2.3.3 Abordagem Baseada em Aprendizagem por Reforço

Em [13], foram desenvolvidos agentes adaptativos que aprendem a patrulhar usando técnicas de Aprendizagem por Reforço (AR) [19]. Aprendizagem por Reforço é freqüentemente descrita como o problema de “aprender o que fazer (como mapear situações em ações) para maximizar uma recompensa numérica” [13]. Em termos gerais, tem-se um agente que toma decisões baseadas em uma determinada representação do ambiente (estado), e recebe um feedback sobre o resultado das suas ações (recompensa).

Primeiramente, o uso da maioria dos algoritmos de AR envolve a modelagem do problema como um Processo de Decisão de Markov (MDP) [31]. Considere um agente que toma decisões seqüencialmente em um ambiente. A cada passo, este agente escolhe uma **ação** de um conjunto finito A , baseado em um sinal de **estado** do ambiente. Este estado pertence a um conjunto finito S , e incorpora as percepções atuais e passadas (eventualmente) de tal maneira que toda a informação relevante para a tomada da decisão está contida nele.

A dinâmica deste processo é descrita por dois componentes: a **distribuição de probabilidade da transição entre estados**, P , e a **função da recompensa instantânea esperada**, R . Estas duas funções são definidas no domínio das triplas $\langle s, a, s' \rangle$, onde s é o estado atual, a é a ação executada pelo agente e s' é o próximo estado no qual o agente irá estar após executar esta ação. Em outras palavras, a distribuição P é uma função que dá a probabilidade de se atingir o estado s' dado que o agente estava no estado s e

executou a ação a , enquanto que a função R diz a recompensa esperada nesta mesma situação [13].

Em resumo, um **Processo de Decisão de Markov finito** é definido por uma tupla $\langle S, A, P, R \rangle$ dos elementos definidos anteriormente. Neste formalismo, o agente age de acordo com alguma **política de ações** $\Pi(s, a)$, que representa a probabilidade de escolha da ação $a \in A$ quando o agente estiver no estado $s \in S$ [13]. O objetivo de um agente em um MDP é maximizar um critério de desempenho a longo prazo, chamado de **retorno**, que é geralmente definido como a soma das recompensas futuras descontadas [13]. Mais formalmente, o agente tenta aprender, dentre o espaço de todas as políticas de ações possíveis, uma **política de ações ótima** Π^* , que maximiza o retorno esperado, também chamado de **função de valor dos estados**. De maneira similar, define-se a **função de valor das ações**, $Q_\Pi(s, a)$ [19].

Q-Learning é um algoritmo tradicional de aprendizagem por reforço para resolução de MDPs. Uma das principais vantagens deste algoritmo, além da sua simplicidade, é o fato de ele não precisar de um modelo do ambiente (*model-free*) [13]. Sendo assim, é possível utilizá-lo em um ambiente onde não se conhece *a priori* a distribuição de probabilidade da transição de estados P nem a função de recompensas esperadas instantâneas R . Como o próprio nome indica, a idéia é computar a função valor das ações, $Q^*(s, a)$, ótima. É provado que a aprendizagem de tal função pode ser feita iterativamente, computando o valor de cada par estado-ação, *on-line*.

Como a teoria do MDP trata apenas do caso de um único agente, o caso multiagente necessário para a realização da tarefa de patrulha foi tratado como um conjunto de agentes cooperativos que aprendem a coordenar suas ações através de AR [13].

Neste contexto, o trabalho descrito em [13] investigou o desenvolvimento de uma inteligência coletiva onde agentes otimizam suas funções de utilidade privadas, que são alinhadas à utilidade global, segundo o conceito de *Wonderful Life Utility (WLU)* [13], de forma que a utilidade de cada agente assegura que os agentes não trabalhem contra o grupo. Do ponto de vista da comunicação, foram consideradas duas abordagens: sistema **Black-Box**, no qual os agentes não conhecem as ações dos outros, apenas as percebem por rastros no ambiente; e sistema **Gray-Box**, no qual os agentes podem comunicar seus atos e futuras ações, essa comunicação foi implementada através de *flags*.

2.3.4 Estratégia do Ciclo Único

Em [9], considera-se que os agentes são capazes de descobrir o caminho ótimo no grafo a ser patrulhado, e a tarefa de patrulhamento consiste em seguir este ciclo repetidamente. Tal abordagem é chamada de estratégia do ciclo único.

Considerando um único agente patrulhando uma área, a estratégia mais simples a ser executada é encontrar um ciclo que percorra toda a área, e fazer o agente andar por este ciclo. Neste caso, a ociosidade máxima será a distância a ser percorrida. Ou seja, no caso de um único agente, a estratégia ótima em termos do critério da *Pior ociosidade* (ver seção 2.3) é a estratégia de ciclo único cujo ciclo é calculado como a solução ótima para o Problema do Caixeiro Viajante (*Travelling Salesman Problem - TSP*) [9, 32], problema clássico já bastante estudado na área de pesquisa operacional.

Uma maneira de estender tal abordagem para o caso multiagente é manter todos os agentes seguindo este mesmo ciclo, separados por uma distância similar e constante entre eles (ver figura 2.2).

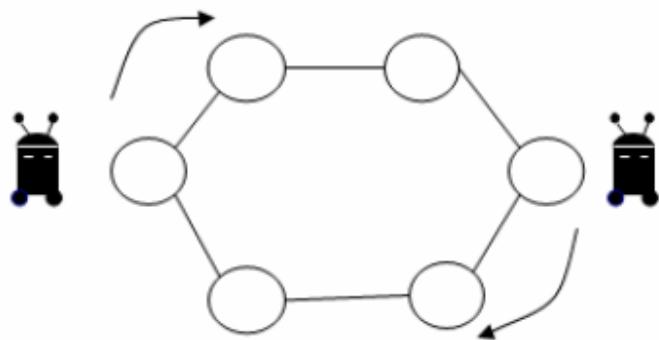


Figura 2-2– Execução da estratégia de ciclo único por dois agentes. Os agentes percorrem o mesmo caminho, separados por um intervalo de tempo

Esta abordagem é totalmente centralizada e utiliza um esquema rígido e explícito de coordenação: os agentes seguem um mesmo caminho e mantêm a mesma distância entre eles. No entanto, como este esquema é centralizado, predefinido e fixo, este tipo de arquitetura terá problemas em certas situações, tais como em ambientes dinâmicos (e.g.

robôs precisando recarregar baterias); em grafos grandes (devido à complexidade das soluções aproximadas do TSP); e em tarefas de patrulha onde as regiões tenham diferentes prioridades.

2.3.5 Abordagem Baseada na Teoria dos Campos Potenciais

Em [10], foi estudado o problema da patrulha em terrenos do mundo real, utilizando robôs. Para isto, foram utilizadas técnicas de navegação de robôs baseadas na teoria dos campos potenciais, ou propagação de dados [33].

O algoritmo desenvolvido, chamado de estratégia CLinG, se baseia na propagação das informações globais sobre o estado do mundo (ociosidade dos nós, posição dos outros agentes, etc), de maneira que os agentes possam agir baseados apenas na informação do terreno (local), mas que ao mesmo tempo, esta mesma informação, resume o estado de todo o mapa (global).

Essa propagação é feita através de *hardware*, com o espalhamento de dispositivos de transmissão de dados (*relays*). Esses dispositivos propagam informações entre si e para os robôs via comunicação sem fio. Os robôs possuem localização e podem estimar sua autonomia de bateria, e mandam essas informações para o mundo de informações propagadas (PDW), que existe em um computador remoto e computa a força tarefa (TS) de cada região [10]. Os robôs patrulham de acordo com os dados propagados, escalam o gradiente de força tarefa e visitam o primeiro vértice de gradiente encontrado.

Os resultados obtidos utilizando-se robôs não são comparáveis aos resultados obtidos por software. Agentes virtuais se movem pelo grafo muito rapidamente e sem que ocorra nenhuma interferência física, e interferências são problemas críticos conhecidos em robótica e concentrações de robôs devem ser evitadas.

Outro ponto a ser considerado é que agentes virtuais não precisam recarregar bateria, o que traz consequências sérias na eficiência da patrulha, com robôs. Além disso, robôs coletam informações durante o seu tempo ou duração de visita. Por exemplo, se o robô estiver equipado com uma câmera, ele precisará parar e fazer uma panorâmica do local. Além disso, o robô pode estar programado para fazer uma tarefa secundária, como coletar o lixo encontrado, o que trará um maior *delay*.

2.4 AVALIAÇÃO DAS ABORDAGENS SEGUNDO OS NOVOS CRITÉRIOS

Neste trabalho, além dos critérios propostos na seção 2.2.2 (*adaptabilidade, escalabilidade e estabilidade*), as abordagens de sistemas multiagente para patrulhamento também serão avaliadas segundo o critério de *ociosidade média* descrito na seção 2.2.1.

As abordagens anteriores, com exceção dos agentes simples do trabalho pioneiro, precisam de pré-processamento [34]. Portanto, não possuem uma boa *adaptabilidade offline*, nem *online*. Os agentes aprendizes precisam ser treinados para cada mapa, bem como os agentes baseados na estratégia de ciclo-único que também precisam computar o melhor ciclo a ser percorrido para cada mapa, o que significa que, além de não serem adaptáveis *offline*, também não são preparados para lidarem com ambientes dinâmicos, ou seja, não são adaptáveis *online*. A abordagem baseada em utilidade também precisa de grande parte do tempo da simulação para se estabilizar.

Comparações iniciais entre essas abordagens foram feitas em [8, 34]. Quanto à estabilidade, os agentes baseados em utilidade, os baseados em ciclo-único e os aprendizes se mostraram mais estáveis que os agentes pioneiros, e com números muito próximos.

O critério de escalabilidade não foi testado em trabalhos anteriores, mas acredita-se que os agentes aprendizes e os agentes baseados na estratégia de ciclo-único gastarão muito tempo durante suas fases de pré-processamento, devido ao enorme número de simulações necessárias na fase de aprendizagem dos agentes aprendizes e à complexidade computacional do cálculo do TSP.

A comparação feita em [8, 34] mostrou que os agentes baseados na estratégia de ciclo-único possuem o melhor desempenho em termos de ociosidade média. Os agentes pioneiros, descritos na seção 2.3.1, mostraram os piores desempenhos em termos de ociosidade média.

Os agentes baseados na teoria de campos potenciais não foram comparados com nenhuma outra abordagem, pois os resultados obtidos utilizando-se robôs não são comparáveis aos resultados obtidos por software.

2.5 CONCLUSÃO

Neste capítulo, foi apresentada uma visão geral do estado da arte em pesquisas relacionadas à patrulha, citando trabalhos relacionados ao tema e pesquisas pioneiras de agentes patrulheiros. Foi também apresentada uma discussão inicial sobre o desempenho desses agentes quanto aos novos critérios de avaliação propostos.

Este trabalho se propõe a desenvolver novas alternativas de agentes para a patrulha, baseados na abordagem negociadora. Tal abordagem não necessitaria de ajustes para cada situação, como no caso dos agentes aprendizes, que precisam ser treinados para cada mapa, bem como os agentes baseados na estratégia de ciclo-único que também precisam ser ajustados para cada mapa (ajustes estes que, em ambos os casos, podem ser proibitivos em grafos grandes). Além disso, esta abordagem não requereria os esquemas centralizados e explícitos de coordenação, que criam gargalos e eram críticos ao desempenho do HPCC, por exemplo.

Seguindo a metodologia proposta em [1, 6], este trabalho propõe a utilização dos critérios de avaliação por ele propostos, bem como os novos critérios, estudar o desempenho de agentes capazes de se comunicar diretamente a fim de negociar nós para patrulhar.

No próximo capítulo, serão explicados os principais conceitos de negociação em sistemas multiagentes, a fim de possibilitar o entendimento da abordagem implementada neste trabalho.

Capítulo 3

Negociação

Neste capítulo, serão apresentados os principais conceitos de negociação em sistemas multiagentes, com destaque à teoria dos leilões, para possibilitar o entendimento da abordagem implementada neste trabalho.

3.1 NEGOCIAÇÃO

A patrulha é uma tarefa realizada por agentes autônomos, o que implica que um agente não pode forçar outro agente a realizar um serviço, ou modificar seu estado interno. Ao invés disso, um agente deve tentar convencer o outro a cooperar com ele. Para tanto, eles precisam negociar.

Negociação é o processo que permite que grupos de agentes se comuniquem entre si para tentar estabelecer um acordo, mutuamente aceitável, para a solução de um determinado problema. Em outras palavras, é um processo executado na intenção de possibilitar que os agentes colaboradores atinjam um consenso na distribuição de tarefas e recursos [11]. Os agentes primeiro comunicam seus posicionamentos, que podem ser conflitantes e, então, tentam atingir um acordo, de valor ou ponto de vista, fazendo concessões ou buscando alternativas [35]. A capacidade de alcançar acordos, sem um intermediário, é uma habilidade fundamental de agentes inteligentes autônomos [14].

Negociação é a forma chave de interação em sistemas compostos de múltiplos agentes autônomos, capazes de decidir que ações devem executar e sob que condições [36]. Estes agentes podem precisar interagir para alcançar seus objetivos, ou porque não possuem capacidade ou recursos suficientes para solucionar seu problema, ou porque há interdependências entre os agentes [37]. Por exemplo, um pintor só pode pintar uma parede se estiver de posse das ferramentas e tinta necessárias e se o pedreiro a tenha terminado de construir.

A negociação possui quatro componentes principais descritos com detalhes a seguir:

- **O protocolo:** define as “regras de encontro” entre agentes [37], ou seja, explicita que propostas os agentes podem fazer. Também define os estados da negociação (por exemplo, aceitando ofertas ou negociação encerrada), além dos eventos que causam mudanças nestes estados (e.g., não haver mais compradores ou lance aceito).
- **O objeto a ser acordado:** é o assunto discutido na negociação, que pode ser um preço, qualidade, data de entrega, ou no caso da patrulha um nó a ser visitado.
- **A estratégia do agente:** define a forma de gerar propostas que um agente aplicará para conseguir, dado um protocolo, atingir o seu objetivo na

negociação. O agente, sendo racional, deve ter como objetivo maximizar o resultado esperado de sua função de utilidade [19]. A função de utilidade descreve o grau de contentamento de um agente com um estado do mundo [19].

- **Uma regra:** que determina quando o acordo foi atingido.

Uma fonte de complexidade em negociações é o número de agentes envolvidos no processo e forma como estes interagem. A negociação pode ser bilateral ou entre múltiplos atores.

A negociação bilateral [38] ocorre somente entre dois oponentes que atingem um acordo através da troca de propostas. Como exemplo, podemos citar a negociação de um cliente com um vendedor por um desconto.

As negociações multilaterais são tipicamente processos de leilão. Neste tipo de negociação pode haver um número ilimitado de participantes envolvidos na negociação, ou seja, mais de um participante com o papel de vendedor ou de comprador. Os participantes com o mesmo papel concorrem entre si. As negociações multilaterais podem ser de dois tipos: negociação de um-para-muitos e negociação muitos-para-muitos.

Na negociação de um-para-muitos, um único agente negocia com vários outros. Como exemplo, podemos citar os leilões de obras de arte em galerias, onde temos um vendedor e vários compradores.

Na negociação muitos-para-muitos vários agentes negociam com vários outros simultaneamente. Como exemplo, podemos citar uma feira livre, onde os feirantes tentam vender para vários clientes ao mesmo tempo.

Leilões são muito populares por serem cenários de interação bastante simples, o que os faz uma boa primeira escolha como forma para os agentes chegarem a um acordo. Agentes autônomos podem fazer uso de leilões para alocar tarefas e recursos [14], entre outros.

3.2 LEILÕES

Leilões representam a classe de mecanismos básicos de sistemas econômicos que computam o resultado de interações sociais. O leilão aceita mensagens em forma de

lances que expressam o desejo de aquisição do bem em leilão por um certo valor. Esta definição inclui uma grande variedade de transações como leilões de arte, carros usados, ações, entre outros, além do processo competitivo utilizado por companhias e pelo governo para aquisição de bens e serviços.

Essencialmente, leilões são usados por três razões: revelação da avaliação dos compradores, rapidez de venda e prevenir acordos desonestos entre vendedor e comprador [39].

Suponha que uma pessoa seja proprietária de um objeto único e valioso – por exemplo, uma pintura de Picasso e, por alguma razão, resolve vendê-lo. Esta pessoa tem uma idéia clara do seu preço de reserva – preço mínimo que o proprietário aceitaria receber pelo bem – mas, obviamente, pretende conseguir mais que este valor [4]. Durante o leilão o valor que cada indivíduo está disposto a trocar pelo bem em questão é revelado a medida em que lances são feitos.

Quando a venda do bem precisa acontecer em curto prazo, pode-se fazer uso de leilões, já que estes acontecem com data certa de início e fim. Leilões de obras de arte como os que ocorrem na *Sotheby's* [40] e na *Christie's* [41], em geral são divulgados durante alguns dias por todo o mundo e duram apenas algumas horas.

Governos e grandes empresas fazem uso de leilão tanto para economizar na aquisição de grandes quantidades de bens e serviços, como para prevenir acordos desonestos entre funcionários e lojas ou prestadoras de serviços. Como leilões são divulgados é criada concorrência, o que tende a diminuir os preços, e a compra só é realizada caso haja mais de uma empresa tentando vender o bem. A empresa que fornecer o serviço ou produto pelo menor custo, ganha o leilão. Desta forma, evita-se que haja protecionismo, favorecimento e corrupção, além de estimular a diminuição de custos e a concorrência.

A mais antiga referência a leilão é encontrada em Herodotos, na compra de homens e esposas na Babilônia em 500 a.C. Estes leilões eram únicos, pois chegavam a começar com preços negativos [39].

Na Roma antiga, leilões eram usados em várias negociações comerciais, como, por exemplo, em vendas de propriedades e de bens pilhados em guerras. O leilão mais

notável aconteceu em 193 d.C. quando a guarda pretoriana leiloou todo o império após matar o imperador [39].

Na Internet, os leilões já existiam antes da popularização da *World Wide Web*. Antes do surgimento do primeiro *web browser*, já existiam leilões em *newsgroups*.

Os primeiros leilões na Web foram o Onsale [42], que começou a funcionar em maio de 1995 e o eBay [43] em setembro do mesmo ano. Atualmente, existem mais de 500 leilões on-line, entre os quais se destacam, além dos já citados, Yahoo![44] e Arremate [45].

Os leilões em *e-Commerce* trouxeram vantagens temporais e geográficas, reduzindo o custo da aquisição antes, durante e após a transação [46]. Antes da transação, diminui o custo da busca de produtos, comparação entre os produtos e preços, através de buscas simples na *Web*, pode-se encontrar vários modelos do item desejado em vários *sites*. Nos *sites* de leilão, a organização por categorias também facilita a busca. Durante a aquisição, os prováveis compradores não precisam mais se deslocar para participar de leilões; estes podem durar de horas a semanas; e o comprador pode dar seus lances no horário e local que achar mais conveniente. O custo de comunicação entre as partes é reduzido, bem como a necessidade de espaços físicos para encontros e processamento de documentos em papéis. Após a aquisição, o comércio eletrônico permite às companhias diminuir o custo da comunicação para monitorar a desempenho do contrato ou confirmar entregas. Adicionalmente, os participantes podem guardar registros das transações, e podem divulgar sua satisfação com o negócio [46].

Consumidores procuram comprar bens que satisfaçam suas necessidades a um bom preço. Quando o número de vendedores é pequeno, os compradores podem fazer uma análise e selecionar o melhor negócio. Mas, à medida que o número de possibilidades aumenta, procurar e comparar ofertas se torna um processo caótico [47].

A Internet trouxe a possibilidade de comércio eletrônico, mas associado a ele o problema de gerenciar a enorme quantidade de ofertas e demanda. Para auxiliar os compradores em potencial, antes da transação, a executar tarefas como: pesquisa do melhor produto para suas necessidades e identificação dos *sites* onde este produto está com melhor preço, começaram a surgir ferramentas, como o AuctionPatrol [48] e o McFind [49].

Também para auxiliarem seus compradores, também denominados arrematantes, durante a aquisição do produto, os *sites* de leilão criaram *proxies* [43, 45], que são ferramenta que auxiliam o comprador assumindo seu lugar durante o leilão. O comprador informa o valor máximo que está disposto a pagar pelo bem, e o *proxy* aumenta o lance em pequenos incrementos até que esse valor seja atingido ou até que vença o leilão, sendo opcional manter o comprador informado através de *emails*. *Sites* como eBay [43] e o Arremate [45] disponibilizam *proxies* para os seus arrematantes.

Ainda para auxiliar arrematantes na tarefa de comprar em leilões e também para permitir o estudo e a simulação deste mecanismo de compras, começaram a surgir pesquisas sobre o uso de agentes de software para automatizar tarefas inerentes à participação em um leilão. Como exemplo destes sistemas de agentes temos o *AuctionBot* [50], *AgILE* [47] e o *eMediator* [51].

AuctionBot [50] é um servidor de leilões *online*, versátil, que suporta agentes de *software* e humanos desenvolvido na universidade de Michigan. Este servidor é gratuito e implementa uma grande variedade de leilões, o que o torna uma boa escolha para quem deseja desenvolver ou prototipar *online marketplaces* [50].

AgILE [47] é um sistema desenvolvido no Brasil, onde os agentes participam de leilões comerciais, na *Internet*, ou em seu servidor [47]. Este servidor implementa apenas dois tipos de leilão, o inglês e o holandês, que serão descritos na seção 3.2.2. Os agentes podem participar de leilões simultâneos, mas apenas dos dois tipos já descritos. Não há *interface* para o desenvolvimento de agentes por terceiros.

O *eMediator* [51] é um servidor de comércio eletrônico composto de um *software* de apoio configuração e execução de leilões entre pessoas e/ou agentes. Contempla todos os tipos de leilão e ainda agentes móveis capazes de configurar leilões, mover-se para outros leilões, dar lances entre outras habilidades. Os seus agentes não são capazes de gerenciar leilões simultâneos [51]. Além disso, este servidor fornece suporte a agentes desenvolvidos por terceiros. Este servidor está disponível para uso em [52].

Há também desvantagens, como o fato de o item leiloado estar distante do comprador este não poder verificar sua integridade. Este problema pode ser minimizado com a divulgação de fotos do item. Mas, não há garantia de que a foto é realmente do item que será enviado, como também o leiloeiro pode ser desonesto e não enviar o item,

ou enviar um objeto diferente do leiloado. Para diminuir as chances de pessoas desonestas se aproveitarem da oportunidade para repassar produtos danificados, ou deixarem de fazer entregas, muitos *sites* implementam um sistema de avaliação de compradores e vendedores (leiloeiros). Assim, antes de comprar um produto, o arrematante pode checar o *status* do vendedor e ver se é confiável. Bem como o vendedor pode checar o comprador. Nesses sistemas de leilão, dar o lance vencedor e não comprar o produto por este valor é um erro grave, podendo levar o comprador a ser expulso da comunidade.

Após a transação ser finalizada, e o produto adquirido ser entregue ao vencedor do leilão, os participantes do leilão – compradores e vendedores – podem acessar o *site* e elogiar ou criticar uns aos outros. Desta forma, bons vendedores, aqueles que entregam exatamente o produto adquirido e no prazo e estado de conservação divulgado, tem uma melhor reputação que os desonestos, o que ajuda os novos compradores a identificar bons negociantes. Da mesma forma, arrematantes que não desistem da aquisição após vencerem a disputa também tem melhor reputação, pois os vendedores também podem classificá-los [45].

Muitos problemas inerentes à utilização de sistemas multiagentes envolvem alocação de recursos como, por exemplo, acesso a dispositivos ou a agentes especialistas e estes problemas podem ser resolvidos com ajuda da Economia [53].

Assim, a Teoria dos Leilões é um ramo importante da teoria da Economia por razões práticas, empíricas e teóricas. Práticas porque um enorme volume de transações econômicas é conduzido através de leilões. Por exemplo, estes são largamente utilizados em muitas transações como: privatização de estatais, venda de concessões de telefonia, direito de exploração de petróleo, contratos governamentais ou de empresas de grande porte, imóveis, gado, produtos agrícolas, carros, peças de arte, antiguidades, venda de *commodities* em bolsa de valores, entre outros. Empíricas porque leilões são ambientes econômicos simples e bem definidos, de fácil implementação que provêm ambientes de teste muito valiosos para a teoria econômica, especialmente para teoria dos jogos com informação incompleta. Finalmente, teóricas porque teoria dos leilões permite o entendimento de outros métodos de formação de preços e negociações nos quais compradores e vendedores são envolvidos ativamente na determinação do preço, como

por exemplo, negociações em lojas de varejo [51]. Há uma analogia muito próxima entre a teoria de leilões e a teoria do monopólio, e a teoria de leilões pode também ajudar a desenvolver um modelo de oligopólio para determinação de preços [51].

3.2.1 Características dos Leilões

Para um melhor entendimento na Teoria de Leilões, é importante o conhecimento de um pequeno grupo de conceitos e características básicas que permitem alterar tanto o protocolo do leilão, quanto às estratégias a serem seguidas pelos agentes. Nesta seção, serão apresentados estas características e termos comuns utilizados na área. Essa revisão de características foi baseada em [14].

Participantes

Um leilão toma forma entre um agente leiloeiro e uma coleção de agentes licitantes (arrematantes). O objetivo do leiloeiro é maximizar o valor da mercadoria alocada e os licitantes minimizar este valor. O leiloeiro tentará atingir seu objetivo seguindo o projeto de um mecanismo de leilão apropriado – um conjunto de “regras de encontro” – enquanto os licitantes tentarão alcançar seus objetivos usando uma estratégia de acordo com as regras de encontro, mas que também proporcionem um resultado ótimo para si, ou seja, aquisição do produto por um valor o mais baixo possível [14].

Então, são de dois tipos as entidades que participam de leilões:

- **Arrematante:** entidade que participa do leilão com o objetivo de adquirir o bem que será leiloado.
- **Leiloeiro:** entidade responsável pelo andamento do leilão. Pode ser um terceiro que não está comprando ou vendendo produtos, mas se coloca a serviço do proprietário do bem ou serviço a ser leiloado e dos possíveis compradores, para facilitar o leilão. Diz-se que o leilão é clássico quando o leiloeiro oferece um bem por um valor inicial e aguarda os lances dos arrematantes, neste caso, o leiloeiro está a serviço do que deseja vender o bem. Já no leilão reverso, o leiloeiro está a serviço do arrematante, divulga que precisa adquirir o bem e aguarda os lances dos vendedores.

Conceitos

O **valor de reserva** é o máximo valor que um arrematante está disposto a pagar pelo produto ou preço mínimo que o proprietário (leiloeiro) aceitaria receber pelo bem. Como o objetivo do arrematante é conseguir o bem pelo menor preço possível, este valor não deve ser revelado.

Um outro conceito interessante é o de **estratégia dominante**. Uma estratégia é dita dominante se independente do comportamento dos outros participantes é a melhor estratégia possível.

Ainda, há o conceito da **maldição do vencedor** que ocorre quando arrematantes, avessos ao risco de não adquirirem o item, acabam pagando um valor muito mais alto do que o valor real do item ou do seu preço de reserva inicial.

Determinação do Valor

O objeto a ser acordado pode ter valor público ou privado: dizer que o objeto possui valor público significa dizer que todos os agentes atribuem o mesmo valor a este produto. Enquanto que, ao assumir que o objeto tem valor privado, é o mesmo que dizer que, levando-se em consideração a posição de cada agente e seus objetivos, o objeto pode assumir um valor diferente para cada agente.

Determinação do Vencedor e do Preço

Esta característica determina quem ganha o leilão e quanto vai pagar pelo prêmio. As possibilidades são:

- *Leilão de primeiro preço (first-price)*: neste tipo de leilão, o agente que ofereceu o maior valor, fica com o objeto e paga o preço que ofereceu;
- *Leilão de segundo preço (second-price)*: neste tipo de leilão, o agente que ofereceu o maior valor, fica com o objeto e paga o segundo maior valor oferecido. Na prática, só funciona com leilão *sealed-bid*.

Divulgação dos Lances

No protocolo de leilão, outra variante determina se os agentes podem ou não conhecer os lances dos outros agentes. Quanto a esta variante, as possibilidades de leilão são:

- *Open-cry* (oral): neste caso, cada licitante sabe todos os valores oferecidos pelos outros;
- *Sealed-bid* (fechado ou escrito): aos licitantes não é permitido conhecer os lances feitos pelos outros;

Execução

O mecanismo pelo qual o leilão procede pode ser de três formas:

- *One-Shot*: neste caso só é permitido a cada participante dar um lance, ou seja, o leilão é encerrado após o primeiro turno;
- *Ascendente*: os lances começam baixos e vão aumentando, em vários turnos;
- *Descendente*: os lances começam altos e vão aumentando, em vários turnos.

Encerramento

Os leilões podem ser agrupados quanto à forma de encerramento da seguinte forma:

- *Temporal*: o leilão é encerrado em momento pré-determinado pelo leiloeiro. Este momento é conhecido por todos os arrematantes;
- *Por evento*: o leilão é encerrado quando o evento definido pelo leiloeiro ocorre. Os eventos mais comumente utilizados são: apresentação de um lance ou fim de um determinado período sem um lance.

3.2.2 Tipos mais Conhecidos de Leilões

Os tipos de leilões mais conhecidos são o Inglês [14, 54], o Holandês [14, 54], o Americano (mais conhecido com *first price sealed bid*) [14] e o Vickrey [14]. Há ainda o leilão duplo e o multi-item [54]. A seguir, eles são descritos em termos de suas características, se há estratégias dominantes e como poderia gerar comportamento anti-social.

Inglês

Também chamado de aberto, oral, ou leilão de lances ascendentes. Neste leilão, os arrematantes dão lances crescentes para um determinado item, até que nenhum arrematante esteja disposto a dar um lance maior que o atual, que será considerado vencedor, e será o preço a ser pago.

É o tipo mais conhecido, e comumente usado para vender obras de arte ou antiguidades. Opcionalmente, o leiloeiro pode determinar um valor inicial, ou valor de reserva, que é o mínimo valor aceito por aquele item. É um leilão *open-cry, first price*, ascendente. A estratégia ideal a ser seguida é aumentar o lance em pequenos incrementos, e depois se retirar (caso o valor de reserva do comprador seja atingido). Este tipo leilão é sujeito à maldição do vencedor.

Holandês

Funciona de forma inversa ao leilão inglês. Neste leilão, o leiloeiro informa o preço fixado como inicial, em geral exorbitante, e diminui este preço progressivamente, até que um arrematante esteja disposto a pagar o preço corrente. Caso mais de um arrematante esteja disposto a pagar o valor corrente, o valor é incrementado progressivamente, até que reste apenas um arrematante ainda disposto a pagar o preço agora mais alto. Tem este nome por ser utilizado por vendedores de flores (Tulipas) na Holanda. Também é utilizado para leiloar peixes em Israel [55].

É um leilão *open-cry, first price*, descendente. Não há estratégia dominante, pois é impossível prever qual valor os outros participantes estão dispostos a pagar e também é suscetível a maldição do vencedor, já que por aversão ao risco de perder o objeto em leilão, um arrematante pode aceitar pagar um preço ainda muito alto.

Americano

Os arrematantes entregam envelopes lacrados com seu respectivo lance. Cada arrematante só pode dar um lance. Os envelopes são abertos juntos e o que tiver o maior lance é o vencedor, cabendo ao arrematante pagar o valor do lance pelo bem.

Leilões deste tipo foram utilizados em privatizações de algumas estatais brasileiras.

É um leilão *sealed-bid, first price, one-shot*. Como o arrematante não tem como saber os lances dos outros participantes e o valor a ser pago é o valor oferecido, não há estratégia dominante.

Vickrey

Similar ao Americano, a diferença é que o vencedor paga o valor do segundo maior lance do leilão. É um leilão *sealed-bid, second price, one-shot*. A estratégia dominante é oferecer o valor real da mercadoria

Pode gerar comportamento anti-social, pois caso o arrematante saiba dos prováveis lances dos concorrentes e seu valor de reserva não seja competitivo, ele pode oferecer um valor acima do seu valor de reserva apenas para que o vencedor do leilão pague mais pelo bem.

Leilão Duplo

No caso de leilões duplos, onde a iniciativa dos lances é de ambos, comprador e vendedor, como nas bolsas de valores, é necessário um mecanismo diferente para a determinação de preços. Já que há lances de compra e venda, o preço pode ser definido de três formas:

Valor médio: é a média entre um lance de compra e um de venda;

Valor do lance de compra: o preço final é o preço do lance de compra;

Valor do lance de venda: o preço final é o preço do lance de venda.

No caso do preço estabelecido ser o do lance de compra, o negócio só é realizado quando existe um lance de venda com valor menor ou igual ao de compra.

O esperado dos leilões duplos é que se tornem vários leilões ingleses simultâneos, com a diferença de que estes vendedores tenderão a baixar os preços, dado a concorrência [54].

Leilão Multi-item

Em muitos casos o leilão pode incluir vários itens (idênticos ou não). Nestas ocasiões, o valor que o arrematante irá oferecer por um destes itens irá depender de quais outros itens ele poderá obter [56]. É preferível que se permita que os lances sejam dados em *combinações de itens*. Por esta razão, estes leilões são chamados de *leilões combinatórios*.

Por exemplo, se uma fazenda é posta em leilão, um arrematante pode desejar dois bois e trinta vacas, e pode não estar interessado em comprar apenas os bois. O leilão combinatório permite que os arrematantes expressem suas preferências e que melhores

alocações de mercadorias ou recursos sejam feitas. Porém, os lances vencedores e os sub-lotes só serão definidos ao final para que todos os lances ofertados sejam avaliados de forma a garantir a alocação que traga mais benefício ao leiloeiro.

Leilões combinatórios podem ser usados no mercado de eletricidade, banda de rede, entre outros [51].

3.2.3 Leilões em SMA

Negociações têm sido usadas para resolver conflitos em uma grande variedade de domínios de sistemas multiagentes. Essas negociações podem envolver o uso de recursos compartilhados, alocações de tarefas e conflitos entre vendedores e compradores [57].

Em muitos domínios acordos devem ser alcançados pelos agentes a respeito da distribuição de um conjunto de itens. Por exemplo, no ambiente de informação armazenada, os agentes necessitam decidir-se sobre a alocação de informações, isto é, os artigos sob a consideração são conjuntos de dados. Quando resolver supõe programar o uso de um recurso, um acordo deve ser alcançado determinando-se o tempo e horário de uso a ser atribuído a cada agente. Quando os agentes necessitam se decidir em atribuições de uma tarefa, então os artigos são as tarefas e uma decisão deve ser feita de qual agente realizará uma tarefa dada. A maioria destes conflitos pode ser resolvida eficientemente fornecendo aos agentes um sistema monetário, onde os agentes são modelados como compradores e vendedores, e resolvendo os conflitos usando transferência do dinheiro. Por exemplo, um servidor pode "vender" uma série de dados a um outro servidor ao realocá-los; um sub-contratado pode ser pago a fim realizar uma tarefa [54]. Podemos citar os sistemas M+ [58] e Murdoch [59] como exemplos de sistemas multiagentes que fazem uso de leilões para alocar tarefas. Ambos os sistemas são baseados no protocolo de *Contract Net* [14], colocam tarefas disponíveis em leilão e robôs interessados fazem ofertas que são as suas capacidades de realizar a tarefa. O robô melhor preparado para executar a tarefa ganha o leilão e a executa.

Como visto, há muitos padrões de iteração em leilões. Dado o padrão de interação, determina-se o tipo de protocolo que será utilizado. Dado o protocolo, os agentes necessitam decidir sua estratégia. O Vickrey é amplamente utilizado em DAI – *Distributed Artificial Intelligence* – e em pesquisas de comércio eletrônico no caso de

leilões de um-para-muitos, devido ao fato deste protocolo incentivar os arrematantes a darem lances justos.

3.3 CONCLUSÃO

Negociação é o processo que permite que grupos de agentes se comuniquem entre si para tentar estabelecer um acordo, mutuamente aceitável, para a solução de um determinado problema.

Neste capítulo, foram mostrados os conceitos fundamentais do mecanismo de negociação, seus componentes e técnicas, bem como foi apresentado um apanhado geral sobre a Teoria de Leilões e o estado da arte no uso desta teoria.

Os leilões transformaram-se uma área de interesse crescente desde que um volume enorme de transações econômicas passou a ser conduzido com estas vendas públicas. A formação de casas de leilões eletrônicas virtuais na Internet como *eBay* [43] aumentou ainda mais o interesse.

Leilões são muito populares por serem cenários de interação bastante simples, o que os faz uma boa primeira escolha como forma para os agentes chegarem a um acordo. Agentes autônomos podem fazer uso de leilões para alocar tarefas e recursos, entre outros, como é o caso dos sistemas M+ [58] e Murdoch [59].

Leilões possuem várias características básicas que permitem alterar tanto o protocolo do leilão, quanto às estratégias a serem seguidas pelos agentes, entre estas características podemos citar determinação de valor (público ou privado), vencedor (o agente que deu o maior lance ou o segundo maior), preço (o maior lance, ou o segundo maior), forma de execução (uma rodada, ascendente, descendente), encerramento (temporal ou por evento) entre outros.

Por ser um cenário de interação bastante simples, e permitir que acordos sejam atingidos rapidamente, leilão é uma boa escolha como ferramenta de negociação para agentes patrulhadores.

No próximo capítulo, será mostrada a nossa abordagem negociadora de agentes para patrulha.

Capítulo 4

Negociação na Patrulha

Neste capítulo, serão descritos as abordagens implementadas e os principais critérios utilizados para avaliar seu desempenho.

4.1 NEGOCIAÇÃO NA PATRULHA

Neste trabalho, novas alternativas de agentes para a patrulha são apresentadas. Alternativas que seguem uma abordagem negociadora que não necessita de ajustes para cada situação como no caso dos agentes aprendizes ou dos agentes baseados na estratégia de ciclo-único, que precisam ser ajustados para cada mapa, ajustes estes que, em ambos os casos, podem ser proibitivos em grafos com muitos nós.

Além disso, esta abordagem não requer os esquemas centralizados e explícitos de coordenação, que criam gargalos e eram críticos ao desempenho do HPCC (ver seção 2.4), por exemplo.

Como visto, todas as pesquisas citadas no capítulo 2 usaram sistemas multiagentes para patrulhamento onde os agentes não tinham como se comunicar entre si diretamente, a não ser que fossem de hierarquias diferentes (o coordenador enviava mensagens aos coordenados [1, 6, 12]), ou identificando sinais, os *flags* ou marcas deixadas no ambiente pelos agentes ou através de *blackboard*. Na abordagem negociadora, as limitações de comunicação dos trabalhos precedentes foram removidas, permitindo que os agentes troquem mensagens entre si.

Com negociações implementadas por leilões, os agentes particionam o grafo melhor e mais rapidamente, obtendo um melhor desempenho nos critérios de ociosidade média e desvio padrão da ociosidade, descritos no capítulo 2. Por consequência, os sistemas são mais estáveis. Os sistemas multiagentes negociadores para a patrulha são sistemas flexíveis que podem supervisionar grafos com regiões e mudanças na topologia durante simulações.

Neste trabalho, os agentes negociadores serão apresentados seguindo a metodologia de desenvolvimento incremental, e serão simulados em vários cenários, inclusive em mundos maiores, a fim de se confirmar sua capacidade de formarem sistemas de patrulhamento escaláveis, estáveis e adaptáveis. Por fim, também é desejável melhorar os resultados da ociosidade média.

4.2 DESCRIÇÃO DA ABORDAGEM UTILIZADA

Os agentes leiloeiros recebem um conjunto de nós sorteados aleatoriamente que deverão patrulhar. A partir daí, eles descobrirão o caminho que farão para visitar estes nós, caminho este que deverá minimizar a ociosidade dos nós. Ou seja, os agentes buscam o caminho que minimize a quantidade de ciclos necessária para percorrer todos os nós e recomeçar a visitá-los. O custo do caminho é calculado como a soma dos comprimentos das arestas que ligam seus nós.

A preferência de um agente entre diferentes estados do mundo é capturada por uma função de utilidade, a qual atribui um valor real para expressar o desejo por um estado. A função de utilidade descreve o grau de contentamento de um agente com um dado estado do mundo [19]. Como no caso destes agentes patrulhadores, a função de utilidade está diretamente ligada à distância a ser percorrida, ou seja, os agentes sempre tentarão manter um conjunto de nós bastante próximos. Como a distribuição inicial é randômica, os agentes tentarão sempre renegociar estes nós, ou seja, todos os agentes são leiloeiros em potencial.

Para maximizar sua função de utilidade, o agente leiloeiro calculará qual nó está prejudicando mais o seu desempenho, e tentará trocá-lo por outro que maximize sua função de utilidade. Na tentativa de fazer a melhor troca possível, o agente promoverá um leilão, no qual oferecerá o nó que deseja trocar aos seus colegas. Cada companheiro, por sua vez, ao receber a informação de um nó oferecido pelo leiloeiro, calcula o seu custo atual e verifica os resultados de sua função de utilidade caso troque um de seus nós pelo leiloado. Após testar a troca para cada um de seus nós, o companheiro pode propor uma troca (fazer um lance, ou seja, oferecer um nó em troca do leiloado) ou informar que não tem interesse nesse nó. A resposta depende do fato deste encontrar ou não um nó seu que, quando trocado pelo leiloado, maximize o resultado de sua função de utilidade.

Após receber todos os lances, o leiloeiro testará a troca do seu nó que está sendo leiloadado por todos os oferecidos. Caso alguma troca maximize sua função de utilidade, este agente se comunica com agente que propôs a melhor troca, avisando que aceita a proposta (neste caso, ele remove o nó leiloadado da sua base e adiciona o proposto,

enquanto o companheiro também faz a troca), e informa a cada um dos outros agentes que rejeita seus lances.

Esses leilões poderão ser de uma ou mais rodadas. O agente leiloeiro, então, escolherá fazer negócio com o agente que lhe propuser a melhor troca. Assim, cada agente só fará uma proposta caso a troca de algum dos seus nós pelo que está sendo leiloado maximize sua função de utilidade. Vale lembrar que o nó que está sendo leiloado tem valor privado para os agentes, já que significa que para cada agente este nó pode ter um valor menor ou maior, dependendo do impacto que ele tiver na função de utilidade do agente em questão.

Nas seções seguintes, vamos definir a estratégia de negociação adotada (seção 4.2), seguida de uma apresentação de como se dá a comunicação entre os agentes neste trabalho (seção 4.3). A seção 4.4 traz as configurações possíveis de definição de agentes negociadores. E, por fim, a seção 4.5 apresenta as alternativas de agentes da abordagem negociadora.

4.3 NEGOCIAÇÃO

No capítulo 3, foi dito que a negociação possui três componentes principais, sendo eles: o protocolo, o objeto a ser acordado e a estratégia dos agentes. Vejamos agora, como definimos estes componentes neste trabalho.

Para agentes leiloeiros, o protocolo de negociação é formado pelas seguintes ações:

- *Informar que deseja leiloar um nó* – a cada ciclo, o agente escolhe um ou dois nós para serem leiloados;
- *Propor uma troca* – ao receber a informação de que um nó interessante para ele está sendo leiloadado, o agente envia uma mensagem com o seu lance;
- *Recusar-se a participar do leilão* - ao receber a informação de que um nó que não interessa a ele está sendo leiloadado, o agente envia uma mensagem informando seu desinteresse em participar do leilão;
- *Aceitar a proposta* – ao escolher o melhor lance, o agente no papel do leiloeiro envia uma mensagem ao vencedor do leilão informando que aceita a troca;

- *Rejeitar a proposta* - o agente no papel do leiloeiro envia uma mensagem a cada perdedor do leilão informando que seu lance não foi o vencedor.

O objeto a ser acordado é o nó a ser trocado e a estratégia dos agentes é escolher um nó próximo aos que quer manter, com a intenção de particionar dinamicamente o grafo, de forma a ficar com um conjunto de nós os mais próximos possíveis.

Os primeiros agentes leiloeiros desenvolvidos utilizavam leilão de uma rodada [29]. Pois se tratava de um estudo preliminar, a opção mais simples foi implementada, para que uma metodologia incremental fosse seguida, caso os resultados apontassem que esta abordagem poderia alcançar bom desempenho.

Na tentativa de criar mais possibilidades de trocas, neste trabalho foi implementado um agente cujos leilões podem se dar em duas rodadas de lances. Por fim, foi ainda implementado um agente que envia vários lances no mesmo leilão.

4.4 COMUNICAÇÃO

Neste trabalho, os agentes se comunicam diretamente através de atos de fala [14]. Para implementar esses atos de fala foi utilizada uma simplificação do protocolo de comunicação em sistemas multiagentes da FIPA¹-ACL² [60], descrita a seguir. Neste trabalho, apenas as performativas *inform*, *propose*, *refuse*, *accept-proposal* e *reject-proposal* foram utilizadas, por serem as relacionadas com o protocolo de leilões descrito na seção anterior. A estrutura da mensagem foi simplificada: assumimos que todos os agentes conhecem a ontologia e usam a mesma linguagem, já que só trabalhamos com sistema homogêneos. Assim, os atos de fala usados neste trabalho seguem o formato mostrado na Figura 4-1.

(FIPA – performative
:sender <word>
:receiver <word>
:content <expression>
...)

¹ FIPA: Foundation for Intelligent Physical Agents

² ACL: Agent Communication Language

Figura 4-1 – Estrutura das Mensagem Utilizadas Neste Trabalho

O *:sender* e o *:receiver* são, respectivamente, os identificadores do agente que enviou a mensagem e do que recebeu a mensagem.

O *:content* (conteúdo) das mensagens pode ser um nó, um conjunto de nós, o resultado da função de utilidade de um agente, ou um *array* com as ociosidades dos nós relativas a um agente.

Quando um agente leiloeiro envia uma mensagem cuja performativa é *inform*, o conteúdo da mensagem será o nó que o agente está leiloando. Ao recebê-la, o receptor deve então verificar se trocar um de seus nós pelo informado maximiza sua função de utilidade. Caso isso aconteça, o receptor enviará uma mensagem cuja performativa será *propose*, pois estará propondo a troca do seu nó pelo informado. O conteúdo desta mensagem será o nó proposto em troca do informado. Caso não haja nenhuma troca vantajosa para o receptor, este enviará uma mensagem cuja performativa será *refuse*, informando que não deseja participar do leilão. Neste caso, o conteúdo da mensagem é o nó leiloado, mas este conteúdo é ignorado.

Depois que o leiloeiro recebe todas as propostas ou recusas, irá verificar qual o melhor lance. Para o agente que lançou a melhor proposta, o leiloeiro enviará uma mensagem de *accept-proposal* e para os demais, enviará uma mensagem de *reject-proposal*. Caso nenhuma proposta maximize a função de utilidade do leiloeiro, todas as mensagens enviadas como resposta serão *reject-proposal*.

4.5 CONFIGURAÇÕES DOS AGENTES NEGOCIADORES

Há quatro configurações principais que compõem os agentes negociadores que serão apresentados aqui. Estas variam de acordo com os quatro parâmetros seguintes: (1) seu comportamento quanto indivíduos em um grupo; (2) o protocolo do leilão; (3) como escolhem o nó a ser leiloado ou a oferecer para troca; e (4) seu algoritmo para determinar o percurso a ser percorrido. A seguir, descreveremos as possibilidades testadas para cada uma destas configurações neste trabalho.

4.5.1 Comportamento

O comportamento do agente pode ser em interesse próprio ou cooperativo. Em um sistema cooperativo, os agentes podem aceitar trocar um de seus nós por outro que diminua o valor de sua função de utilidade se a função de utilidade do grupo aumentar com esta troca. Ou seja, os agentes cooperativos consideram a função de utilidade do grupo mais do que a individual.

4.5.2 O Protocolo de Leilão

Os leilões podem ser de uma, duas rodadas, ou várias rodadas. Os nós leiloados têm valor privado que significa que o valor do nó no leilão depende da função de utilidade do agente. Os lances são secretos, cada licitante não sabe a oferta dos outros. Também, dependendo do protocolo do leilão, os agentes podem trocar, um nó por outro, dois para um, ou dois para dois. O agente é responsável por patrulhar um conjunto dos nós recebidos no começo da simulação. A distribuição inicial dos nós é randômica.

4.5.3 Escolha do Nó a Ser Leiloadado

Foram implementadas várias formas de escolher o nó a ser leiloadado.

Primeiro, foi implementado um algoritmo simples de ordenação (*insertion sort*), onde é calculada a distância que o agente tem que andar para visitar seus nós, removendo cada nó e recalculando a distância a fim de encontrar o nó que mais aumenta o trajeto.

Para seguir uma metodologia incremental, foi implementada uma alternativa mais simples e de menor custo computacional, utilizando-se o baricentro como base para esta decisão. O baricentro, ou centro de gravidade de um polígono, é uma coordenada cujo valor de x é encontrado somando-se os valores do eixo x dos pontos que definem o polígono e dividindo esta soma pelo número de pontos; o y é encontrado somando-se os valores do eixo y dos mesmos pontos e também dividindo esta soma pelo número de pontos. Seja o polígono p formado pelos pontos $(x_1, y_1), (x_2, y_2), \dots, (x_{n-1}, y_{n-1}), (x_n, y_n)$, sendo n o número de pontos que definem o polígono. O baricentro B de p é dado pela equação (7).

$$B_p = \left(\frac{\sum_{i=0}^n x_i}{n}, \frac{\sum_{i=0}^n y_i}{n} \right)$$

Equação 7

Neste caso, o nó a ser leiloado seria o mais distante do baricentro do polígono, definido pelo seu conjunto de nós.

Foi mostrado em [9] que o patrulhamento para um único agente é muito parecido com o problema do caixeiro viajante (*TSP – Travelling Salesman Problem*) [19]. O TSP é um problema de otimização combinatorial formulado da seguinte forma: dado um conjunto de cidades em um mapa deve-se encontrar o menor ciclo que passe por cada cidade exatamente uma vez.

Como os agentes negociadores partitionam o grafo, pode-se considerar que cada agente, na sua região, tem um problema parecido com o do caixeiro viajante. Como o custo computacional da solução do TSP proposta em [9] é muito alto, neste trabalho foi implementado o algoritmo de Kruskal [61] para encontrar a árvore de menor custo (*MCST – Minimum Cost Spanning Tree*) formada pelo conjunto de nós do agente, já que é provado que percorrer a MCST custa, no máximo, duas vezes mais que o caminho encontrado pelo melhor algoritmo para TSP [19, 61].

O algoritmo de Kruskal [61] consiste na construção de uma árvore que inclui todos os pontos minimizando o peso total das arestas da árvore. O algoritmo funciona da forma mostrada na Figura 4-2.

1. Cria um conjunto F de árvores, onde cada nó do grafo é uma árvore
2. Cria um conjunto S contendo todas as arestas do grafo
3. Enquanto S não é vazio
 4. Remove a aresta com o menor peso de S
 5. Se essa aresta conectar duas árvores diferentes, adicione em F , conectando duas árvores em uma
 6. Se não, descarte a aresta.

Figura 4-2 – Algoritmo de Kruskal

Ao final deste procedimento, o conjunto F só tem um componente, a MCST. Como o algoritmo constrói a árvore a partir das menores arestas, pode-se considerar que o nó que é inserido por último na árvore é um bom nó para ser leiloado.

Todos os algoritmos (Kruskal, cálculo baricentro, inserção) de escolha do nó a ser leiloado, apresentados nessa seção, computam apenas distância. Contudo, muitas vezes os conjuntos gerados pelas negociações dos agentes não são disjuntos. Conseguir conjuntos disjuntos depende da topologia do grafo e da quantidade de agentes. Então, nós de um agente podem estar no caminho que um segundo agente faz para visitar seus nós. Neste caso, os nós que já estão no caminho do segundo agente poderiam trocar de dono, facilitando o trabalho do primeiro de manter a ociosidade dos seus outros nós baixa. Por essa razão, a função de utilidade do agente foi estendida para levar em consideração a ociosidade além da distância.

Considerando Ociosidade na Escolha

Neste trabalho, foram testadas duas funções de utilidade para os agentes: considerando apenas distância e considerando ociosidade e distância.

Quando a função de utilidade considera apenas distância, ela é representada pelo negativo da soma das distâncias que o agente deve percorrer para visitar todos os seus nós uma vez.

Para que os agentes possam usar a ociosidade média de um nó para calcular seu valor, cada agente deve guardar as dez últimas ociosidades medidas em cada um de seus nós. Como ociosidade e distância estão em escalas diferentes de medida, foi necessário usar a idéia da conversão de escalas diferentes. Assim a função de utilidade é o negativo da soma do valor de cada nó do agente, como mostrado na equação (9), e o valor do nó é calculado como mostrado na equação (8).

$$NV = \alpha[(I - \text{ociosidade})/(I - i)] + (1 - \alpha)[(\text{distância} - d)/(D - d)].$$

Equação 8

$$F_u = -\sum_{i=0}^{Nós} NV_i$$

Equação 9

Na equação acima, I é a ociosidade média mais elevada entre todos os nós do agente, i é a ociosidade média mais baixa, D é a maior distância entre dois nós do agente, d a menor distância entre dois nós do agente. Dado que o menor caminho para percorrer os nós do agente já foi encontrado, $distância$ é a distância do nó anterior no percurso ao nó avaliado e α é o parâmetro que indica qual deve ser a importância da ociosidade no valor do nó. Durante os testes o vários valores para α foram testados a fim de se determinar, empiricamente, o melhor valor.

Melhora no Sistema de Escolha do Nô

Mesmo os agentes cooperativos têm uma limitação no que se refere a trocas. Eles oferecerão sempre seu pior nó, e às vezes trocas são necessárias ainda, mas não acontecem porque os agentes se mantêm oferecendo cada um o mesmo nó, já que a função de utilidade não muda. A fim de facilitar trocas do nó, cada agente randomicamente escolhe um nó após vinte ciclos sem trocar. Também foi implementada e testada a construção de uma fila da prioridade de nós a serem trocados de cada agente. A cada vinte ciclos sem trocas, o próximo nó da fila é oferecido em leilão.

4.5.4 Algoritmo de Percurso

Alguns procedimentos diferentes de escolher o trajeto dos agentes para visitar seu nós foram testadas.

O primeiro algoritmo de percurso implementado fazia os agentes andarem ciclicamente pelos seus nós seguindo o trajeto criado pelo algoritmo de inserção descrito na seção 4.5.3.

A segunda alternativa implementada e testada neste trabalho foi, ao invés de um percurso definido, fazer os agentes escolherem como destino o seu nó mais ocioso e visitá-lo. Essa idéia surgiu porque, por mais que os agentes façam boas divisões do grafo, muito dificilmente os conjuntos ficam realmente disjuntos, depende da topologia do grafo e do número de agentes. Então, em um percurso definido para visitar seus nós, um agente pode visitar nós de outros agentes. Essa forma de escolha de nó a ser visitado foi testada em [1, 6], mas naqueles trabalhos os agentes escolhiam o nó mais ocioso do grafo inteiro, e se não houvesse coordenação centralizada, todos os agentes se dirigiam para este nó.

Neste trabalho, cada agente escolhe o nó mais ocioso no conjunto que é responsável por supervisionar.

O terceiro procedimento implementado foi visitar o nó do agente com a ociosidade mais elevada considerando a distância, como o *HPCC (Heuristic Pathfinder Cognitive Coordinated)* [8, 12] descrito na seção 2.3.2. Este algoritmo de seleção de percurso, ou de nó a ser visitado, é uma evolução do segundo (descrito acima). Além de levar em consideração a ociosidade, também considera a distância de onde está até o nó a ser escolhido.

Por fim, foi implementado um algoritmo de percurso que faz o agente andar pela *MCST* (árvore de menor custo) formada pelos seus nós pelas razões explicadas na seção 4.5.3.

4.6 TIPOS DE AGENTES

Como será observado, o que difere os tipos de agentes implementados é o protocolo de leilão que cada um utiliza e o seu comportamento em relação ao grupo. Estes tipos de agentes negociadores foram testados com várias combinações de algoritmos de escolha de nós a serem leiloados e de visitar os nós pelos quais são responsáveis. Estas várias opções foram apresentadas na seção anterior.

Os tipos agentes negociadores para a tarefa da patrulha desenvolvidos neste trabalho, mostrados na ordem em que foram desenvolvidos, são:

4.6.1 Agente Leiloeiro

O agente BA (*Bidder Agent*) foi o primeiro a ser desenvolvido. Ele procura fazer a melhor troca para si. Sempre leiloa o nó que está prejudicando mais a sua função de utilidade, e sempre escolhe a troca que lhe traz mais vantagens. A função de utilidade de outros agentes não é levada em conta. O seu leilão é de apenas uma rodada.

4.6.2 Agente Leiloeiro Cooperativo

O agente CBA (*Cooperative Bidder Agent*) foi implementado para comparar o desempenho do sistema quando o comportamento é cooperativo ou não. Este agente, quando está leiloando um nó, leva em consideração o contentamento dos agentes

interessados no seu nó que está sendo leiloado. Analisa os lances do leilão procurando identificar não a melhor troca para si, mas a troca que aumenta o contentamento de todos os agentes no ambiente. Escolhe a troca que faz com que a soma da sua função de utilidade com a função de utilidade do agente que fez a proposta seja a maior possível. O seu leilão é de apenas uma rodada. Ele é em tudo similar ao BA, só difere no fato de ser cooperativo.

O agente cooperativo apresentou um desempenho pior que o agente não cooperativo.

4.6.3 Agente Negociador em Encontros

Este agente apenas se comunica com outro agente com o qual se encontra, ao invés de com vários outros, como os dois primeiros. Os agentes se encontram quando estão em um nó ao mesmo tempo.

Nesta abordagem, cada agente possui em sua base de conhecimento a visão que ele tem da ociosidade de cada nó, ou seja, ele armazena há quantos ciclos não visita cada nó. Conseqüentemente, os agentes têm visões diferentes das ociosidades de cada nó até que encontrem um parceiro. Ao se encontrarem, cada um informa a sua base de conhecimento ao outro, para que ambos fiquem com as mesmas informações sobre o mundo. Desta forma, os agentes conseguem uma visão mais próxima da realidade. Depois de atualizar a base, eles negociam como irão visitar os nós que eles acreditam que têm as maiores ociosidades, de forma que eles possam cobrir o maior número de nós possível.

Estes agentes se movem no grafo sempre em direção ao nó que não visitam há mais tempo que esteja mais próximo. A sua função de utilidade é definida pelo negativo da soma de suas ociosidades relativas.

Este agente não apresentou bons resultados e possui mais limitações que os outros, portanto, este estudo não foi levado adiante.

4.6.4 Agente Facilitador de Trocas

Foram testadas duas alternativas que faziam uso de facilitadores de trocas. Estes agentes intermediários têm uma visão mais abrangente do mundo, funcionando como um catálogo de agentes. Neste caso, o facilitador conhece todos os agentes que estão coexistindo no ambiente, bem como os nós que estão sob a responsabilidade de cada um destes agentes. Quando um patrulheiro resolve fazer um leilão, pergunta ao facilitador a

quais agentes deve propor a troca. Em seguida, o facilitador indica os agentes aos quais a troca pode trazer benefícios. A partir daí, ocorre um leilão direcionado.

Este tipo de alternativa, cria um gargalo, uma dependência em um agente para haver as trocas e, portanto, este estudo não foi levado à diante. Além disso, não apresentou bons resultados nas simulações.

4.6.5 Agente Leiloeiro de Dois Lances

Assim como o Agente Leiloeiro (BA), o agente *TSBA (Two-Shots Bidder Agent)* apenas oferece e aceita lances que melhorem sua função de utilidade, mas este promove duas rodadas de leilão com o objetivo de conseguir mais lances, e consequentemente, melhores ofertas.

Embora oferece dois lances em leilão, este agente apresentou resultados bem próximos aos do BA.

4.6.6 Agente Leiloeiro Flexível

O agente *FBA (Flexible Bidder Agent)* difere dos outros pelo número de nós trocados. Estes agentes podem trocar até dois nós em cada rodada de leilão, o que significa que o número de nós que estes agentes têm que patrulhar pode mudar durante a simulação. Podem trocar dois nós por outros dois, dois nós por um, ou um por outro. Este agente pode oferecer até dois nós em leilões, e estes leilões são de várias rodadas, já que os licitantes podem oferecer todas as possíveis trocas que lhes tragam vantagens por cada um dos nós oferecidos. Ele também tem o Agente Leiloeiro (BA) como base.

Este agente apresentou os melhores resultados entre todos os negociadores.

4.7 CONCLUSÃO

Os leilões são muito populares por serem cenários simples de interação, o que os tornam uma primeira escolha para ajudarem agentes a alcançarem acordos [25, 26]. Os agentes autônomos podem empregar leilões para distribuir tarefas e recursos entre eles. Cada agente negociador recebe randomicamente alguns nós do gráfico para patrulhar. A função de utilidade destes agentes é o negativo da distância que têm que andar para visitar todos os nós responsáveis por patrulhar. Em cada ciclo da simulação, cada agente

procurará pelo nó que está diminuindo sua utilidade. Uma vez que o nó é encontrado, o agente tenta negociá-lo. Os agentes podem negociar seus nós fazendo leilões a fim trocar seu pior nó por um melhor, mais próximo dos seus outros nós.

Um resumo das possíveis configurações dos agentes é mostrado no Quadro 2.

Quadro 4.2 - Resumo das principais características das abordagens propostas

Parâmetros de Configuração / Agentes	Número de lances no leilão	Comportamento cooperativo	Algoritmos de percurso	Algoritmo de escolha de nó a ser leiloado
BA	Um	Não	Todos implementados	Considera apenas distância
TSBA	Dois	Não	Todos implementados	Considera apenas distância
FBA	Vários	Não	Todos implementados	Considera distância e ociosidade
CBA	Um	Sim	Todos implementados	Considera apenas distância

De fato, os agentes negociadores podem dividir o grafo e, desta forma, satisfazem os critérios de: (1) escalabilidade, já que os agentes podem patrulhar mapas de quaisquer tamanhos com simples algoritmos de percurso, (2) adaptabilidade *offline*, já que não possuem pré-planejamento para visitarem seus nós, pois é uma abordagem flexível em sua essência e (3) apresenta uma estabilidade maior que as abordagens anteriores.

Neste capítulo, foram vistos os principais conceitos de comunicação e negociação que possibilitam a existência de sistemas multiagentes negociadores para patrulha, bem como os tipos de agentes implementados seguindo esta abordagem, bem como os principais critérios utilizados para avaliar seu desempenho.

No próximo capítulo, serão apresentados os cenários de simulação dos experimentos, os resultados obtidos e uma discussão sobre eles.

Capítulo 5

Experimentos e Resultados

Neste capítulo, serão apresentados os detalhes da implementação, cenários de simulação dos experimentos, os resultados obtidos e uma discussão sobre estes.

5.1 INTRODUÇÃO

Neste capítulo serão mostrados os resultados obtidos pelas simulações dos agentes propostos no capítulo anterior com várias configurações.

Em primeiro lugar, serão levantadas as hipóteses a serem verificadas ou refutadas pelos experimentos. Na seção seguinte, serão apresentados os cenários de simulação, os grafos utilizados nos experimentos. Em seguida, serão apresentados os experimentos realizados com os agentes da abordagem negociadora e os resultados obtidos que determinaram a escolha do melhor modelo de agente leiloeiro, bem como sua melhor configuração. Por fim, serão apresentados os resultados do melhor modelo de agente da abordagem negociadora, comparados aos resultados das abordagens anteriormente desenvolvidas.

Os resultados serão avaliados de acordo com os critérios de adaptabilidade *offline*, ociosidade média, escalabilidade e estabilidade descritos no capítulo 2.

5.2 HIPÓTESES

Os experimentos tiveram como objetivo avaliar os diferentes tipos de agentes negociadores, e verificar se as seguintes hipóteses são válidas:

1. O agente negociador flexível (FBA) é o que apresenta melhor desempenho entre todos os negociadores
2. Quanto mais baseado em MCST é o agente, melhor é seu resultado
3. A opção pela fila de prioridades, após vinte ciclos sem trocas, melhora o desempenho dos agentes
4. Escolher o nó a ser visitado com base na ociosidade e distância é a melhor estratégia de definição de próximo nó a ser visitado
5. Escolher o nó a ser leiloado com base na ociosidade e distância é a melhor estratégia de definição de próximo nó a ser leiloado
6. Os agentes negociadores são os que precisam de menos ciclos e pré-processamento para um novo grafo, ou seja, são mais adaptáveis

7. Os agentes negociadores são capazes de patrulhar grafos de qualquer tamanho, ou seja, formam sistemas com mais escalabilidade
8. Agentes negociadores formam sistemas estáveis

As primeiras cinco hipóteses serão verificadas na seção 5.4 que apresenta uma síntese dos experimentos realizados com a abordagem negociadora para determinar a melhor configuração de agente leiloeiro. As últimas três hipóteses serão verificadas na seção 5.5 onde serão comparados os resultados obtidos pela abordagem negociadora com os obtidos pelas abordagens de trabalhos anteriores.

5.3 IMPLEMENTAÇÃO

Neste trabalho, todas as implementações e simulações foram executadas no mesmo simulador usado nos trabalhos anteriores [1, 6, 8, 9, 13, 34]

O simulador da patrulha foi desenvolvido em [62] e vem sendo usado por todos os trabalhos apresentados na seção 2.3, com exceção do apresentado na seção 2.3.5, pois nele, a patrulha é feita por robôs. O simulador foi implementado utilizando a linguagem C++ [63] e OpenGL [64] como biblioteca gráfica.

Neste trabalho, foram implementados em C++ os tipos de agentes da abordagem negociadora descritos no capítulo 4, com todas as variações de configuração também descritas no capítulo 4. Também foram implementadas classes que representavam atos de fala a serem trocados pelos agentes.

Neste trabalho, também foram feitas correções de erros de implementação existentes no simulador que permitiam vazamento de memória.

5.4 CENÁRIOS DE SIMULAÇÃO

Nesta investigação, foram realizados experimentos variando o número e o tipo de agentes patrulheiros e o mapa do terreno.

No começo da simulação, todas as ociosidades instantâneas dos nós são zero, como se eles tivessem sido visitados no momento inicial. Conseqüentemente, há uma fase transitória em que a ociosidade instantânea do grafo tende a ser pequena e/ou a variar bastante, não correspondendo à realidade da fase estável. Na fase estável, os agentes

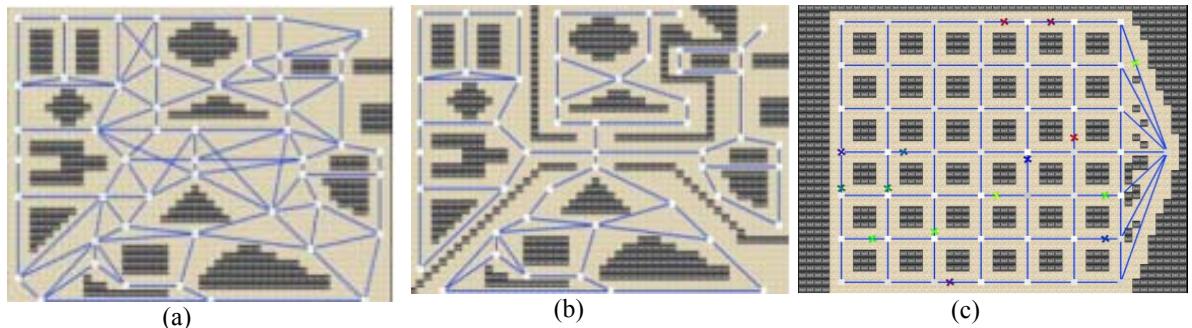
estão distribuídos pelos nós do grafo e a ociosidade média já não apresenta grandes variações.

Considerando o problema de qual a quantidade ideal de ciclos de uma simulação, uma alternativa razoável é visitar cada nó K vezes. No pior caso, um nó ficará PO iterações sem ser visitado, onde PO é a **pior ociosidade** da simulação. Uma aproximação relativa é escolher $K*PO$ ciclos. Um número preliminar de simulações foi executado com objetivo de identificar o valor de PO e percebemos que a média de PO é aproximadamente 1000, considerando todas as abordagens que foram comparadas (abordagens descritas na seção 2.3, além da implementada neste trabalho). Fixamos o valor de 15 para K de forma que cada nó seja visitado pelo menos 15 vezes, desta forma realizamos 15.000 ciclos em cada simulação.

Como cenários dos primeiros experimentos, foram utilizados os mapas da Figura 5-1: Regular (a), Regular com gargalos (b), Grade (c), Círculo (d), Corredor (e) e Ilhas (f). Esses mapas foram propostos em [8], são todos mapas de 50 nós e de topologias bem específicas e diferentes, para testar o impacto da topologia no desempenho dos agentes.

Os cenários foram testados com duas populações, de cinco e de quinze agentes, esses números foram escolhidos para simular ambientes menos ou mais populosos e verificar o desempenho das abordagens com a variação do número de agentes.

Como os nós e as posições iniciais dos agentes são escolhidos randomicamente, utilizamos cada cenário com a mesma população dez vezes, e consideramos a média dos resultados.



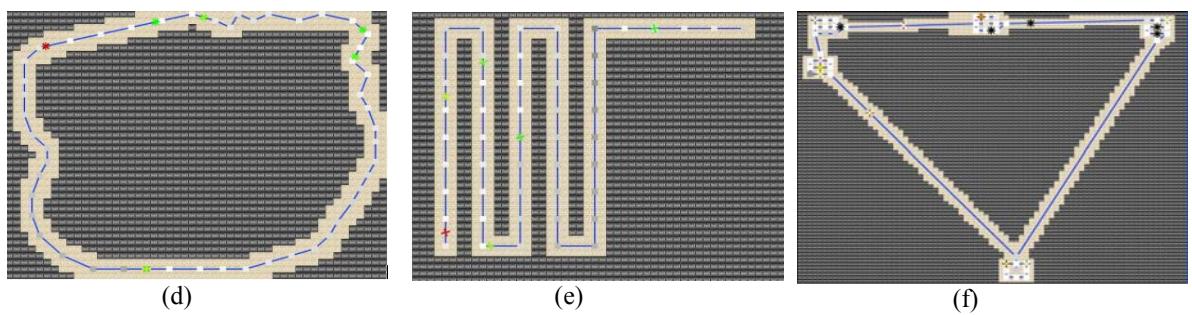


Figura 5-1 – Mapas utilizados nos experimentos

Para testar o critério de escalabilidade os experimentos foram também executados nos mapas da Figura 5-2: Mapa de 100 nós (a), Mapa de 200 nós (b) e Mapa de 500 nós (c). Os cenários foram testados com populações de 10, 20 e 50 agentes, respectivamente, para manter a proporção de poucos agentes por número de nós dos primeiros experimentos.

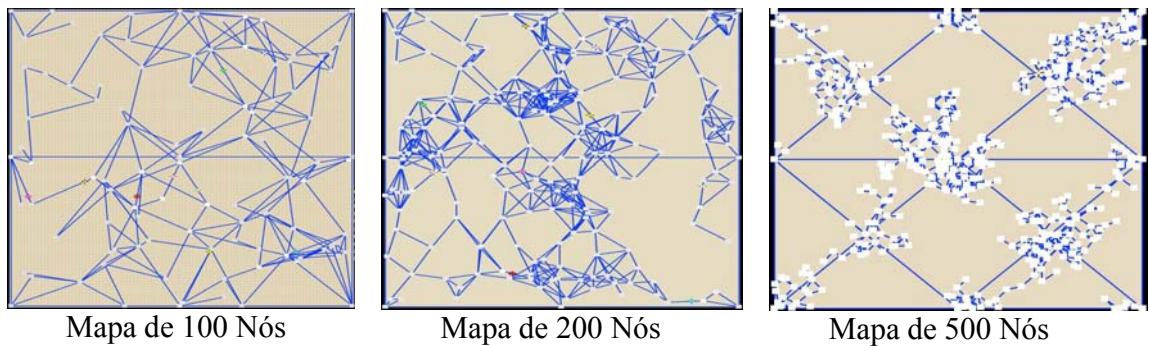


Figura 5-2 – Mapas utilizados nos experimentos

Primeiro foram realizados experimentos comparando o desempenho dos agentes negociadores entre si, com as variações e escolha de nó a ser leiloado e escolha de nó a ser visitado para determinar a melhor configuração entre os agentes leiloeiros propostos. Depois foram executados experimentos para comparar o desempenho da abordagem implementada neste trabalho com as abordagens anteriores [1, 8, 9, 13]. Na próxima seção, serão mostrados os resultados obtidos.

5.5 ENCONTRANDO O LEILOEIRO DE MELHOR DESEMPENHO

Nesta seção, mostraremos os resultados obtidos quando comparados os quatro tipos de agentes da abordagem negociadora (seção 4.7) com diferentes configurações. Foram testadas, para cada configuração, populações de cinco e quinze agentes.

Os agentes negociadores baseados em facilitadores e comunicador em encontros tiveram seus resultados omitidos. Os agentes negociadores baseados em facilitadores criam um gargalo, uma dependência em um agente para haver as trocas e, portanto, este estudo não foi levado à diante. Além disso, não apresentou bons resultados nas simulações. Os agentes comunicadores em encontros não apresentaram bons resultados e possuem mais limitações que os outros.

Para comparar os agentes leiloeiros entre sim foi utilizado o critério de ociosidade média.

5.5.1 Hipótese 1 – Melhor Leiloeiro

Para confirmar a hipótese que o agente negociador flexível (FBA) é o que apresenta melhor desempenho entre todos os negociadores (pois possui um protocolo de leilão mais flexível) foram simulados diversos cenários com os quatro tipos de agentes: Agente Leiloeiro Flexível (FBA), Agente Leiloeiro de Dois Lances (TSBA), Agente Leiloeiro Cooperativo (CBA) e Agente Leiloeiro não-cooperativo (BA), usando todas as configurações propostas.

Nesta configuração de teste usada como exemplo, os agentes definem um caminho e o nó a ser leiloado de acordo com o algoritmo de inserção já descrito na seção 4.5.

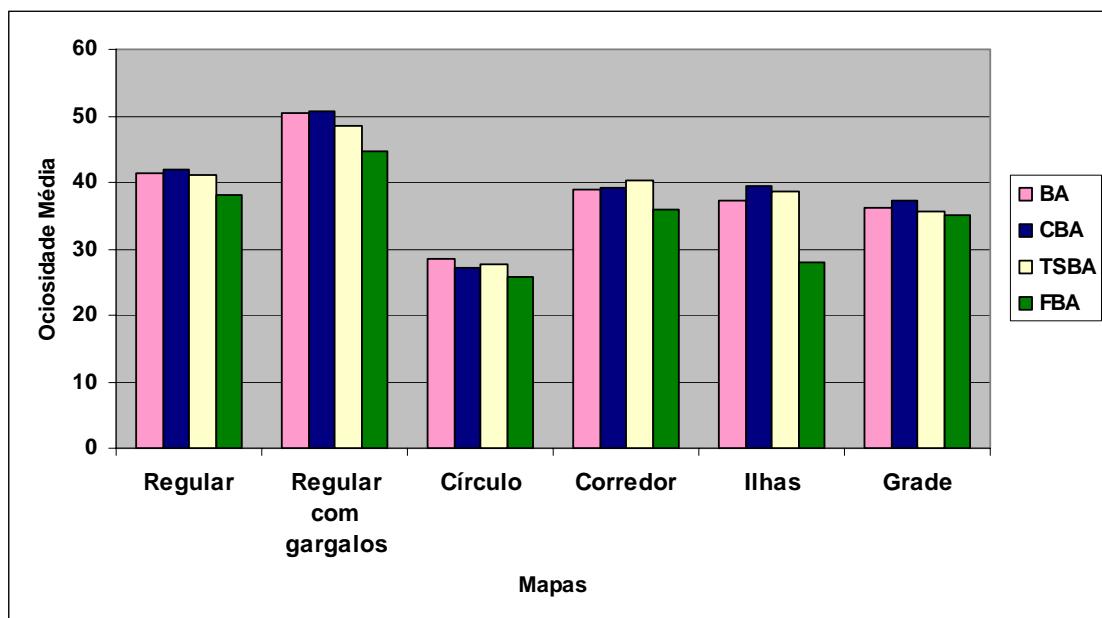


Figura 5-3– Comparação entre desempenho dos agentes (BA, CBA, TSBA, FBA) em termos de ociosidade média, na simulação com população de 5 agentes

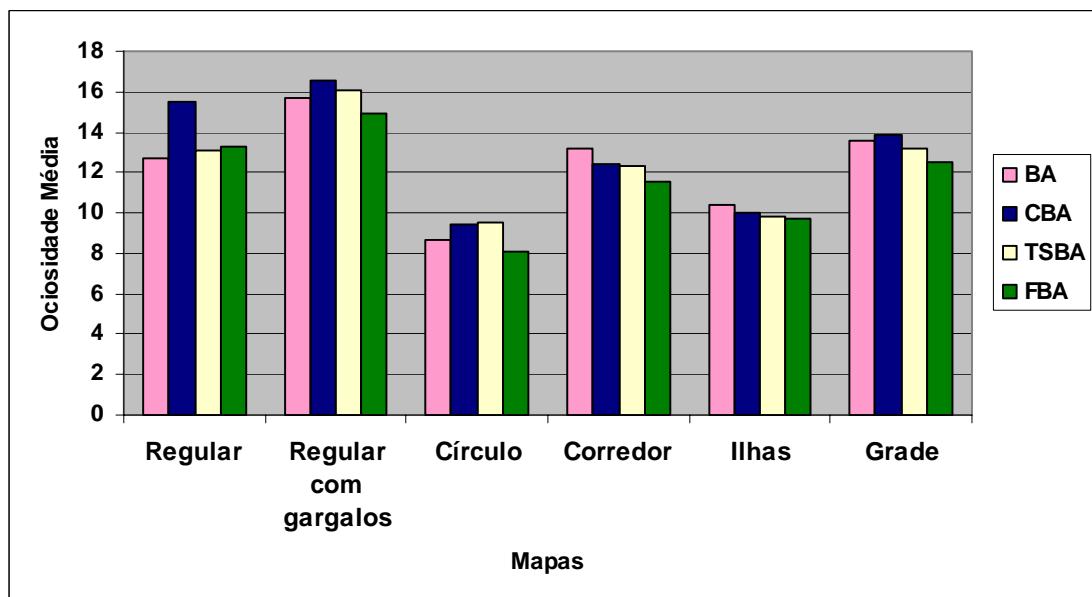


Figura 5-4– Comparação entre desempenho dos agentes (BA, CBA, TSBA, FBA) em termos de ociosidade média, na simulação com população de 15 agentes

Como pôde ser visto nas figuras 5-3 e 5-4, o agente negociador não cooperativo (BA) tem melhor desempenho do que o agente cooperativo (CBA) na maioria dos mapas. Por isso, este agente foi escolhido para ser estendido, para melhorar seu protocolo de leilão gerando os agentes leiloeiros Flexível (FBA) e de Dois Lances (TSBA) .

Também, pode-se notar que o agente negociador flexível (FBA) é o que apresenta melhor desempenho entre todos os negociadores na maioria dos mapas. Desta forma, este agente foi o escolhido para testar a maioria das hipóteses apresentadas na seção 5.2.

5.5.2 Hipótese 2 - Agentes Baseados em MCST Obtêm Melhores Resultados

Para verificar a segunda hipótese, os agentes escolhem o seu próximo nó a ser visitado e/ou o nó a ser leiloado de acordo com a árvore gerada pelo algoritmo de Kruskal a MCST. De acordo com este algoritmo, a última aresta a ser inserida é a maior aresta, logo o agente tenta negociar o último nó inserido na árvore. Após vinte ciclos sem trocas, os agentes ofereceram um nó escolhido randomicamente.

Para verificar esta hipótese, foi escolhido o agente leiloeiro não cooperativo mais simples (BA). Nas figuras 5-5 e 5-6, chamamos de MMBA o Agente Leiloeiro que escolhe tanto o nó a ser leiloado, quanto o percurso baseado em MCST. Como pode ser notado o seu desempenho é bem pior que o do Agente Leiloeiro que apenas escolhe o percurso baseado na MCST (nas figuras 5-5 e 5-6, representado pela sigla MBA), que ainda apresenta um pior resultado que o agente que anda para o seu nó mais ocioso considerando também distância (na figura, representado pela sigla BA).

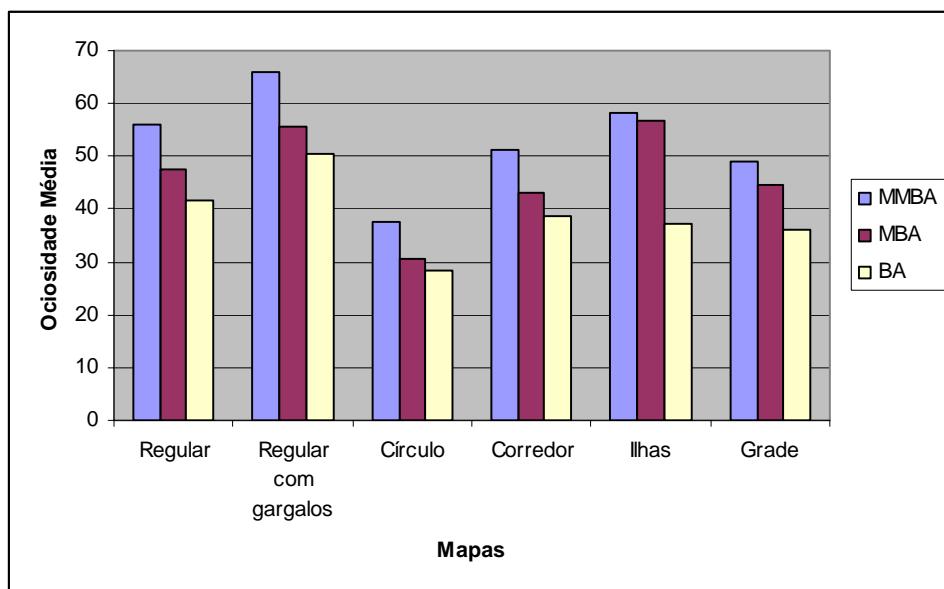


Figura 5-5 – Comparação entre desempenho da Ociosidade média dos agentes (BA, MBA, MMBA) na simulação com população de 5 agentes

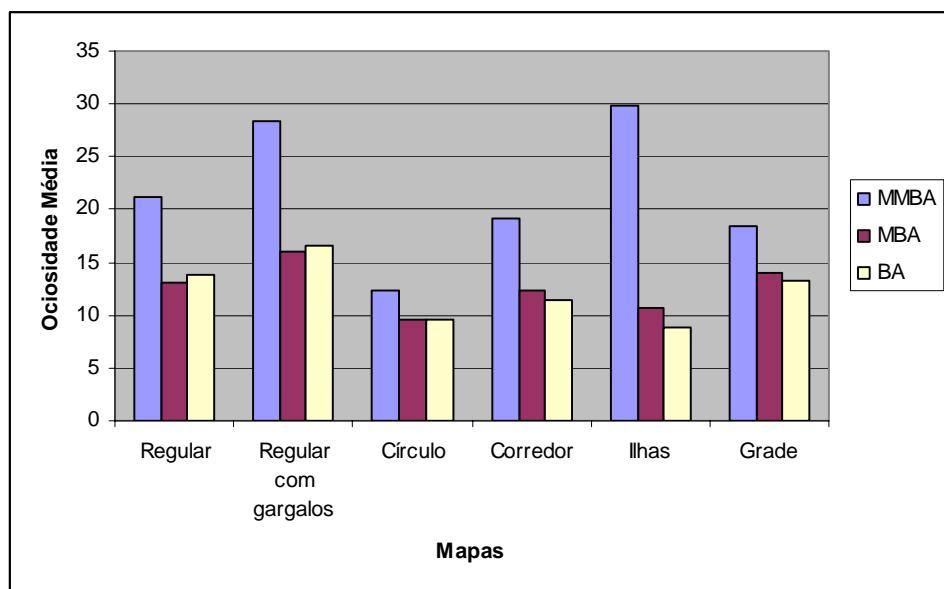


Figura 5-6 – Comparação entre desempenho da Ociosidade média dos agentes (BA, MBA, MMBA) na simulação com população de 15 agentes

Como pode ser visto, para uma população de cinco agentes, quanto mais baseado em MCST é o agente, pior é seu resultado. Ou seja, a hipótese de que o algoritmo de Kruskal traria benefícios para os agentes negociadores foi refutada. Esse resultado, mostrado aqui

com o tipo mais simples de agente, foi o mesmo resultado obtido pelos demais tipos, portanto não serão mostrados os demais experimentos realizados. O algoritmo de Kruskal não trouxe bons resultados, além de aumentar a complexidade da escolha do próximo nó a ser visitado e do nó a ser leiloado.

5.5.3 Hipótese 3 – Fila de Prioridades Melhora o Desempenho dos Agentes

Neste experimento, os agentes definem um caminho e escolhem para ser oferecido em leilão, o nó que minimiza a função de utilidade do agente identificado pelo algoritmo de ordenação descrito na seção 4.5.

Neste experimento, pretende-se verificar a hipótese de que a fila de prioridades traz um melhor resultado que a escolha randômica, após vinte ciclos sem trocas.

Nas figuras 5-7 e 5-8, após vinte ciclos sem trocas, os agentes RFBA (Agente Leiloeiro Flexível Randômico) ofereceram um nó escolhido randomicamente, enquanto os agentes PQFBA (Agente Leiloeiro Flexível com Fila de Prioridade) ofereceram o nó da vez na suas filas de prioridade.

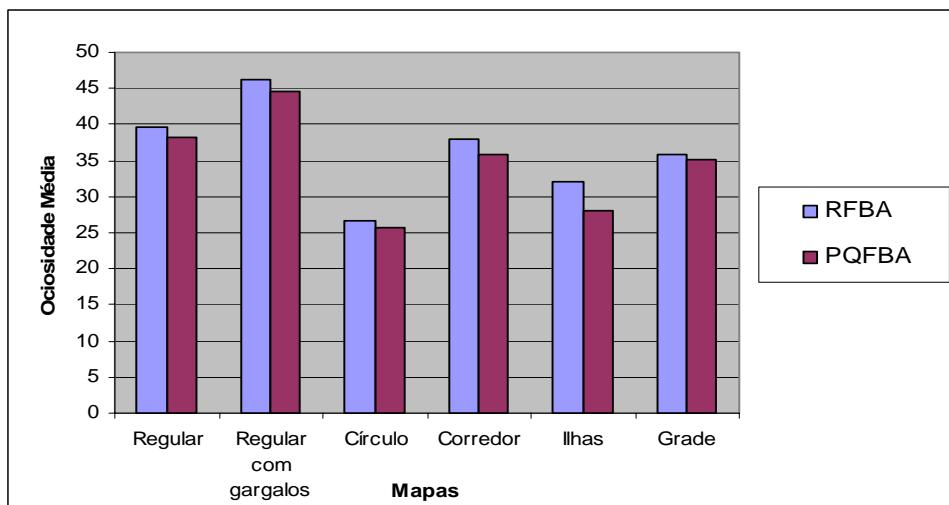


Figura 5-7 – Comparação do desempenho da Ociosidade média dos agentes (RFBA, PQFBA) na simulação com população de 5 agentes

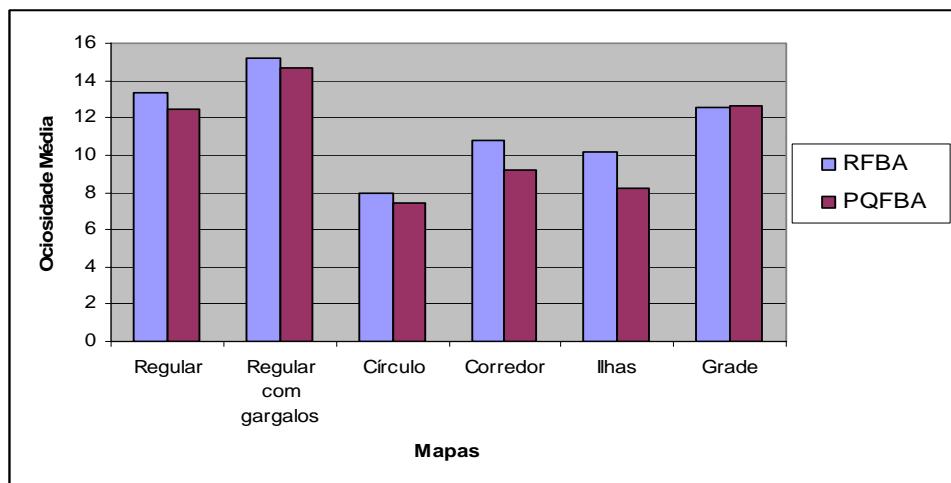


Figura 5-8– Comparação do desempenho da Ociosidade média dos agentes (RFBA, PQFBA) na simulação com população de 15 agentes

Pode-se observar, pelos gráficos, que a opção pela fila de prioridades traz uma melhora de até 20% no desempenho dos agentes.

5.5.4 Hipótese 4 – Uso de ociosidade e distância é a melhor estratégia na escolha de próximo nó a ser visitado

Para verificar esta hipótese, foram comparadas três estratégias de escolha de nó a ser visitado, descritas na seção 4.5.4. O agente FBA foi testado com essas três estratégias de percurso. O agente LFBA é o agente FBA que escolhe visitar o seu nó mais ocioso independente de distância. O agente CFBA anda em ciclos pelos seus nós seguindo o trajeto criado pelo algoritmo de inserção descrito na seção 4.5.3. O agente HFBA visita seu nó com ociosidade mais elevada considerando a distância, como o HPCC (*Heuristic Pathfinder Cognitive Coordinated*) descrito na seção 2.3.2.

Neste experimento, é escolhido para ser oferecido em leilão o nó que minimiza a função de utilidade do agente, identificado pelo algoritmo de ordenação já descrito. Nas figuras 5-9 e 5-10, após vinte ciclos sem trocas, os agentes ofereceram um nó escolhido randomicamente.

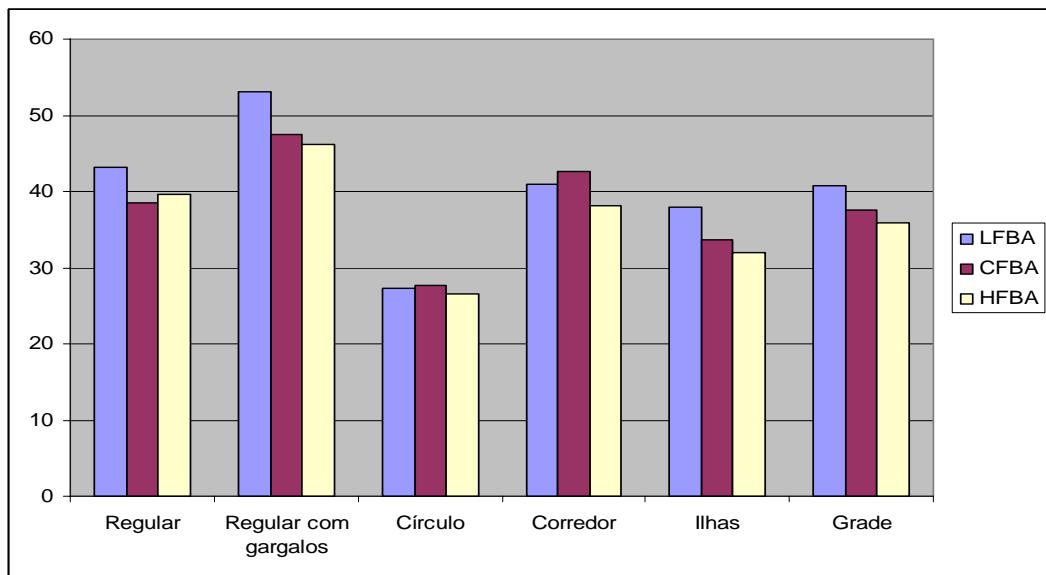


Figura 5.9– Comparação do desempenho da Ociosidade média dos agentes (LFBA, CFBA, HFBA) na simulação com população de 5 agentes

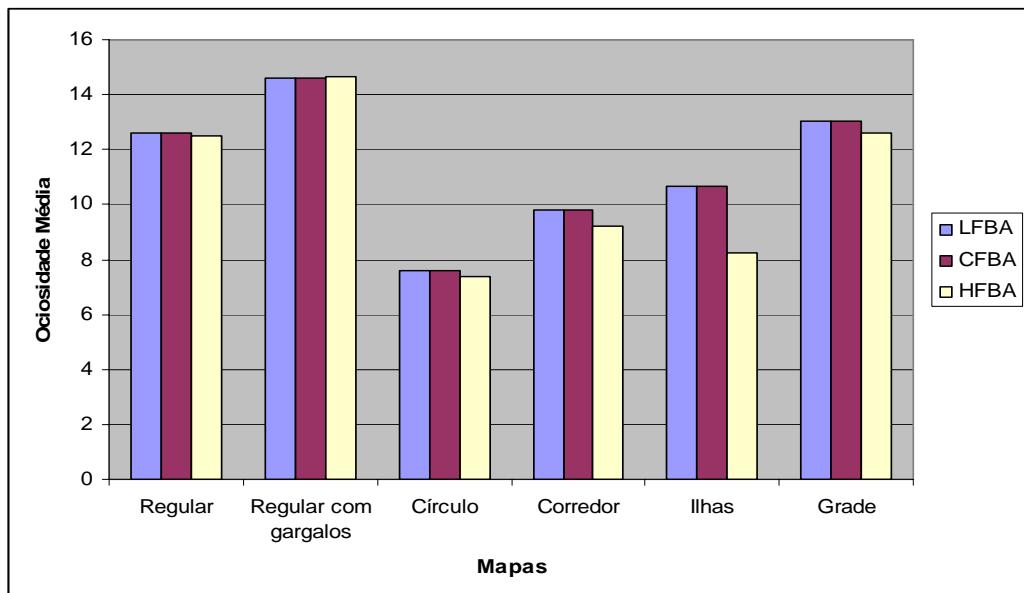


Figura 5.10– Comparação do desempenho da Ociosidade média dos agentes (LFBA, CFBA, HFBA) na simulação com população de 15 agentes

A partir destes resultados, entende-se que escolher o próximo nó a ser visitado com base apenas na ociosidade (LFBA) não é uma boa estratégia, sendo melhor escolher apenas pela distância (CFBA), quando o mapa estiver sendo patrulhado por poucos

agentes. Quando o mapa está sendo patrulado por muitos agentes, não há diferença significativa entre esses dois resultados.

Quando o agente FBA é combinado com o algoritmo de definição de percurso do HPCC, apresenta melhor resultado. Comparando o HFBA com o LFBA, o desempenho do primeiro chega a ser 20% superior, além de ser 10% superior ao desempenho do CFBA. Portanto, é esta configuração do FBA que será estendida para considerar também ociosidade em sua função de utilidade. A configuração estendida será comparada com as abordagens de trabalhos anteriores.

5.5.5 Hipótese 5 – Uso de ociosidade e distância é a melhor estratégia na escolha do nó a ser leiloado

Neste experimento, é demonstrada a melhora no desempenho do Agente Leiloeiro Flexível (FBA) com o uso de ociosidade na função de utilidade. Foi escolhido o melhor tipo de agente (FBA), o melhor critério de escolha de percurso, que, como visto até agora, é visitar o nó mais ocioso considerando distância.

No gráfico, são mostrados o Agente Leiloeiro Flexível (FBA) e o Agente Leiloeiro Flexível que considera ociosidade em sua função de utilidade (IFBA). O FBA escolhe o nó a ser leiloado pelo algoritmo de ordenação descrito anteriormente. A escolha do nó a ser leiloado baseada no algoritmo de Kruskal (seção 5.4.2) não apresentou bom desempenho, portanto os resultados não serão mostrados novamente aqui. A escolha baseada no baricentro, como sugerida na seção 4.5.3, também não apresentou bons resultados, e estes também serão omitidos.

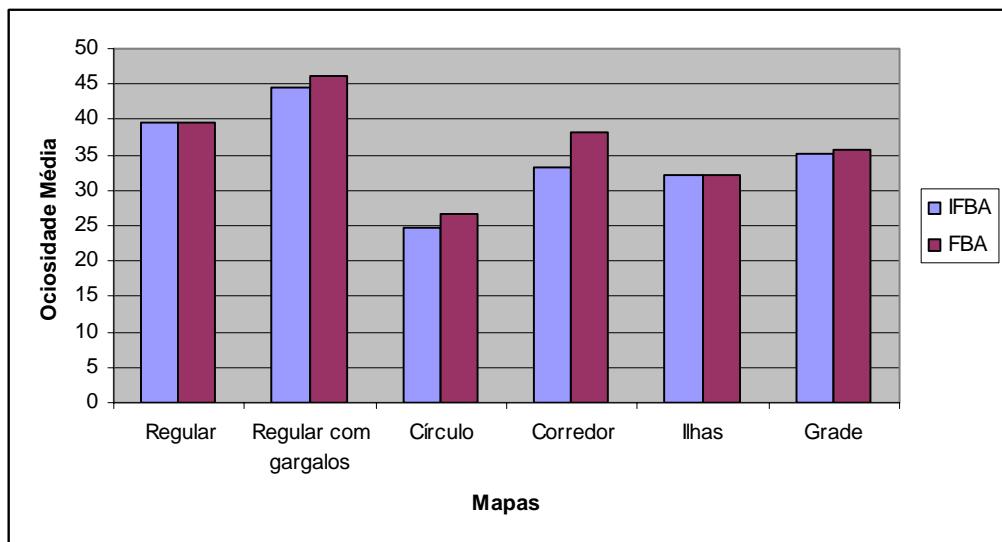


Figura 5-11 – Comparação do desempenho da Ociosidade média dos agentes (FBA, IFBA) na simulação com população de 5 agentes

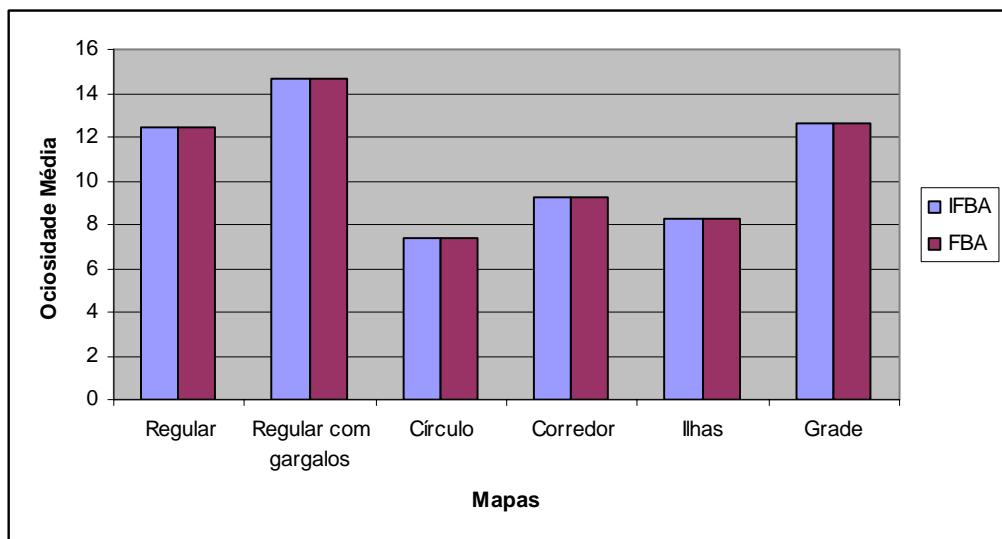


Figura 5-12 – Comparação do desempenho da Ociosidade média dos agentes (FBA, IFBA) na simulação com população de 15 agentes

Considerar a ociosidade média apresentou resultados de até 20% de melhora no desempenho dos agentes para simulações com 5 agentes. Entretanto, nos mapas com 15 agentes, adicionar ociosidade à função de utilidade trouxe melhora de, no máximo, 2%. Ou seja, quanto mais agentes há patrulhando o grafo menos importante é considerar a ociosidade.

5.5.6 Discussão

Pelos gráficos mostrados nas seções anteriores, pode-se concluir que: (1) o melhor tipo de agente negociador é o Agente Leiloeiro Flexível; (2) a melhor forma de percorrer os nós é cada agente se dirigir ao seu nó mais ocioso considerando a distância, que é o algoritmo de percurso do HPCC (seção 2.2.2) implementado por uma abordagem descentralizada; (3) o melhor algoritmo de escolha de nó a ser leiloado é o mais simples, baseado em *insertion sort* (seção 4.5.4); (4) a escolha do nó a ser leiloado é complementada por uma fila de prioridades de nós que indica qual nó deve ser oferecido, caso se passem mais de 20 ciclos sem trocas. Esta configuração será chamada apenas de Agente Leiloeiro Flexível (FBA) a partir de agora.

Ainda em relação à função de escolha do nó a ser leiloado, vale lembrar que considerar a ociosidade torna-se menos necessário com o aumento do número de agentes, como demonstraram os experimentos. Ainda assim, o FBA considera ociosidade na sua decisão nos experimentos que serão mostrados na próxima seção.

Na próxima seção, o Agente Leiloeiro Flexível (FBA) será comparado com os agentes das abordagens anteriores de patrulha.

5.6 ANÁLISE COMPARATIVA ENTRE ABORDAGENS

Para esta comparação, foi escolhida a melhor configuração de agente negociador para a patrulha, visto na seção 5.5.6, FBA.

Nesta fase de comparação, o objetivo é verificar as seguintes hipóteses: os agentes negociadores, representados pelo FBA, são mais estáveis, escaláveis e adaptáveis.

Os agentes que serão comparados ao FBA são os melhores de cada abordagem apresentada no capítulo 2. São eles: o agente HPCC (*Heuristic Pathfinder Cognitive Coordinated Agent*), GBLA (*Gray-Box Learner Agent*) e SC (*Single-Cycle Agent*). Estes agentes estão descritos nas seções 2.2.2, 2.2.3 e 2.2.4, respectivamente. Os resultados obtidos por estes agentes, nos mapas da Figura 5-1, estavam disponíveis desde [8]. Para a comparação de escalabilidade foi solicitado aos desenvolvedores de cada um deles que simulassesem seus agentes nos novos mapas.

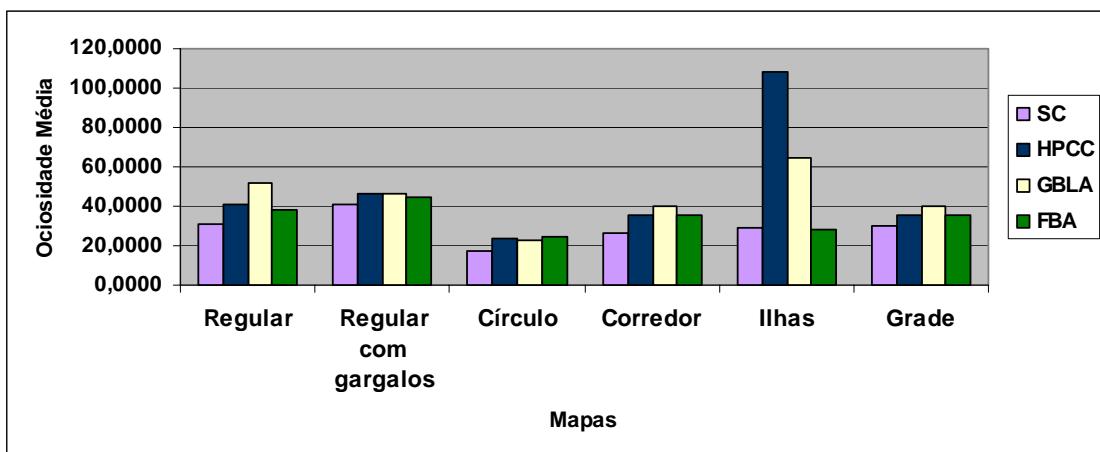


Figura 5-13 – Comparação entre desempenho dos agentes (SC, HPCC, GBLA, FBA) quanto a Ociosidade média em simulações com 5 agentes

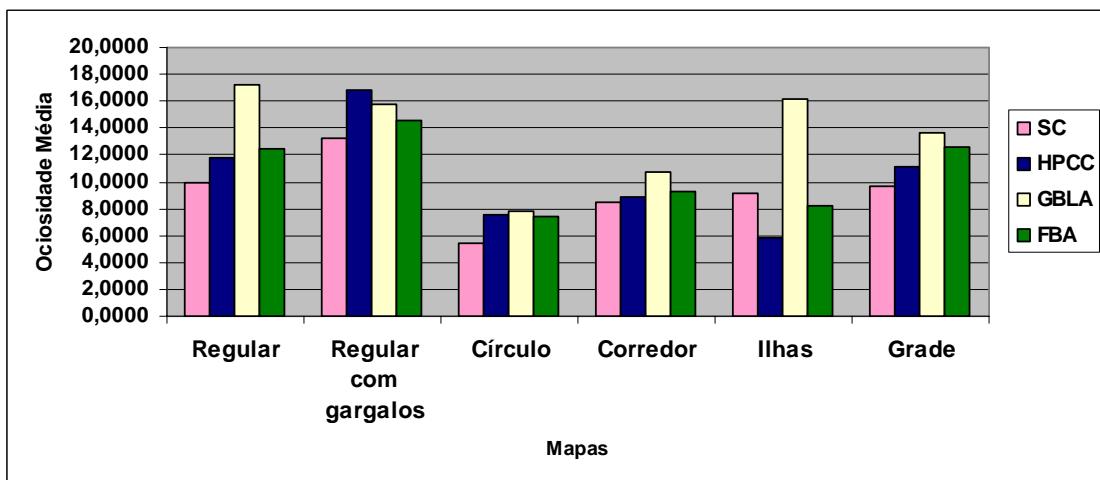


Figura 5-14 – Comparação entre desempenho dos agentes (SC, HPCC, GBLA, FBA) quanto a Ociosidade média em simulações com 15 agentes

Está claro, por estes gráficos, que o Agente *Single-cycle* (SC) obtém o melhor desempenho para todos os casos de ociosidade média, a não ser em duas situações: (1) no mapa de Ilhas com cinco agentes, onde o Agente Leiloeiro Flexível (FBA) tem o melhor resultado; e (2) no mapa Ilhas com quinze agentes, onde perde para os agentes HPCC e para os agentes FBA.

A respeito do FBA, para as simulações com cinco agentes, o FBA tem o segundo melhor desempenho em todos os mapas, exceto no Círculo, onde apresenta o pior

resultado e no Ilhas, onde tem o melhor. Para 15 agentes, o FBA é o segundo melhor em três mapas: Círculo, Ilhas e Regular com gargalos.

Os agentes negociadores não precisam de coordenação central como o HPCC ou o SC, portanto não possuem o gargalo da centralização de ações. A falha do coordenador no SC (antes do início da simulação) e no HPCC (durante a simulação) fará o sistema parar de trabalhar.

O Agente Leiloeiro Flexível apresentou resultados melhores do que GBLA, a outra abordagem distribuída, em todos os cenários de teste e tem a vantagem de não necessitar de tempo da aprendizagem (pré-processamento).

Os SMA negociadores são mais estáveis, enquanto o SC, GBLA e o HPCC exibem uma variação equivalente do desempenho (o desvio padrão varia 0,77 a 0,99 em todos os grafos) [8], o FBA apresentou o desvio padrão que varia entre 0,05 a 0,35.

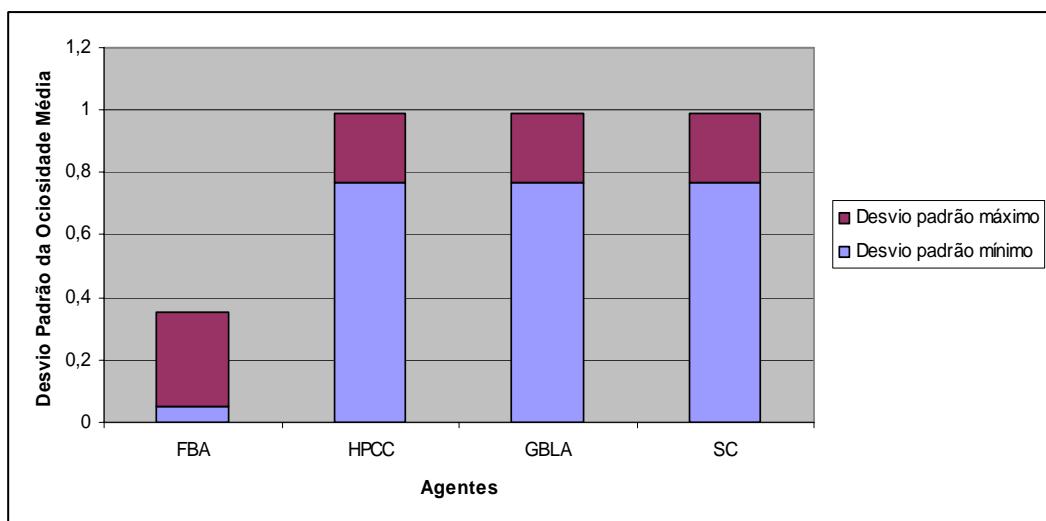


Figura 5-15 – Variação do desvio padrão da ociosidade média. Estes resultados são a média para todos os mapas

A respeito do critério de adaptabilidade *offline*, o FBA é a melhor alternativa, porque os agentes negociadores não necessitam de nenhum tempo da aprendizagem, nem preparação de um novo percurso específico para ser capaz de patrulhar em cada topologia de mapa como o SC. Neste aspecto, o HPCC embora não necessite de preparação de estratégia para cada novo mapa, gasta mais ciclos que os negociadores para estabilizarem o patrulhamento. Consideramos ainda como *offline* a fase antes do sistema se estabilizar. A figura 5-16, mostra a quantidade de ciclos necessária para os agentes FBA se

estabilizarem e uma comparação com os agentes HPCC. Por volta do ciclo 200, o FBA está completamente adaptado ao mapa e manterá sua ociosidade estável. Por sua vez, o HPCC continua variando a sua ociosidade ainda por volta do ciclo 5000.

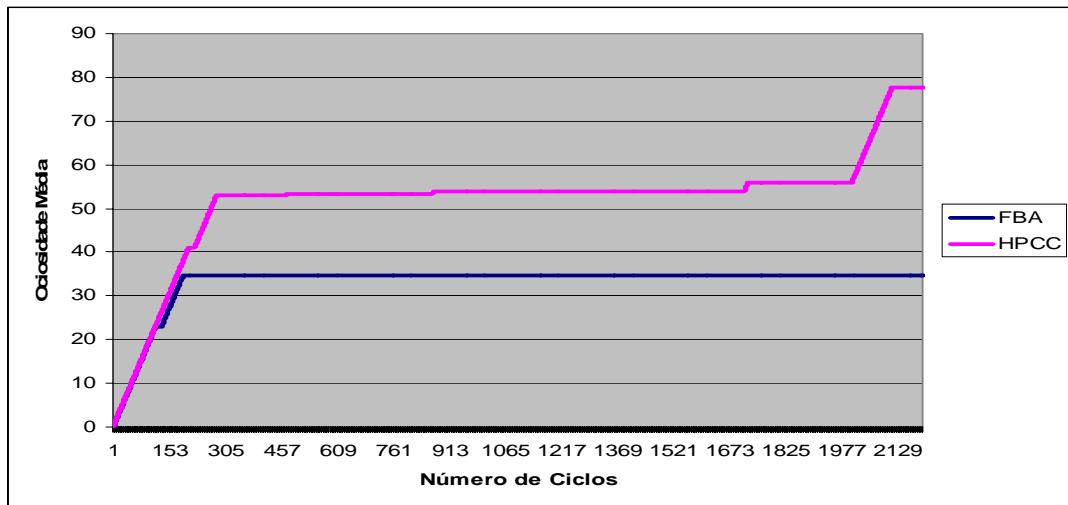


Figura 5-16 – Quantidade de ciclos necessária para os agentes FBA e HPCC se estabilizarem.
Simulação com cinco agentes no mapa Ilhas

Embora não tenhamos testado explicitamente o critério de adaptabilidade *online*, está claro que o SC teria que parar a simulação e encontrar um outro ciclo para continuar patrulhando e o GBLA requereria também nova etapa de aprendizagem. Enquanto os negociadores, aqui representados pelo FBA, e o HPCC, por gerarem seu percurso ao decidirem que nó visitar, facilmente se adaptariam a mudanças na topologia do grafo durante o patrulhamento.

Considerando escalabilidade, os agentes negociadores podem patrulhar em mundos de vários tamanhos. O FBA foi testado em um mundo com 100, um de 200 e em outro com 500 nós os resultados são mostrados em Figura 5-17. O HPCC não conseguiu patrulhar o mundo de 500 nós, o GBLA já não pôde patrulhar um mundo de 200 nós, pois ao usar *QTables* na fase de aprendizagem consome muita memória, o que torna esta fase impraticável, o que não aconteceria se o algoritmo de aproximação fosse uma rede neural, por exemplo. O SC, porém não foi testado, mas espera-se que tivesse um bom desempenho, mas que muito tempo fosse consumido para gerar a melhor rota.

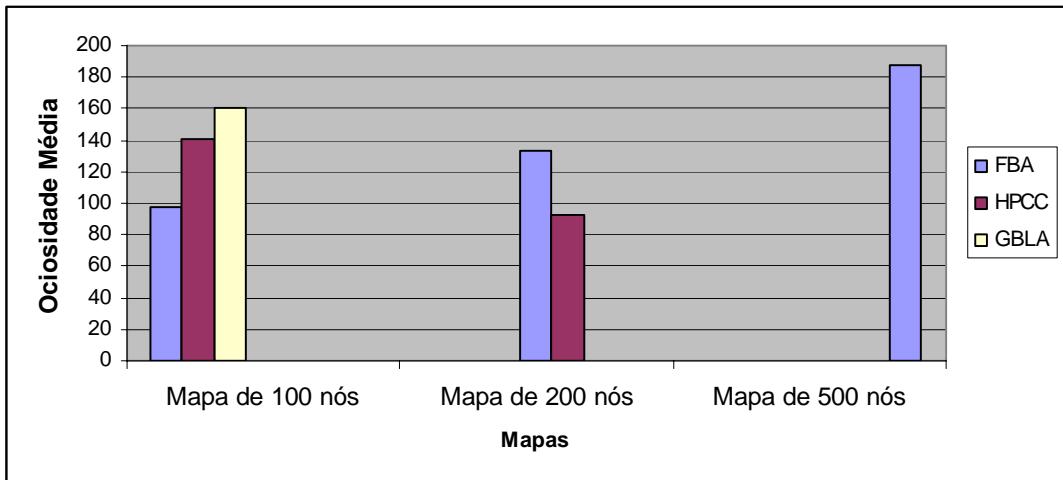


Figura 5-17 – Resultados das simulações em mapas grandes. No Mapa de 100 nós, a população era de 10 agentes, no Mapa de 200 nós, a população era de 20 agentes e no Mapa de 500 nós de 50 agentes

Com relação os critérios propostos por este trabalho os agentes negociadores se mostraram mais estáveis, adaptáveis e escaláveis.

5.7 CONCLUSÃO

Como pôde ser visto neste capítulo, o trabalho que adotou a abordagem baseada em negociação rendeu resultados bons quando comparados aos anteriores. A habilidade de comunicação dos agentes foi explorada para execução de leilões que permitiram o particionamento dinâmico e natural do grafo pelos agentes em tempo de execução.

Foram testadas várias estratégias de escolha do nó a ser leiloado, como pela distância ao baricentro, ou pela MCST. Também foram testadas várias estratégias de escolha do próximo nó a ser visitado, como andando ciclicamente, pela MCST, para o nó mais ocioso e para o nó mais ocioso considerando distância.

Além disso, estendeu-se o algoritmo de escolha do nó a ser leiloado primeiro com uma escolha randômica, e depois com a implementação de uma fila de prioridades, caso o agente passasse mais de vinte ciclos sem trocar nenhum nó.

Todas as decisões baseadas em MCST mostraram resultados ruins, bem como a escolha do nó a ser leiloado baseado no baricentro do polígono formado pelos nós do agente, cujos resultados foram omitidos, por terem sido ainda piores.

O agente que mostrou melhor resultado foi FBA com os algoritmos de escolha do nó a ser leiloado e do próximo nó a ser visitado baseados em ociosidade e distância. Este agente foi comparado com as abordagens de trabalhos anteriores, e apresentou bons resultados com respeito à ociosidade média.

Com relação aos critérios propostos por este trabalho, foi mostrado que os agentes negociadores, em geral, são mais estáveis, adaptáveis e escaláveis.

Capítulo 6

Conclusão e Trabalhos Futuros

Neste capítulo, serão apresentados os resultados alcançados, as dificuldades encontradas e os trabalhos futuros.

6.1 RESULTADOS ALCANÇADOS

Neste trabalho, uma solução baseada nos mecanismos de negociação foi desenvolvida, estendida e integrada com técnicas de decisão avançadas e técnicas *pathfinding*.

Na patrulha multiagente baseada em negociação, cada agente recebe alguns nós do gráfico para patrulhar. A função de utilidade destes agentes considera a distância que tem que ser percorrida para visitar todos os nós responsáveis por patrulhar, e também a ociosidade destes nós. Em cada ciclo da simulação, cada agente negociará o nó que está diminuindo sua utilidade, ou seja, aumentando o percurso ou com a alta ociosidade média. Os agentes podem negociar seus nós fazendo leilões a fim trocar seu pior nó por um melhor, mais próximo dos seus outros nós.

Os leilões são muito populares por serem cenários simples de interação, os que lhes torna uma primeira escolha para ajudar a agentes alcançar accordos. Os agentes autônomos podem empregar leilões para distribuir tarefas e recursos entre eles.

Este trabalho também propôs novos critérios de avaliação de sistemas multiagentes para a tarefa da patrulha. Foi apresentada uma discussão a respeito de escalabilidade, adaptabilidade e estabilidade.

Os agentes desenvolvidos neste trabalho foram testados primeiro nos mapas propostos em trabalhos anteriores [8], e por mapas maiores implementados neste trabalho, com o intuito de avaliar esta solução e as anteriores com relação a escalabilidade.

De fato, os agentes negociadores podem dividir o grafo e, desta forma, apresentaram uma estabilidade maior que as abordagens anteriores. Além disso, essa abordagem também satisfaz os critérios de escalabilidade, já que os agentes podem patrulhar grafos muito grandes. Bem como adaptabilidade, já que não possuem pré-planejamento para visitarem seus nós. É uma abordagem flexível em sua essência.

Este trabalho apresentou bons resultados que mostram que as alternativas baseadas em negociação implementadas são interessantes e nos dão direções para estender esta pesquisa investindo nos agentes leiloeiros, pois rendeu resultados bons quando comparados àqueles disponíveis anteriormente, tanto por seu desempenho avaliado pelo

critério ociosidade média, até então considerado mais importante, quanto quando avaliado de acordo com os novos critérios propostos por esse trabalho.

Os sistemas de agentes negociadores provaram ser adaptáveis, estáveis e escaláveis. E embora o critério de adaptabilidade *online* não tenha sido avaliado, não há nenhuma limitação na implementação da abordagem que impeça que este critério seja atendido.

Este trabalho gerou dois artigos científicos [8, 65].

6.2 DIFICULDADES ENCONTRADAS

Como visto, o Simulador foi implementado em C++, linguagem escolhida por ser orientada a objetos e proporcionar a criação de softwares de melhor desempenho (mais rápidos). Entretanto, ao construir softwares utilizando esta linguagem, o desenvolvedor precisa se preocupar com alocação e liberação de memória. A forma como o simulador foi implementado dificulta o controle de memória, o que favorece a ocorrência problemas de *memory leak*.

Muito esforço de implementação foi feito para minimizar este problema, pois era necessário simular diversos cenários. Contudo, o acumulo de memória não liberada, não permitia que isto fosse feito automaticamente. Alguns métodos do simulador foram reescritos, mas não foi possível resolver todos os problemas de vazamento memória. Para contornar esta situação, foi necessário desenvolver um script. Para executar o simulador para cada cenário. Desta forma, sempre que o simulador simula um cenário, o programa é encerrado, o que liberando a memória alocada por ele, e novamente inicializado para simular um outro cenário.

Embora tenha havido problemas com vazamento de memória, a maior dificuldade deste trabalho foi a execução dos experimentos. Foram codificadas várias configurações, testadas com vários tipos de agentes, pelo menos dez vezes cada uma em nove mapas diferentes.

6.3 TRABALHOS FUTUROS

Com os resultados deste trabalho, o grupo de pesquisa na área de Agentes Inteligentes e Jogos do Centro de Informática (CIn) [7] encerra os estudos preliminares

em patrulha que consideram que todos os nós tem a mesma prioridade, os grafos são estáticos e as populações homogêneas.

Há ainda diversas possibilidades abertas a respeito da pesquisa sobre patrulhamento multiagente baseado em negociação. Atualmente, há um estudo trabalhando em aumentar a complexidade dos agentes e do SMA, incluindo divisão do grafo por meio dos algoritmos da teoria dos grafos com técnicas de otimização para uma melhor formulação do problema.

Um estudo ainda não realizado é o teste de sistemas heterogêneos, combinando várias abordagens na mesma simulação.

Para testar adaptabilidade, as arestas do grafo deixariam de ser estáticas. Um outro trabalho interessante seria colocar obstáculos móveis no terreno. Os agentes teriam que desviar destes. O simulador onde os experimentos deste trabalho foram realizados não foi implementado de forma a permitir esta dinamicidade do grafo durante a simulação. Porém, em paralelo ao desenvolvimento deste trabalho, um novo simulador mais flexível foi desenvolvido, e um trabalho futuro é a implementação dos agentes desenvolvidos até então utilizando as estruturas deste novo simulador, e a realização destes testes de adaptabilidade.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Machado, A.P., Almeida, A.L., Ramalho, G., Zucker, J.-D. & Drougoul, A. (2002). "Multi-Agent Movement Coordination in Patrolling". *First Workshop on Agents in Computer Games*. The 3rd International Conference on Computers and Games (CG'02), Edmonton, Canadá.
- [2] Andrade, R.C., Macedo, H.T., Ramalho, G.L. & Ferraz, C.A.G. (2001). "Distributed Mobile Autonomous Agents in Network Management". *International Conference on Parallel and Distributed Processing Techniques and Applications*. Las Vegas, Nevada, USA.
- [3] Cho, J. & Garcia-Molina, H. (2000). "Synchronizing a database to Improve Freshness". *2000 ACM International Conference on Management of Data (SIGMOD)*. (May).
- [4] RoboCup (2001). RoboCup Rescue home page, disponível em: <http://www.r.cs.kobe-u.ac.jp/robocup-rescue/> (11/08/2003).
- [5] ARMYSTUDYGUIDE.COM (2005). Patrolling, disponível em: http://www.armystudyguide.com/content/powerpoint/Tactical_and_Survival_Presentations/patrolling-2.shtml (07/09/2006).
- [6] Machado, A., Ramalho, G., Zucker, J. & Drogoul, A. (2002). "Multi-Agent Patrolling: an Empirical Analysis of Alternative Architectures". *Multiagent Based Simulation*. Bolona, Itália.
- [7] Grupo de Inteligência Computacional, disponível em: <http://www.cin.ufpe.br/~compint/> (11/08/2003).
- [8] Almeida, A., Ramalho, G., Santana, H., Tedesco, P., Menezes, T., Corruble, V. & Chevaleyre, Y. (2004). "Recent Advances on Multiagent Patrolling". *The 17th Brazilian Symposium on Artificial Intelligence*. Springer Verlag, (pp. 474-483).
- [9] Chevaleyre, Y. & Sempé, F. (2004). "A theoretical analysis of multi-agent patrolling strategies". *Poster at the International Conference on Autonomous Agents and Multi-Agents Systems (AAMAS)*.
- [10] Sempé, F. & Drogoul, A. (2003). "Adaptive patrol for a group of robots". *IEEE International Conference on Robotics and Automation. (IROS 2003)*. Vol.3, (pp. 2865-2869).
- [11] Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C. & Wooldridge, M. (2001). "Automated Negotiation: Prospects, Methods and Challenges". *International Journal of Group Decision and Negotiation*. Vol.2, (pp. 199-215). Disponível em: <http://citeseer.nj.nec.com/jennings00automated.html> (16/08/2003).
- [12] Almeida, A. (2003). *Patrulhamento Multiagente em Grafos com Pesos*, Centro de Informática, Universidade Federal de Pernambuco - UFPE, Recife.
- [13] Santana, H., Ramalho, G., Corruble, V. & Ratitch, B. (2004). "Multi-agent Patrolling with Reinforcement Learning". *International Conference on Autonomous Agents and Multiagents Systems*.
- [14] Wooldridge, M.J. (2002). "An Introduction to MultiAgent Systems". John Wiley and Sons, 163-223.
- [15] Abate, F.R. (1996). *The Oxford Dictionary and Thesaurus: The Ultimate Language Reference for American Readers*, Oxford University Press.

- [16] LaValle, S., Lin, D., Guibas, L., Latombe, J. & Motwani, R. (1997). "Finding an unpredictable target in a workspace with obstacles". In *Proceedings of IEEE International Conference on Robotics and Automation*. (pp. 737-742). Disponível em: <http://citeseer.ist.psu.edu/lavalle97finding.html>
- [17] Firaxis (2005). Civilization IV, disponível em: <http://www.civ4.com/> (22/04).
- [18] Blizzard (2000). Diable II, disponível em: <http://www.blizzard.com/diablo2/> (22/04).
- [19] Russell, S.J. & Norvig, P. (1995). "Artificial Intelligence: A Modern Approach". *Prentice Hall Series in Artificial Intelligence*, Prentice Hall, Inc., 796-808.
- [20] O'Rourke, J. (1987). *Art Gallery Theorems and Algorithms*, Oxford University Press.
- [21] Satoh, I. (2003). "Configurable Network Processing for Mobile Agents on the Internet". *Cluster Computing*, Vol. 6. Disponível em: <http://citeseer.ist.psu.edu/satoh03configurable.html> (19/06/2006).
- [22] Arthur, W.B. (1994). "Inductive Reasoning and Bounded Rationality (The El Farol Problem)". *American Economic Review*, 84: 406-411.
- [23] Arkin, R.C. (1992). "Behavior-Based Robot Navigation for Extended Domains. Adaptive Behaviors". Vol. 1, 201-225.
- [24] Balch, T. & Arkin, R.C. (1998). "Behavior-Based Formation Control for Multi-robot Teams". *IEEE Transactions on Robot and Automation*. Vol.14, (pp. 926-939).
- [25] Reynolds, C.W. (1999). "Steering Behaviors for Autonomous Characters". *Game Developers Conference*. Disponível em: <http://www.red3d.com/cwr/steer/>
- [26] Howland, G. (1999). A Practical Guide to Building a Complete Game AI, disponível em: <http://www.gamedev.net/reference/articles/article785.asp> (19/06/2006).
- [27] Pottinger, D.C. (1999). "Coordinated Unit Movement". *Game Developer*, 42-51. January.
- [28] Pottinger, D.C. (1999). "Implementing Coordinated Unit Movement". *Game Developer*, 48-58. February.
- [29] Menezes, T. (2003). *Negociação em sistemas multiagente para patrulhamento*, Trabalho de Graduação, CIn, UFPE, Recife.
- [30] Stout, B.W. (1996). "Smart Moves: Intelligent Path-Finding". *Game Developer*. October/November, 28-35. October/November.
- [31] Sutton, R. & Barto, A. (1998). *Reinforcement Learning: An Introduction.*, Cambridge, MA.
- [32] Reinelt, G. (1994). "The Traveling Salesman: Computational Solutions for TSP Applications". *Lectures Notes in Computer Science* 840.
- [33] Parunak, H.V.D., Brueckner, S.A., J.Sauter & Posdamer, J. (2001). "Mechanisms and Military Applications for Synthetic Pheromones". In *Proceedings of Workshop on Autonomy Oriented Computation*.
- [34] Menezes, T., Tedesco, P. & Ramalho, G. (2006). "Negotiator Agents for the Patrolling Task". *The 18th Brazilian Symposium on Artificial Intelligence (SBIA)*. Springer-Verlag, (pp. 48-57), Ribeirão Preto - SP.
- [35] Huhns, M.N. & Stephens, L.M. (1999). "Multiagent Systems and Societies of Agents". In Weiss, G. (ed.), *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, Vol. 1, MIT Press, 79-114.
- [36] Faratin, P., Sierra, C. & Jennings, N. (2002). "Using Similarity Criteria to Make Issue Trade-Offs in Automated Negotiations". *Artificial Intelligence Journal*.

- [37] Faratin, P., Sierra, C. & Jennings, N. (1998). "Negotiation Decisions Functions for Autonomous Agents".
- [38] Raiffa, H. (1982). *The Art and Science of Negotiation*, Harvard University Press, Cambridge.
- [39] Wolfstetter, E. (1996). "Auctions - An Introduction". *Journal of Economic Surveys*, Vol. 10, 367-420.
- [40] Sotheby's Auctions, disponível em: <http://search.sothbys.com/> (20/03/2005).
- [41] Christie's, disponível em: http://www.christies.com/home_page/home_page.asp (20/03/2005).
- [42] OnSale, disponível em: <http://marketplace.onsale.com/> (20/03/2005).
- [43] eBay (2001). eBay - Your Personal Trading Community, disponível em: <http://www.ebay.com> (20/03/2005).
- [44] Yahoo! Auctions, disponível em: <http://sg.auctions.yahoo.com/> (20/03/2005).
- [45] Arremate.com, disponível em: <http://www.arremate.com.br/> (20/03/2005).
- [46] Wurman, P., Walsh, W. & Wellman, M. (1998). "Flexible double auctions for electronic commerce: Theory and implementation". *Decision Support Systems*. Vol.24, (pp. 17-27). Disponível em: citeseer.ist.psu.edu/article/wurman97flexible.html
- [47] Garcia, A.C.B., Lopes, A. & Bentes, C. (2001). "Electronic Auction with autonomous intelligent agents: Finding opportunities by being there". *Revista Iberoamericana de Inteligencia Artificial*. 13, 45-52. Disponível em: <http://aepia.dsic.upv.es/revista/numeros/13/bicharra.PDF> 21/06/2006.
- [48] Auction Patrol, disponível em: <http://www.auctionpatrol.com/> (20/03/2005).
- [49] MCFind, disponível em: <http://www.mcfind.com/auctions.html> (20/03/2005).
- [50] Wurman, P.R., Wellman, M.P. & Walsh, W.E. (1998). "The Michigan Internet AuctionBot: A Congurable Auction Server for Human and Software Agents". In Wooldridge, K.P.S.a.M. (ed.), *In proceedings of Second International Conference on Autonomous Agents*. ACM Press, (pp. 301-308), Minneapolis, MN, USA. Disponível em: citeseer.ist.psu.edu/wurman98michigan.html
- [51] Sandholm, T. (2000). "eMediator: A Next Generation Electronic Commerce Server". *International Conference on Autonomous Agents*. (pp. 341-348). Disponível em: citeseer.ist.psu.edu/sandholm02emediator.html
- [52] Sandholm, T. (2002). eMediator, disponível em: <http://www.cs.cmu.edu/~amem/eMediator/> (21/06/2006).
- [53] Wurman, P., Wellman, M. & Walsh, W. (2002). "Specifying Rules for Electronic Auctions". *AI Magazine*. 23(3), 15--24. Disponível em: <http://citeseer.ist.psu.edu/wurman02specifying.html>.
- [54] Lucking-Reiley, D. & Spulber, D.F. (2001). "Business-to-Business Electronic Commerce". *American Economic Association - Journal of Economic Perspectives*., Vol. 15(1), 55-68. Disponível em: file:///E:/%5CCIn%5CMestrado%5Cdissertacao%5CDis-versao-0.5.doc.
- [55] Reynolds, K. (1996). Auctions - Going, Going, Gone! A Survey of Auction Types, Agorics, Inc., disponível em: <http://www.agorics.com/Library/auctions.html> (21/06/2006).
- [56] Ben-Ameur, H., Chaib-draa, B. & Kropf, P. (2001). "Multiagent auctions for multiple items". *Third International Bi-Conference Workshop on Agent-Oriented Information Systems AOIS-2001*. Montreal. (28 Mai).

- [57] Kraus, S. (2001). "Automated Negotiation and Decision Making in Multiagent Environments". *ACAI-EASSS*.
- [58] Botelho, S. & Alami, R. (2000). "M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement". *IEEE International Conference on Robotics and Automation (ICRA)*. (pp. 293-298).
- [59] Gerkey, B.P. & M. J. Mataric (2002). "Sold!: Auction methods for multi-robot coordination". *IEEE Transactions on Robotics and Automation*. Vol.18, (pp. 758-768).
- [60] FIPA Communicative Act Library Specification, disponível em: www.fipa.org/specs/fipa00037/SC00037J.html (11/08/2003).
- [61] Manber, U. (1989). "Graph Algorithms". *Introduction to Algorithms - A Creative Approach*, Addison-Wesley, Tucson, 208 - 211.
- [62] Machado, A. (2002). *Patrulha Multiagente: uma análise empírica e sistemática*, Dissertação de Mestrado, Centro de Informática, Universidade Federal de Pernambuco, Recife.
- [63] Stroustrup, B. (1997). *C++ Programming Language*, Addison-Wesley.
- [64] OpenGL - The Industry's Foundation for High Performance Graphics, disponível em: <http://www.opengl.org/> (11/08/2003).
- [65] Menezes, T., Tedesco, P. & Ramalho, G. (2006). "Negotiator agents for the patrolling task". In Schiman, J., Coelho, H. & Rezende, S.O. (eds.), *International Joint Conference 2006, 10th Ibero-American Conference on AI, 18th Brazilian Symposium on AI (IBERAMIA/SBIA)*. Springer-Verlag, (pp. 23-27), Ribeirão Preto - SP, Brazil.



SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Ata de Defesa de Dissertação de Mestrado do
Centro de Informática da Universidade Federal de Pernambuco, 21 de agosto de 2006.

Ao vigésimo primeiro dia do mês de agosto do ano dois mil e seis, às catorze horas, no Centro de Informática da Universidade Federal de Pernambuco, teve início a quingentésima quadragésima sétima defesa de dissertação de Mestrado em Ciência da Computação intitulada “Negociação em Sistemas Multiagentes para Patrulhamento”, da candidata **Talita Rodrigues Menezes**, a qual já havia preenchido anteriormente as demais condições exigidas para a obtenção do grau de mestre. A Banca Examinadora, composta pelos professores Flávia de Almeida Barros, pertencente ao Centro de Informática desta Universidade, André Mauricio Cunha Campos, pertencente ao Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte e Patrícia Cabral de Azevedo Restelli Tedesco, pertencente ao Centro de Informática desta Universidade, sendo a primeira presidente da Banca Examinadora e a última orientadora do trabalho de dissertação, resolveu: **Aprovar por unanimidade e dar o prazo de trinta dias para a entrega da versão final do trabalho.** E para constar lavrei a presente ata que vai por mim assinada e pela Banca Examinadora. Recife, 21 de agosto de 2006.

Maria Lília Pinheiro de Freitas
Maria Lília Pinheiro de Freitas
(secretária)

Flávia de Almeida Barros
Profa. Flávia de Almeida Barros
(primeira examinadora)

André Mauricio Cunha Campos
Prof. André Mauricio Cunha Campos
(segundo examinador)

Patrícia Cabral de Azevedo Restelli Tedesco
Profa. Patrícia Cabral de Azevedo Restelli Tedesco
(terceira examinadora)