

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

DENISE DE OLIVEIRA

**Um Estudo de Coordenação Dinâmica de
Agentes Aplicado ao Gerenciamento de
Tráfego Veicular Urbano**

Dissertação apresentada como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

Prof^ª. Dra. Ana Lucia Cetertich Bazzan
Orientadora

Porto Alegre, abril de 2005

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Oliveira, Denise de

Um Estudo de Coordenação Dinâmica de Agentes Aplicado ao Gerenciamento de Tráfego Veicular Urbano / Denise de Oliveira. – Porto Alegre: PPGC da UFRGS, 2005.

76 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2005. Orientadora: Ana Lucia Cetertich Bazzan.

1. Sistemas multiagente. 2. Sincronização de Semáforos. 3. Coordenação. 4. Swarm Intelligence. 5. Problema de Otimização de Restrições. I. Bazzan, Ana Lucia Cetertich. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. José Carlos Ferraz Hennemann

Vice-Reitor: Prof. Pedro Cezar Dutra Fonseca

Pró-Reitora de Pós-Graduação: Prof^a. Valquíria Linck Bassani

Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux

Coordenador do PPGC: Prof. Flávio Rech Wagner

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	5
LISTA DE FIGURAS	6
LISTA DE TABELAS	7
LISTA DE ALGORITMOS	8
RESUMO	9
ABSTRACT	10
1 INTRODUÇÃO	11
2 COORDENAÇÃO DE SEMÁFOROS	13
2.1 Introdução	13
2.1.1 Definições gerais	14
2.2 Princípios de Sincronização	14
2.3 Sistemas de Controle de Tráfego	15
2.3.1 TRANSYT	16
2.3.2 SCOOT	16
2.3.3 SCATS	18
2.4 Conclusão	18
3 SWARM INTELLIGENCE	19
3.1 Características dos insetos sociais	20
3.2 Organização das sociedades de insetos	21
3.2.1 Especialização das castas operárias	21
3.2.2 Comunicação	22
3.3 Modelos Teóricos de Divisão de Trabalho em Insetos Sociais	23
3.3.1 Mecanismos de regulação da divisão de trabalho	24
3.4 Conclusão	25
4 OTIMIZAÇÃO DE RESTRIÇÕES	26
4.1 Introdução	26
4.2 Resolução de PSR via Busca com Retrocesso	27
4.2.1 <i>Branch and Bound</i>	28
4.3 Mediação Cooperativa	29
4.3.1 Inicialização	30
4.3.2 Verificação da visão do agente	30

4.3.3	Mediação	31
4.4	Conclusão	32
5	ABORDAGENS PROPOSTAS DE COORDENAÇÃO DE SEMÁFOROS	33
5.1	Modelo Utilizando Insetos Sociais	33
5.1.1	Descrição	33
5.1.2	Simulador Microscópico de Tráfego	33
5.1.3	Cálculo do Estímulo	34
5.1.4	Alocação de Planos	36
5.1.5	Reforço	37
5.1.6	Cenário e Experimentos	38
5.1.7	Resultados	40
5.1.8	Conclusões	42
5.2	Coordenação Distribuída de Semáforos utilizando Mediação Cooperativa	42
5.2.1	Descrição	42
5.2.2	Exemplo	45
5.2.3	Cenário e Experimentos	47
5.2.4	Resultados	48
5.2.5	Conclusões	54
5.3	Considerações sobre as abordagens	54
6	CONCLUSÃO E TRABALHOS FUTUROS	55
	REFERÊNCIAS	56
APÊNDICE A	PLANOS SEMAFÓRICOS DO CENÁRIO UTILIZADO NO MODELO DE INSETOS SOCIAIS	60
APÊNDICE B	ALOCAÇÃO DINÂMICA DE TAREFAS INSPIRADA EM INSETOS SOCIAIS	68
B.1	Introdução	68
B.2	TÆMS, GPGP e DTC	69
B.3	O modelo de alocação de tarefas inspirado em insetos sociais	69
B.3.1	Estímulo	69
B.3.2	Polietismo	70
B.4	Experimentos	71
B.4.1	Cenário I	72
B.4.2	Cenário II	73
B.4.3	Cenário III	74
B.5	Conclusões	75

LISTA DE ABREVIATURAS E SIGLAS

Av.	Avenida
COP	<i>Constraint Optimization Problem</i>
CSP	<i>Constraint Satisfaction Problem</i>
ITSUMO	<i>Intelligent Transportation System for Urban Mobility</i>
LO	Leste–Oeste
MEF	Máquina de Estados Finitos
NS	Norte–Sul
OL	Oeste–Leste
OptApo	<i>Optimal Asynchronous Partial Overlay</i>
PCSP	<i>Partial Constraint Satisfaction Problem</i>
PCF	Padrão Cíclico de Fluxo
PSR	Problema de Satisfação de Restrições
SCATS	<i>Sydney Coordinated Adaptive Traffic System</i>
SCOOT	<i>Split Cycle and Offset Optimization Technique</i>
SCTU	Sistema de Controle de Tráfego Urbano
SeSAm	<i>Shell for Simulated Agent Systems</i>
SMA	Sistema Multiagente
SN	Sul–Norte
TRANSYT	<i>Traffic Network Study Tool</i>
XML	<i>Extensible Markup Language</i>

LISTA DE FIGURAS

Figura 2.1:	Exemplo de <i>onda verde</i> causando congestionamento nas vias secundárias.	13
Figura 2.2:	Diagrama tempo-espaço de sincronização.	15
Figura 2.3:	Previsão de Filas	17
Figura 4.1:	Mapa a ser colorido e a direita o grafo de restrições que o representa.	26
Figura 4.2:	Parte da árvore de busca para a coloração do mapa da Figura 4.1.	27
Figura 4.3:	Parte da árvore de busca por <i>Branch and Bound</i>	29
Figura 5.1:	Rede analisada vista com o Visualizador 3D	35
Figura 5.2:	À direita, gráfico que mostra o decaimento linear do grau de sucesso e, à esquerda, o decaimento exponencial	38
Figura 5.3:	Sub-rede analisada com a localização dos semáforos	38
Figura 5.4:	Planos semafóricos básicos (a linha em preto representa o tempo de verde)	39
Figura 5.5:	Uma rede com 25 intersecções (os nós com círculos pontilhados executam o plano SN/NS e os com linha cheia executam o plano LO/OL).	43
Figura 5.6:	Uma configuração particular para o subsistema do nó B4.	45
Figura 5.7:	Movimentos de tráfego possíveis nos nodos com 8 faixas.	47
Figura 5.8:	Movimentos de tráfego das fases dos semáforos neste cenário.	48
Figura 5.9:	Cenário com volume de tráfego constante: custo ao longo da simulação.	49
Figura 5.10:	Cenário com volume de tráfego constante: número de mediações por ciclo.	49
Figura 5.11:	Grupos formados após a mediação.	50
Figura 5.12:	Mudança na inserção de veículos: Caso II (esquerda) e Caso III (direita).	50
Figura 5.13:	Situação com uma mudança no volume de tráfego no passo de simulação 400: custo ao longo da simulação.	51
Figura 5.14:	Situação com uma mudança no volume de tráfego no passo de simulação 400: número de mediações por ciclo.	51
Figura 5.15:	Situação com duas mudanças no volume de tráfego (nos passos 400 e 700): custo ao longo da simulação.	52
Figura 5.16:	Situação com duas mudanças no volume de tráfego (nos passos 400 e 700): número de mediações por ciclo.	53
Figura 5.17:	Caso com coordenação fixa: as setas em negrito indicam mudança na inserção de veículos para a variante 1 (esquerda) e variante 2 (direita).	53
Figura 5.18:	Caso com coordenação fixa.	54

LISTA DE TABELAS

Tabela 2.1:	Estratégias de Controle de Semáforos	16
Tabela 3.1:	Diferentes níveis de organização social encontrados entre os insetos. .	20
Tabela 3.2:	Diferentes níveis de organização subsocial encontrados entre os insetos.	21
Tabela 5.1:	Tempo médio para completar o percurso em segundos	41
Tabela 5.2:	Tabela de Preferências	47

LISTA DE ALGORITMOS

4.1	<i>Branch and Bound</i>	29
4.2	OptApo: Escolhe Solução	31
5.1	Calcula o custo global F_i	43
5.2	Calcula a relação de custo $f(x_i, x_j)$	44

RESUMO

Em sistemas de controle de tráfego veicular urbano existem diversas abordagens que lidam com o problema da redução do congestionamento de tráfego. Uma classe destas abordagens aplica a sincronização de semáforos a fim de permitir que os veículos que viajam em um sentido possam passar diversos cruzamentos sem paradas.

Este trabalho apresenta dois métodos de coordenação de agentes aplicados à sincronização de semáforos. O primeiro método inspira-se em mecanismos de coordenação observados em insetos sociais e o segundo modela o problema de coordenação de semáforos como um problema de otimização de restrições distribuído e faz sua resolução em tempo real utilizando mediação cooperativa.

Inicialmente são apresentados conceitos básicos de sistemas de tráfego urbano, *Swarm Intelligence* e problemas de otimização de restrições. A partir dos conceitos iniciais, são apresentados os modelos propostos.

Os resultados mostram que as abordagens propostas geram a coordenação entre os semáforos sendo que o modo que os agentes estão coordenados pode mudar para se adaptar às mudanças nas condições do ambiente, gerando melhores condições de fluxo de tráfego.

Palavras-chave: Sistemas multiagente, Sincronização de Semáforos, Coordenação, Swarm Intelligence, Problema de Otimização de Restrições.

Study of dynamic agents coordination applied to urban traffic control

ABSTRACT

In urban traffic control systems, there are several approaches to tackle the problem of reducing traffic jams. A class of these approaches deals with synchronization of traffic lights to allow vehicles traveling in a given direction to pass through several junctions without stopping.

This paper presents two methods of dynamic coordination of agents applied to traffic lights synchronization. The first method is inspired in coordination mechanisms observed in social insects and the second models the coordination problem as a distributed constraint optimization problem and solves it in real time using a cooperative mediation method.

Initially, basic concepts about urban traffic control, Swarm Intelligence, and constraint optimization problems are presented. Using these initial concepts, the proposed models are presented.

The results show that proposed approaches generates coordination among the traffic lights so that the way the agents are coordinated can change to adapt to the modifications in the environment conditions, leading to a better traffic flow.

Keywords: multiagent systems, urban traffic control, coordination, Swarm Intelligence, Constraint Optimization Problems.

1 INTRODUÇÃO

Propostas para a redução do congestionamento das vias em grandes cidades vêm sendo desenvolvidas por diferentes disciplinas, como engenharia de tráfego, inteligência artificial, física entre outras. Uma abordagem clássica utiliza a sincronização dos semáforos de modo que os veículos possam atravessar uma via arterial em uma direção com uma velocidade específica, sem paradas (TRANSYT-7F, 1988). A coordenação dos semáforos em uma “onda verde”, significa que todos os semáforos adjacentes executam planos de temporização semafóricos que permitem aos veículos percorrerem uma via sem paradas.

A coordenação é um ponto chave dos Sistemas Multiagente (SMAs) em geral porque os agentes (entidades autônomas) neste tipo de sistema, geralmente, não tem percepção total do ambiente e devem executar tarefas em conjunto com outros agentes para atingir um objetivo global. A coordenação, em SMA, é definida como a busca de dois ou mais agentes por um mesmo objetivo global (cooperação) na qual as ações de um indivíduo são relacionadas com as ações de um ou mais agentes dentro de um grupo de cooperação. A coordenação também pode ser definida, de modo mais geral, (MALONE; CROWSTON, 1994): “Coordenação é o ato de trabalhar em conjunto.”

A coordenação no caso dos semáforos é, no caso dos sistemas centralizados, controlada por um sistema central ou pré-programada. Nos SMAs existem diversos mecanismos de coordenação, sendo que neste trabalho serão apresentadas a coordenação implícita (através de regras de conduta individuais) e por negociação utilizando comunicação.

A abordagem tradicional de coordenação de semáforos funciona em ambientes cujo fluxo do tráfego seja bem definido ao longo do dia. As grandes cidades não têm um fluxo de tráfego bem definido visto que os centros de comércio e negócios estão localizados em diferentes pontos da cidade. Além disso, o fluxo do tráfego também é altamente afetado por acidentes, causas naturais (como enchentes ou nevascas), obras nas vias, etc. Vias secundárias podem ter fluxo maior de veículos que vias arteriais em alguns momentos. Esses motivos nos levam a perceber que a coordenação semafórica buscando “onda verde” apenas em vias principais não é suficiente, visto que diversas rotas com alto fluxo se interceptam e têm grande variação em seus fluxos ao longo do dia. A determinação em tempo real de qual plano é mais apropriado para cada semáforo a fim de manter a rede com melhores condições de mobilidade é uma tarefa complexa que requer contagens e estudos do fluxo de tráfego em diferentes pontos da rede viária. Além disto, os programas disponíveis para otimização de vias são comerciais e podem ter custo elevado. Abordagens mais flexíveis, robustas e não comerciais são uma necessidade.

A determinação dos planos mais apropriados para cada hora do dia é uma tarefa complexa que requer contagens e estudos do fluxo de tráfego em diferentes pontos da rede de vias. SMAs oferecem soluções mais flexíveis e robustas. Em (BAZZAN, 1997, 2005) uma abordagem baseada em sistemas multiagente é descrita, onde cada semáforo é mo-

delado como um agente. Cada agente possui planos pré-definidos para coordenação com agentes adjacentes. Planos diferentes podem ser escolhidos para haver coordenação em diferentes sentidos de acordo com a hora do dia. Essa abordagem utiliza técnicas de teoria dos jogos evolucionária.

Os principais benefícios dessa abordagem são: os agentes podem criar subgrupos de sincronização para melhor atender às necessidades do fluxo em alguma direção, não há necessidade de um controle central e não há comunicação nem negociação direta entre os agentes. No entanto, são necessárias matrizes de pagamento (*pay-off matrices*) e essas matrizes devem ser formalizadas explicitamente pelo projetista do sistema. Isto faz com que a abordagem consuma tempo quando diferentes opções de coordenação são possíveis e/ou a rede de tráfego é muito complexa (não há somente uma via principal a ser considerada).

Com o objetivo de satisfazer tais necessidades, esse trabalho apresenta duas abordagens de coordenação de semáforos: na primeira, cada semáforo comporta-se como um inseto social que baseia o seu processo decisório no processo de escolha de tarefas encontrado em insetos sociais; na segunda, a formação de grupos ocorre a partir da mediação colaborativa entre agentes.

O texto deste trabalho encontra-se organizado da seguinte forma:

O Capítulo 2 faz uma breve introdução sobre conceitos básicos de coordenação sema-fórica e apresenta quatro diferentes abordagens de coordenação existentes. O Capítulo 3 apresenta conceitos de inteligência coletiva encontrada em insetos sociais (*Swarm Intelligence*) e como esse método de organização pode ser utilizado. No Capítulo 4, os problemas de satisfação e otimização de restrições são apresentados, mostrando-se como modelar problemas diferentes com este enfoque, e também algumas abordagens existentes para sua solução. As abordagens propostas são apresentadas no Capítulo 5, juntamente com a análise de seus resultados. As conclusões e trabalhos futuros, encontram-se no Capítulo 6.

2 COORDENAÇÃO DE SEMÁFOROS

2.1 Introdução

A partir da década de 60 do século XX, Sistemas de Controle de Tráfego Urbano(SCTUs) vêm sendo desenvolvidos tendo como principais objetivos:

- maximizar a capacidade de fluxo das redes urbanas;
- aumentar a segurança do tráfego;
- diminuir os tempos de percurso;
- diminuir os impactos negativos do tráfego no ambiente e no consumo de energia.

Em redes de tráfego onde não há um padrão bem definido de fluxo, por exemplo picos da manhã e fim de tarde, um sistema de controle de tráfego tradicional pode não ser eficaz. Nas grandes cidades os sistemas de tráfego que consideram fluxos bem definidos não podem lidar com fluxos variáveis onde não há uma concentração somente na zona central da cidade, devido à dispersão dos centros de negócios ao longo da rede. Além disso, em algumas cidades, algumas vias consideradas secundárias tornaram-se também vias importantes devido à saturação das vias arteriais. Atualmente, as abordagens visam a otimização de rotas (geralmente rotas principais) e a rede como um todo nem sempre obtém uma melhora significativa na qualidade do tráfego.

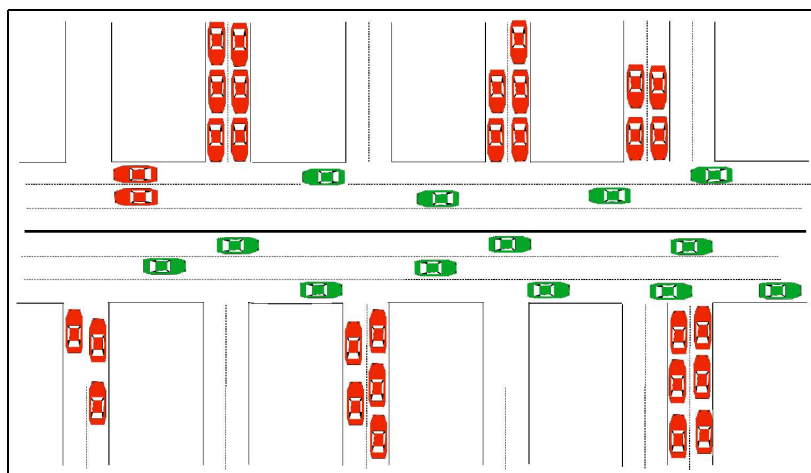


Figura 2.1: Exemplo de *onda verde* causando congestionamento nas vias secundárias.

A Figura 2.1, onde os veículos em vermelho estão parados e os verdes estão em movimento, mostra um exemplo de onda verde ineficiente. A sincronização fixa, no momento mostrado na figura, está priorizando uma avenida arterial cujo fluxo de tráfego é menos intenso que as vias secundárias, gerando diversos engarrafamentos nas transversais à arterial. Os sistemas tradicionais que utilizam o conceito de otimização de rotas tendem a gerar este tipo de onda verde. Abordagens adaptativas permitem a criação de ondas verdes mais curtas que estejam mais de acordo com o fluxo de tráfego da rede e não apenas de uma rota.

2.1.1 Definições gerais

Existem alguns conceitos básicos que devem ser apresentados para a melhor compreensão do funcionamento dos semáforos:

- Semáforo: pode ser classificado, (LEITE, 1980), como sendo toda a instalação de controle existente em um cruzamento, incluindo os sinais luminosos, os fios, os instrumentos de controle, etc;
- Tempo de ciclo: é o número de segundos para uma seqüência completa das indicações do sinal, sendo que pode ter um tempo variável ou fixo;
- Tempo de fase (*Split*): um intervalo de tempo de ciclo (tempo relativo de verde) reservado para qualquer movimento de tráfego (estágio);
- Plano semaforico: um plano é um conjunto único de fases;
- Defasagem (*offset*): tempo necessário para percorrer o espaço entre dois semáforos adjacentes, que estejam sincronizados, a uma velocidade média.

2.2 Princípios de Sincronização

Existem quatro possibilidades de influenciar as condições do tráfego utilizando semáforos (de acordo com Diakaki, (DIAKAKI; PAPAGEORGIOU; ABOUDOLAS, 2002)):

- Especificação do estágio: para cruzamentos complexos, a especificação do número ótimo de estágios pode ter impacto na eficiência do cruzamento;
- Tempo de fase: o tempo de verde de cada estágio deve ser dimensionado de acordo com a demanda das faixas envolvidas;
- Tempo de ciclo: tempos de ciclo maiores geralmente aumentam a capacidade do cruzamento, por outro lado, tempos de ciclo maiores aumentam os tempos de espera em cruzamentos sub-saturados;
- Defasagem: a especificação da defasagem ótima deve levar em conta a existência de possíveis filas e a velocidade média dos veículos.

O critério para obter a temporização ótima é que esta deve levar a um tempo de espera mínimo. Isto geralmente é alcançado utilizando-se programas de otimização. Geralmente, cada semáforo tem diversos planos semaforicos, sendo que cada um é adequado para o fluxo de tráfego em determinada hora do dia ou condição de tráfego.

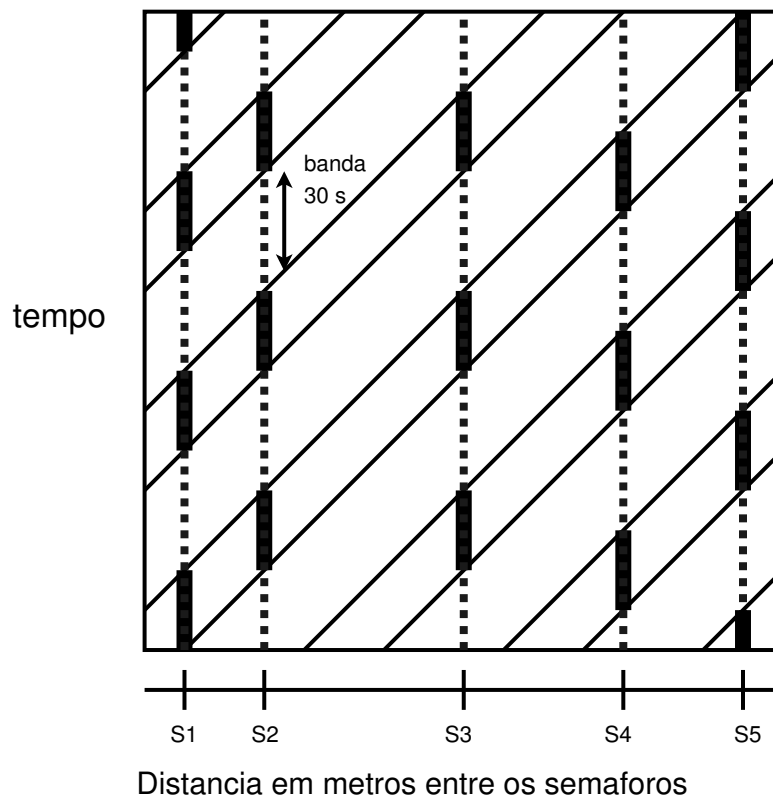


Figura 2.2: Diagrama tempo-espaço de sincronização.

A sincronização é atingida quando dois ou mais semáforos estão executando os mesmos tipos de planos semafóricos de modo que permita um veículo passar pelos semáforos sincronizados sem paradas. Para que a sincronização seja possível, é necessário que haja uma defasagem *offset* relativa ao semáforo anterior. Por exemplo, a Figura 2.2 mostra um sistema de sincronização onde a velocidade média é de 45km/h (12,5 m/s) e o tempo de banda é de 30 segundos. Considerando que do semáforo “S1” ao semáforo “S2” há uma distância de 50 metros, a defasagem do semáforo “S2” em relação ao “S1” seria de 4s (tempo necessário para percorrer 50 metros). Há uma propagação da defasagem até o último semáforo da onda verde (“S5”), que está a 350 metros de “S1”, fazendo com que um veículo saindo de “S1” e mantendo uma velocidade constante de 45Km/h atravessasse todos os semáforos sem paradas.

O problema clássico na sincronização de semáforos é encontrar o tamanho de banda maior (ótimo) para diferentes tempos de ciclo e velocidades. Bons planos semafóricos coordenados são muito trabalhosos de serem criados. De acordo com (ROBERTSON; BRETHERTON, 1991), uma rede com 30 a 40 semáforos demanda um ano de trabalho-homem.

2.3 Sistemas de Controle de Tráfego

Existem basicamente dois tipos de temporização em sistemas de controle de tráfego urbano: fixo ou adaptativo. A sincronização com temporização fixa é a mais amplamente utilizada e a mais simples, já que não requer a instalação de detectores de tráfego. Os detectores são circuitos eletrônicos digitais que quando conectados a uma bobina, laço detector ou *loop*, compõem um conjunto capaz de acusar (detectar), eletricamente, a pre-

Tabela 2.1: Estratégias de Controle de Semáforos

Abordagem	Estrutura	Custo	Desempenho
Sincronização com temporização fixa	Não computadorizada	Baixo	Baixo a médio
	Computadorizada e Centralizada	Médio	
Sincronização em resposta ao tráfego com seleção ou modificação de planos semaforicos	Computadorizada e Centralizada	Médio a alto	Médio a alto
Sincronização com controle completamente distribuído	Distribuída	Médio a alto	Alto

sença de uma massa metálica (veículo) no campo de influência da antena. A temporização que percebe o fluxo de tráfego e adapta-se a ele, exige a instalação de detectores e de um sistema de controle computadorizado mais complexo. O meio termo entre esses dois tipos de sistema também é utilizado. A Tabela 2.1 resume as estratégias de sincronização, mostrando as diferenças na estrutura necessária, custo e desempenho de *hardware* de cada uma delas.

A seguir, serão apresentados brevemente alguns dos sistemas mais utilizados comercialmente.

2.3.1 TRANSYT

O *Traffic Network Study Tool* (TRANSYT) é um dos programas de temporização de tempos de fase mais amplamente utilizados e mais antigos. É executado de maneira *off-line* para determinar o tempo fixo ótimo de semáforos coordenados em qualquer rede de vias na qual o fluxo médio de veículos seja conhecido, utilizando modelos macroscópicos e determinísticos de simulação. Embora o TRANSYT seja geralmente utilizado como ferramenta de otimização *off-line*, pode ser utilizado de um modo *on-line* fazendo-se a atualização dos dados da rede em pequenos intervalos de tempo e realimentando a rede com os resultados obtidos no simulador. O modelo a ser simulado deve ter entradas de veículos e probabilidades de mudança de direção constantes ao longo de todo o período de simulação.

Os critérios de otimização utilizados são: tamanho da fila, maximização do tamanho da “banda” da onda verde e quantidade de paradas. O programa otimiza as fases e as defasagens relativas dado um conjunto de tempos de ciclo realizando diversas interações entre o módulo de simulação de tráfego e o módulo de otimização de semáforos. Os resultados são calculados em relação a uma rota específica na rede definida pelo usuário.

2.3.2 SCOOT

O SCOOT (*Split Cycle and Offset Optimization Technique*) (ROBERTSON; BRETHERTON, 1991) é um modelo centralizado de controle de tráfego desenvolvido pelo *Transportation Road Research Laboratory* no Reino Unido. O SCOOT usa detectores instalados nas vias para medir perfis do fluxo de tráfego em tempo real e, juntamente com tempos de percurso e graus de saturação (ocupação relativa à capacidade nominal da via) pré-determinados, prediz filas em intersecções, como mostra a Figura 2.3. O sistema de previsão de filas se baseia no uso de “Padrões Cíclicos de Fluxo” (PCFs). Um PCF é o fluxo de veículos médio em uma direção em qualquer ponto da via durante o tempo

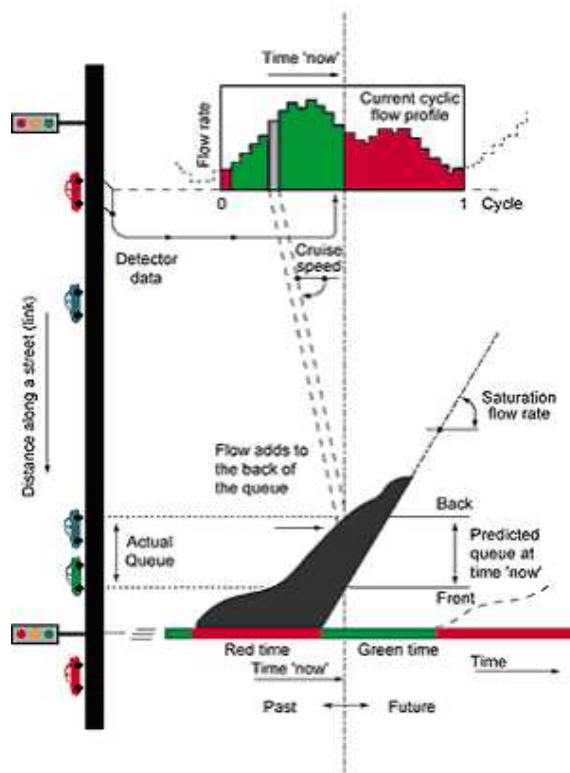


Figura 2.3: Previsão de Filas¹

de verde, sendo então um platô de veículos como passos sucessivos dentro do ciclo do semáforo. De acordo com (ROBERTSON; BRETHERTON, 1991), o sistema utiliza os mesmos critérios de otimização utilizados pelo TRANSYT.

Quando um veículo passa pelo detector, o SCOOT converte a informação em um tipo híbrido composto por informações de fluxo e ocupação. A partir das informações coletadas, o sistema central envia instruções para os equipamentos localizados nos semáforos. Essas instruções são interpretadas e verificadas pelo semáforo. Caso as instruções sejam realizáveis, ele envia uma mensagem de aceitação, caso contrário, envia uma mensagem indicando uma falha na instrução.

SCOOT tem três procedimentos de otimização para sincronizar os semáforos: o de tempos de fase, o de defasagem e o do tempo de ciclo. Cada procedimento otimizador estima o efeito de uma mudança incremental pequena dos tempos no desempenho total da rede de tráfego da região. Um índice de desempenho é calculado baseado em previsões sobre paradas e velocidade dos veículos.

Os ajustes do tempo de ciclo, tempos de fase e as defasagens são feitos em conjunto para operar num grau de saturação pré-definido (geralmente 90%). Os testes mostraram que SCOOT é mais eficaz quando a demanda se aproxima da capacidade, onde a demanda é imprevisível, e quando as distâncias entre intersecções são curtas (ROBERTSON; BRETHERTON, 1991).

¹Figura obtida nos documentos disponíveis no site: http://www.dft.gov.uk/stellent/groups/dft_roads/documents/page/dft_roads_504749-05.hcsp

2.3.3 SCATS

O SCATS (*Sydney Coordinated Adaptive Traffic System*), (LOWRIE, 1982), foi inicialmente desenvolvido na Austrália para aplicação em Sydney e em outras cidades australianas. Atualmente está instalado em mais de 50 cidades no mundo. É um sistema dinâmico de controle de semáforos com uma arquitetura descentralizada. A otimização do sistema se dá através de mudanças no tamanho do ciclo da fase e tempo de defasagem, além disso, permite também que algumas fases não sejam executadas. O sistema possui um biblioteca de planos que podem ser selecionados.

2.4 Conclusão

Abordagens centralizadas de controle de tráfego não podem lidar com a complexidade crescente das redes de tráfego urbano. Os sistemas adaptativos geralmente não permitem mudanças na topologia da coordenação dos semáforos, por exemplo, a troca de direção de coordenação muitas vezes não é possível.

As questões apresentadas mostram que uma otimização, mesmo com atualização em tempo real, que vise apenas à sincronização de uma rota não pode lidar com mudanças nos padrões de tráfego de toda a rede. Isso acontece já que o movimento de veículos é um processo altamente dinâmico onde o plano para o estado corrente, raramente pode ser determinado previamente. Abordagens flexíveis e robustas, embora apresentem elevado custo inicial da instalação de equipamentos de sensoriamento, mostram-se necessárias.

Nos próximos capítulos serão apresentados alguns modelos de coordenação e de divisão de tarefas que serão utilizados nas abordagens de coordenação de semáforo adaptativas propostas neste trabalho.

3 SWARM INTELLIGENCE²

Como visto no Capítulo 2, a formação de grupos de semáforos que se coordenam para melhorar o fluxo de tráfego de uma via é uma tarefa complexa e que deve ser adaptada às condições da via a cada momento. Analisando o comportamento dos insetos sociais vemos que um dos principais fatores responsáveis pelo seu sucesso ecológico é a sua capacidade de alocar dinamicamente as tarefas entre os membros da colônia. Essa divisão de trabalho adaptativa é uma abordagem interessante que pode ser aplicada na divisão e ordenamento de tarefas em Sistemas Multiagentes (SMA) (OLIVEIRA; FERREIRA; BAZZAN, 2004a), apresentado no Apêndice B e aplicada também à coordenação de semáforos, que será vista no modelo proposto na Seção 5.1 do Capítulo 5.

Por se tratarem de entidades biológicas dinâmicas, as colônias sofrem mudanças constantes durante sua vida em relação ao seu tamanho, à quantidade de comida disponível, à época do ano, à ação de predadores e as condições do clima.

As colônias respondem a estas mudanças de condições externas e internas ajustando a quantidade de operários envolvidos em cada tarefa. Esta flexibilidade no comportamento dos operários contribui decisivamente para o sucesso reprodutivo da colônia, permitindo o crescimento, desenvolvimento e o nascimento de novas gerações reprodutivas independente da variação nas condições da colônia.

A pesquisa relacionada aos mecanismos que regulam a divisão de trabalho nas sociedades de insetos tem reportado avanços no entendimento deste processo e modelos heurísticos foram propostos baseados em comportamento orientado a níveis de resposta a estímulos (ROBISON, 1992; GORDON, 1996; GORDON; PEREIRA, 2001; BONABEAU et al., 1997; THERAULAZ; BONABEAU; DENEUBOURG, 1998).

Este capítulo tem como objetivo fazer uma breve introdução sobre *Swarm Intelligence* que servirá como base teórica para a compreensão do modelo de coordenação de semáforos, proposto no Capítulo 5, Seção 5.1. Primeiramente serão apresentados conceitos básicos sobre insetos sociais seguido de alguns modelos teóricos. Os insetos sociais possuem uma das estratégias de sobrevivência mais bem sucedidas da natureza, por isso, há uma imensa quantidade e variedade destes. Segundo Wilson (WILSON, 2000), existem mais espécies de formigas em 1 quilômetro quadrado de uma floresta brasileira do que todas as espécies de primatas existentes no mundo, e uma simples colônia de formigas possui mais habitantes do que todos os elefantes e leões da África somados.

A organização social desses insetos se apresenta aos biólogos como um tópico instigante de estudo e comparação. Entretanto, a riqueza e diversidade dos insetos sociais são tão grandes que, apesar do grande estudo sobre estas criaturas, ainda há muitas espécies

²Neste trabalho, é utilizado o termo *swarm intelligence* por não haver consenso em sua tradução para o português, sendo encontrado como diversos termos diferentes, tais como: inteligência coletiva, inteligência de enxames, inteligência de insetos, etc.

Tabela 3.1: Diferentes níveis de organização social encontrados entre os insetos.

Nível da sociedade	Cuidado cooperativo da prole	Castas reprodutivas	Sobreposição de gerações
Solitário	-	-	-
Comunal	-	-	-
Quasisocial	+	-	-
Semi-social	+	+	-
Social	+	+	+

pouco investigadas ou completamente desconhecidas. Existem cerca de 8.000 espécies de formigas descritas atualmente, mas acredita-se que ainda haja mais 4.000 espécies desconhecidas. Entretanto, destas 12.000 espécies de formigas, apenas 1% foi estudada com algum detalhamento por pesquisadores.

3.1 Características dos insetos sociais

Segundo entomologistas, para que um inseto seja considerado social ele deve possuir três qualidades fundamentais:

- Indivíduos da mesma espécie cooperam nos cuidados dos mais jovens;
- Divisão reprodutiva do trabalho, com indivíduos mais ou menos estéreis;
- Sobreposição de pelo menos duas gerações no trabalho da colônia.

Qualquer inseto que não possua as três características poderá ser classificado como pré-social. O termo pré-social é usado para caracterizar qualquer espécie com um comportamento social maior que apenas o sexual, mas ainda assim, menor que o dos insetos sociais. A tabela 3.1 apresenta uma série de estágios intermediários dentro do espectro definido pelas características dos insetos sociais em uma seqüência parasocial, que avalia o grau de assistência mútua entre adultos de uma mesma geração. Os solitários não apresentam nenhuma das características fundamentais dos insetos sociais; os comunais se juntam para criar um ninho, mas cada fêmea cria sua prole separadamente; no caso dos quasisociais a prole é criada junto, mas as fêmeas ainda geram seus próprios filhotes; nos semi-sociais já existem castas que geram a prole enquanto outros trabalham.

Uma seqüência alternativa é a que considera o nível subsocial dos insetos, que avalia o grau de associação entre a mãe e a prole, conforme mostra a tabela 3.2. Alguns pesquisadores acreditam que esta rota tenha sido seguida por formigas, cupins, vespas e alguns poucos grupos de abelhas.

Contudo, é difícil encontrar insetos nos níveis sociais intermediários, pois normalmente essas sociedades primitivas são superadas por seus primos mais evoluídos socialmente. Wilson afirma, em (WILSON, 2000), que apenas as abelhas e vespas possuem espécies em níveis pré-sociais de organização que competem com espécies sociais.

Um grupo de insetos sociais pode ser imaginado como um organismo difuso. E podem se apresentar em diversos tamanhos diferentes, por exemplo, uma típica colônia de *Tetramorium caespitum* tem cerca de 10.000 formigas, 6,5 gramas e controla 40m² de solo enquanto uma colônia de *Dorylus wilverthi* é formada por 22.000.000 de formigas, pesa mais de 20 kg e controla até 50.000 m² de extensão.

Tabela 3.2: Diferentes níveis de organização subsocial encontrados entre os insetos.

Nível da sociedade	Cuidado cooperativo da prole	Castas reprodutivas	Sobreposição de gerações
Solitário	-	-	-
Sub-social Primitivo	-	-	-
Subsocial Intermediário I	-	-	+
Subsocial Intermediário II	+	-	+
Social	+	+	+

3.2 Organização das sociedades de insetos

De acordo com Wilson (WILSON, 2000), assim que uma espécie chega até o nível “social” a organização da colônia pode avançar em dois sentidos: no aumento do grau de especialização das castas operárias, de forma física e/ou temporal – o polimorfismo e o polietismo temporal, respectivamente – ou no alargamento do código de comunicação por onde os membros da colônia coordenam suas atividades. Com relação ao grau de especialização das classes operárias, Robinson (ROBISON, 1992) acrescenta ainda uma terceira característica: as diferenças individuais na especialização das classes.

3.2.1 Especialização das castas operárias

As castas podem ser de dois tipos: físicas e temporais. As formigas fêmeas se dividem em três castas: operárias, soldados e rainha. Já os machos não se dividem em castas, mas podem, num sentido mais abstrato, serem considerados como uma quarta casta. Em poucas espécies de formiga existe uma clara definição das três castas de fêmeas, em muitos casos não existem formigas soldado, por isso, na maioria das espécies, as mudanças fisiológicas e de comportamento mudam com a idade do indivíduo. Essas mudanças representam saltos de uma casta temporal para outra.

De acordo com Gordon (GORDON, 1996), operárias jovens primeiro desempenham tarefas dentro do ninho, como armazenar alimento e cuidar da prole, enquanto operárias mais velhas desempenham tarefas externas, como forragear. As formigas que cuidam da manutenção do formigueiro estão em transição entre o interior e o exterior, porque esse trabalho é feito em parte dentro do formigueiro, quando as formigas amontoam areia proveniente de túneis escavados, e em parte no exterior, quando a areia é levada para fora da entrada do formigueiro. Até as larvas podem ser consideradas uma sub-casta temporal, em muitas espécies as larvas possuem uma glândula capaz de liberar um material líquido rico em nutrientes que é muito importante na “economia” da colônia.

Os cupim apresentam uma organização similar à das formigas, mas as castas se apresentam em ambos os sexos. Um aspecto interessante dos cupins é que enquanto as larvas de formigas são praticamente inúteis, as dos cupins colaboram no trabalho da colônia. Os cupins também possuem “reprodutores suplementares” e não apenas a rainha e o macho. Os reprodutores suplementares tornam as colônias praticamente imortais, pois se o reprodutor principal é removido, os outros podem substituí-lo.

Com abelhas e vespas, a divisão de castas é definida totalmente pelo polietismo temporal. Isto ocorre, segundo Wilson (WILSON, 2000) por que abelhas e vespas tomaram um caminho evolucionário diferente, por isso não há diferenças físicas entre vespas e abelhas, exceção para a rainha e as operárias. A principal diferença morfológica existente nessas

espécies é o tamanho das operárias. Operárias grandes tendem a forragear enquanto as pequenas tendem a trabalhar dentro da colméia. Dentro dessas espécies, a rainha pode ter um comportamento passivo (máquina de botar ovos) ou um comportamento agressivo com suas irmãs operárias.

De acordo com Robinson (ROBISON, 1992) o polietismo temporal das abelhas está relacionado com a taxa de *Juvenile Hormone* (JH) existente no sangue. Portanto este hormônio está envolvido na regulação do desenvolvimento comportamental de adultos trabalhadores em colméias de abelhas. Os níveis de JH no sangue tipicamente aumentam com o passar do tempo; eles são baixos em abelhas que trabalham dentro da colméia e alto em abelhas encarregadas em forragear ou engajadas em defender a colônia. Além disso, forrageadoras precoces possuem um aumento precoce nos seus níveis de JH, abelhas que trabalham com o cuidado das larvas, possuem um nível baixo e abelhas que passaram a atividades internas, após serem forrageadoras, apresentaram uma redução no seu JH.

O artigo de Schulz et Al. (SCHULZ; HUANG; ROBINSON, 1998) relata-se experiências em que um grupo de forrageadoras transportado para uma colônia acabava por inibir o desenvolvimento de novas abelhas para esta tarefa, e se o contrário ocorresse, fossem eliminadas abelhas forrageadoras, simulando a atividade de predadores, abelhas novas acabavam por se desenvolver mais depressa que o normal. A natureza desses processos inibitórios, segundo os próprios autores do artigo, ainda é desconhecida. No entanto, os resultados obtidos reforçam a idéia de que o nível de JH no sangue está diretamente relacionado com a mudança de comportamento da abelha.

3.2.2 Comunicação

As formas de comunicação entre os insetos sociais são variadas; inclui pequenas pancadas, sons, toques de antenas, liberação de substâncias químicas entre muitos outros. Devido à grande variedade de formas de comunicação torna-se muito complicado estudar todos os detalhes das interações entre os indivíduos de uma colônia, por isso, a comunicação dos insetos sociais respeita três generalizações:

A maior parte da comunicação é química, através de feromônios. O conhecimento de sinais visuais e esparso é muito simples. Em alguns grupos, particularmente os cupins e formigas subterrâneas, esta forma de comunicação não possui nenhum papel no dia-a-dia da colônia. Os sons também são pouco identificáveis pelos insetos sociais e não são definidos em nenhum sistema de comunicação importante. Os toques são amplamente utilizados nas colônias de insetos, mas também não foi identificada nenhuma espécie de código capaz de carregar grandes quantidades de informação. Por outro lado, os sinais químicos se apresentam em praticamente todas as categorias de comunicação, principalmente através de feromônios, capazes de estimular ou inibir diversas características fisiológicas nos membros de uma colônia.

Qualquer forma de comunicação sempre encontra paralelo em espécies pré-sociais.

As hierarquias de dominância que possuem papel chave nas sociedades de abelhas e vespas possuem um precedente no comportamento territorial de muitas espécies solitárias e substâncias de alarme são em muitos casos simples modificações de secreções defensivas.

O comportamento complexo da colônia emerge da integração de simples padrões individuais através da comunicação. (WILSON, 2000) apresenta nove categorias

de resposta nos insetos sociais. São elas: alarme; atração simples ou múltipla; recrutamento (para um novo ponto de coleta de comida ou um ponto específico do ninho); assistência; *trophallaxis* (troca de líquido oral e anal); troca de partículas de comida; efeitos de grupo através do estímulo ou inibição de uma atividade específica; reconhecimento de membros da mesma colônia ou membros de castas diferentes; determinação de casta por inibição ou estimulação.

3.3 Modelos Teóricos de Divisão de Trabalho em Insetos Sociais

Baseado nos estudos e observações realizadas pelos entomologistas a respeito das colônias de insetos sociais, foram concebidos modelos teóricos e matemáticos sobre sua organização. Tais modelos, construídos em sua maioria com o objetivo de simular o funcionamento das colônias para, em paralelo com o estudo de campo e o trabalho empírico, tentar compreender a organização dos insetos reais.

Em (BONABEAU; THRAULAZ; DORIGO, 1999), capítulo 8, são apresentados alguns desses modelos. Uma característica comum a estes modelos é que, em sua maioria, estes consideram a alocação de tarefas orientada pelo padrão de iterações que cada indivíduo experimenta.

Os modelos apresentados até então produzem resultados que se assemelham ao comportamento observado de colônias reais, mas não demonstram o modo como os insetos operam. Assim, estes modelos ainda não puderam ser utilizados para o objetivo a que foram concebidos, pois não se sabe se os detalhes empíricos modelados são suficientes.

Um dos modelos apresentados mais recentemente em (BONABEAU; THRAULAZ; DORIGO, 1999), capítulo 3, pretende ser um modelo genérico que cobre todos os aspectos que envolvem a divisão do trabalho nas colônias.

Não se conhece referência na literatura da aplicação do modelo citado acima para simular colônias reais. Tal modelo tem sido aplicado diretamente em aplicações de Inteligência Artificial.

Neste modelo cada indivíduo da colônia tem um limiar de resposta a estímulos para realizar determinada tarefa. Um indivíduo passa a executar uma tarefa quando o estímulo para executar esta tarefa ultrapassa seu limiar associado.

Seja s a intensidade de um estímulo associado a uma atividade em particular, onde s pode ser o número de encontros com outros indivíduos, uma concentração química ou qualquer outro fator quantitativo que possa ser sentido por um indivíduo. O limiar de resposta θ , expresso em unidades de intensidade de estímulo, é uma variável interna que determina a tendência de um indivíduo, respondendo ao estímulo s , realizar a tarefa associada.

As Equações 3.1 e 3.2 mostram funções para cálculo da probabilidade (T), ou tendência, de um indivíduo atender a resposta a um estímulo. Outras funções podem levar ao mesmo padrão de resultados esperado pelo modelo. Qualquer número inteiro maior que 1 pode ser usado com expoente de s , que determina o índice de crescimento de θ , porém o autor utiliza em todos seus exemplos o expoente 2.

$$T_{\theta}(s) = \frac{s^2}{s^2 + \theta^2} \quad (3.1)$$

$$T_{\theta}(s) = 1 - e^{-\frac{s}{\theta}} \quad (3.2)$$

Em ambas as equações de cálculo de tendência (Equações 3.1 e 3.2), se o valor de θ

for muito pequeno a probabilidade do indivíduo atender ao estímulo tende a 0 e se o valor de θ for muito grande tal probabilidade T irá tender a 1. Com $s = \theta$ a probabilidade é exatamente $1/2$. O comportamento das Equações 3.1 e 3.2 é equivalente, variando apenas a curva de resposta aos estímulos.

Se considerarmos o estímulo associado à tarefa de recolher um objeto como o número de encontros do indivíduo com objetos daquele tipo e se a cada encontro com um novo ítem o indivíduo tiver uma probabilidade ρ de recolher aquele objeto, então a probabilidade do indivíduo não responder aos primeiros N encontros é $(1 - \rho)^N$.

Sendo assim, a probabilidade $P(N)$ de um indivíduo responder a N encontros é dada por: $P(N) = 1 - (1 - \rho)^N = 1 - e^{N \ln(1-\rho)}$

Isto é o que diz a Equação 3.2 com $s = N$ e $\theta = \frac{-1}{\ln(1-\rho)}$.

O limiar de resposta a um estímulo, valor de θ nas funções acima, pode variar para cada indivíduo para refletir o polimorfismo e o polietismo temporal dos insetos sociais. Indivíduos fisicamente diferentes ou de diferentes idades podem ter tendências diferentes a executar determinadas tarefas.

3.3.1 Mecanismos de regulação da divisão de trabalho

De acordo com o modelo de especialização apresentado por Bonabeau em (BONA-BEAU; THRAULAZ; DORIGO, 1999), o limiar de resposta é atualizado por reforço. Cada indivíduo no modelo tem um limiar de resposta diferente para cada tarefa que varia de acordo com as atividades que o agente exerce na colônia. O limiar de resposta θ_{ij} de um indivíduo i é atualizado de acordo com a Equação 3.3 quando o indivíduo está executando a tarefa j . Como o limiar de resposta diminui enquanto o indivíduo executa a tarefa, associa-se esse comportamento a um tipo de aprendizado.

$$\theta_{ij} = \theta_{ij} - \xi \Delta t_{ij} \quad (3.3)$$

Onde:

ξ coeficiente de aprendizado

Δt_{ij} intervalo de tempo do indivíduo i na tarefa j

O limiar de resposta θ_{ij} quando *não* está executando uma tarefa durante um intervalo de tempo Δt_{ij} é dado pela Equação 3.4

$$\theta_{ij} = \theta_{ij} + \rho \Delta t_{ij} \quad (3.4)$$

Onde ρ é o coeficiente de esquecimento.

O estímulo s associado a uma tarefa deve variar conforme sua execução é satisfatória ou não. Uma tarefa pode ser executada de forma ineficiente, seja por um ou mais indivíduos, o que deve aumentar seu estímulo associado para fazer com que outros indivíduos se engajem na sua execução.

A variação da intensidade de um estímulo associado à execução de uma tarefa, que reduz a intensidade do estímulo, e associado ao crescimento natural da demanda é dada pela Equação 3.5.

$$s(t+1) = s(t) + \delta - \frac{\alpha N_{act}}{N} \quad (3.5)$$

Onde:

$s(t + 1)$ estímulo associado a tarefa no tempo $t + 1$.

$s(t)$ estímulo associado a tarefa no tempo tj .

δ crescimento da intensidade em função do tempo.

α fator de escala associado a eficiência da realização da tarefa.

N_{act} número de indivíduos ativos.

N número de indivíduos que podem ser tornar ativos na colônia.

Todo indivíduo tem a mesma probabilidade fixa de deixar de executar uma tarefa e pode retomar sua execução imediatamente se o estímulo que este indivíduo tem para executar a tarefa que acabou de abandonar for superior ao seu limiar.

3.4 Conclusão

O estudo sobre as colônias de insetos sociais revela sua interessante capacidade de adaptação às condições do meio em que vivem. Tal capacidade, garantia de seu sucesso ecológico, mostra como um comportamento sofisticado emerge da iteração de seres bastante simples.

Conhecer os aspectos biológicos dos insetos sociais é fundamental para entender seu comportamento e, por conseqüência, melhor compreender os modelos teóricos elaborados com base na organização de sua colônia. Muitos autores apresentaram modelos neste sentido e alguns destes modelos foram amplamente utilizados na construção de diferentes abordagens para a solução de muitos problemas envolvendo Sistemas Multiagentes.

A adaptabilidade dos insetos sociais a ambientes dinâmicos é um fator importante que pode ser utilizado na coordenação dinâmica de agentes. Os mecanismos de organização dos insetos sociais serão aplicados à coordenação de semáforos no modelo apresentado na Seção 5.1 do Capítulo 5.

4 OTIMIZAÇÃO DE RESTRIÇÕES

4.1 Introdução

Diversos problemas da Ciência da Computação podem ser modelados como problemas de satisfação de restrições (*Constraint Satisfaction Problems – CSP*). Resolver um CSP é encontrar o conjunto de atribuições que satisfaça um conjunto de restrições. Esse problema genérico pode modelar diversas aplicações, tais como: alocação de recursos, escalonamento, reconhecimento de padrões, entre outros. Diversos problemas reais não podem ter todas as restrições sendo respeitadas ao mesmo tempo, para este tipo de problema, busca-se uma solução que reduza ao mínimo o número de restrições quebradas, essa classe de problemas são chamados de problema de otimização de restrições (*Constraint Optimization Problem – COP*) ou problemas de satisfação de restrições parcial (*Partial Constraint Satisfaction Problem – PCSP*).

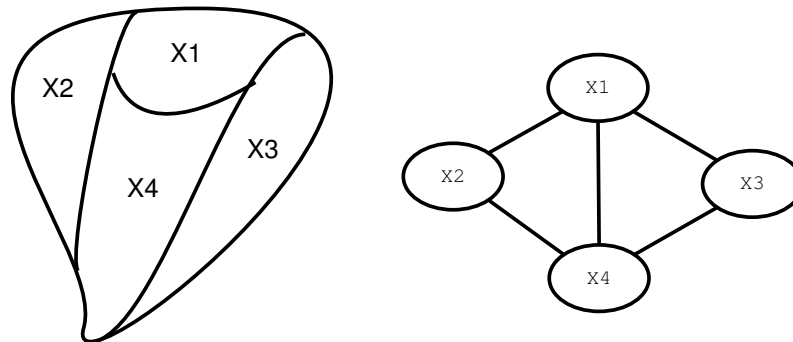


Figura 4.1: Mapa a ser colorido e a direita o grafo de restrições que o representa.

Um Problema de Satisfação de Restrições (PSR) pode ser definido como uma tupla $\langle X, D, C \rangle$, onde: X é um conjunto finito de variáveis, X_1, X_2, \dots, X_n ; C um conjunto de restrições e D uma função de mapeamento definida em $X \times C$. A solução do problema é um conjunto de atribuições de valores às variáveis que satisfaça todas as restrições. No caso de um problema de otimização de restrições, a solução é o conjunto de atribuições que satisfaça o maior número de restrições possível.

Os problemas de otimização de restrições surgem pelos mais diversos motivos, de acordo com (FREUDER; WALLACE, 1992) alguns dos motivos são:

- O problema tem restrições demais e não possui solução completa.

- O problema é muito difícil para ser solucionado completamente, mas uma solução “suficientemente boa” é aceitável.
- Procura-se a melhor solução possível dada a limitação de alguns recursos.
- Soluções em tempo real necessitam de um algoritmo que retorne soluções “em qualquer momento” e que possam retornar uma solução quase imediata e melhorá-la, caso haja tempo.

Os PSRs podem ser representados por um grafo de restrições, no qual as restrições são os arcos e os nodos são as variáveis do problema. Um exemplo clássico de PSR é a coloração de mapas, como o grafo da Figura 4.1. Para a coloração com duas cores não existe um conjunto de atribuição de cores que satisfaça todas as restrições ao mesmo tempo. Considerando cada restrição com um custo 1 teríamos um valor ótimo possível de 1, se fosse um problema de otimização. Considerando 3 cores seria possível encontrar uma solução, ou seja, um conjunto de atribuições na qual não há restrições não satisfeitas, por exemplo: $\{X_1 = azul, X_2 = verde, X_3 = verde, X_4 = branco\}$.

4.2 Resolução de PSR via Busca com Retrocesso

Os PSRs podem ser resolvidos utilizando-se um método no qual geram-se sistematicamente todas as possibilidades possíveis de combinações de variáveis e, a cada combinação gerada, verifica-se se todas as restrições são satisfeitas. O conjunto de variáveis que satisfaz todas as restrições é a solução do problema. O número de combinações que podem ser analisadas por esse método é dado pelo produto Cartesiano de todos os domínios de variáveis, sendo extremamente ineficiente.

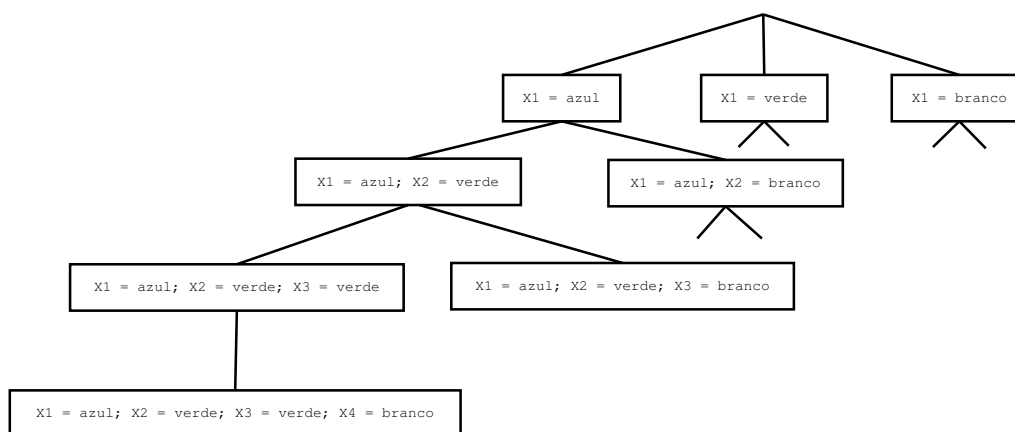


Figura 4.2: Parte da árvore de busca para a coloração do mapa da Figura 4.1.

A busca com retrocesso (*Backtracking*) é um método clássico de resolução de PSR. Segundo (RUSSEL; NORVIG, 1995), a expressão “busca com retrocesso” é utilizada para indicar uma busca em profundidade que escolhe valores para uma variável de cada vez e que efetua o retrocesso quando uma variável não tem valores válidos restantes a serem atribuídos. Neste tipo de resolução, assume-se que os espaços X_i da seleção contêm apenas um número finito de valores distintos. Os números finitos podem ser grandes e

podem corresponder à discretização de intervalos contínuos. Assumindo M_i o número de valores distintos em X_i , e $M = \prod_{i=1}^n M_i$, a aproximação por “força bruta” é formar cada um dos vetores possíveis da amostra de M , avaliar cada resultado na função ϕ do critério de otimização e ver qual produz o valor maior. O algoritmo de busca por retrocesso é projetado gerar a mesma resposta com menos tentativas de M e quando o conjunto M é grande (como é geralmente) isto se transforma um ganho muito importante. Apesar de se melhor que a geração de todas as soluções possíveis, segundo (KUMAR, 1992), para a maioria dos problemas práticos a busca com retrocesso simples o tempo de execução é exponencial.

A idéia básica deste tipo de busca é construir sobre um vetor da amostra um componente a cada momento e usar funções de critério para testar se o vetor que está sendo gerado ainda tem uma possibilidade de sucesso (GOLOMB; BAUMERT, 1965). Por exemplo, para resolver o PSR representado pelo grafo mostrado na Figura 4.1 seria gerada uma árvore de busca, parte desta árvore pode ser vista na Figura 4.2.

O algoritmo de busca com retrocesso bem construído é aquele que elimina grandes regiões do espaço Cartesiano de busca. A eficiência do método depende muito de como a função de critério é modelada, já que uma função com poucos critérios faz com que um espaço muito grande de soluções seja considerado e uma função muito elaborada pode demorar muito tempo para ser executada em cada caso.

4.2.1 *Branch and Bound*

O *Branch and Bound* pode ser utilizado para problemas de otimização de restrições. Ao contrário da busca com retrocesso simples que busca uma solução completa e não retorna uma solução parcial, o algoritmo *Branch and Bound* busca uma solução onde o menor número de restrições seja quebrado. Tomemos como exemplo novamente o caso da coloração do grafo da Figura 4.1: caso tivéssemos apenas duas cores o algoritmo *Branch and Bound* mostrado no Algoritmo 4.1, adaptado de (FREUDER; WALLACE, 1992), seria iniciado tendo os seguintes valores:

- *Variáveis* = $\{X_1, X_2, X_3, X_4\}$;
- *Valores* = $\{\{azul, verde\}, \{azul, verde\}, \{azul, verde\}, \{azul, verde\}\}$;
- *Caminho_de_busca* = $\{\}$;
- $S = 0$, número mínimo de restrições que podem ser violadas;
- $N = \infty$, número máximo de restrições que podem ser violadas;
- *Distância* = 0, o número de restrições violadas;

Uma busca com retrocesso simples não retornaria uma solução já que o problema tem muitas restrições e é impossível resolvê-lo sem a quebra de nenhuma delas. O *Branch and Bound* funciona guardando a melhor solução encontrada até o momento e abandona uma busca quando essa levará a um resultado pior do que o resultado atual. A Figura 4.3 mostra parte da árvore de busca do algoritmo. Seguindo o algoritmo, primeiro seria avaliada a variável X_1 com o primeiro valor possível, no caso “azul”.

Algoritmo 4.1 *Branch and Bound*

```

se Variaveis == vazio então
  Melhor_solucao ← Caminho_de_busca;
  N ← Distancia;
  se  $N \leq S$  então
    retorna “acabado”;
  senão
    retorna “continue a procura”;
  fim se
senão se Valores == vazio então
  retorna “continue a procura”;
senão se Distancia = N então
  retorna “continue a procura”;
senão
  Valor_Atual ← primeiro valor em Valores;
  Nova_Distancia ← Distancia;
  tenta escolhas no Caminho_de_busca com  $Nova\_Distancia < N$ :
  se a escolha é inconsistente com o Valor_Atual então
    Nova_Distancia ← Nova_Distancia + 1;
  fim se
  se  $Nova\_distancia < N$  e
  Branch and Bound(Caminho_de_busca mais Valor_Atual, Nova_Distancia,
  Variaveis menos a primeira, Valores da segunda variável em Variaveis) = “aca-
  bado” então
    retorna “acabado”;
  senão
    retorna Branch and Bound(Caminho_de_busca, Distancia, Variaveis,
    Valores menos Valor_atual);
  fim se
fim se

```

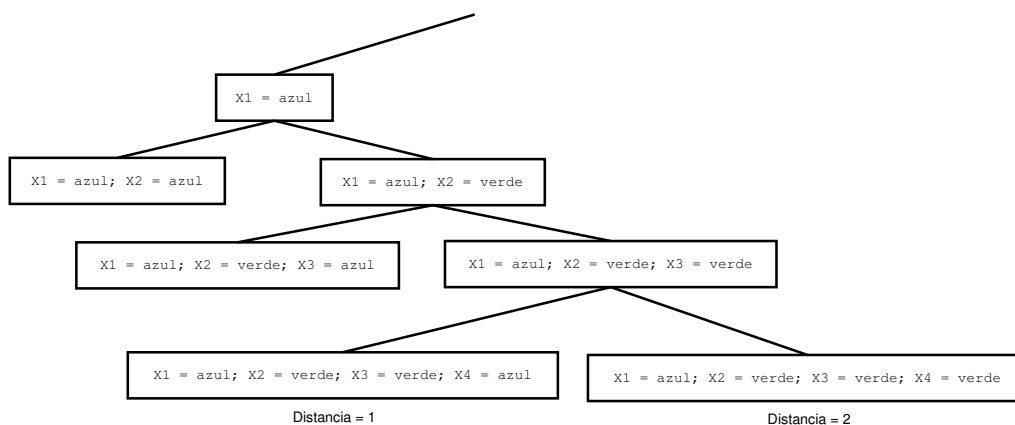


Figura 4.3: Parte da árvore de busca por *Branch and Bound*.

4.3 Mediação Cooperativa

Diversos problemas podem ser modelados como PSR ou problemas de otimização de restrições. A resolução por “Busca com Retrocesso”, bem como suas variantes, é o mé-

tudo de resolução mais utilizado porém tem alto custo para problemas muito complexos e é uma solução que analisa o problema de maneira centralizada.

A mediação cooperativa é um método multiagente de resolução de problemas de otimização distribuídos cuja principal vantagem é a divisão do problema em diversos sub-problemas, que são mais facilmente resolvidos por um agente mediador. O algoritmo OptApo ainda possui a vantagem de criar relacionamentos entre variáveis inicialmente não relacionadas e evitar a repetição de erros por parte do mediador.

O *Optimal Asynchronous Partial Overlay* (OptApo) é um protocolo de mediação cooperativa para a resolução de problemas do tipo otimização de restrições distribuído. O algoritmo OptAPO apresentado em (MAILLER; LESSER, 2004), criado a partir do algoritmo APO apresentado em (MAILLER; LESSER, 2003) como um método de resolução de PSR distribuídos, permite que os agentes estendam o contexto utilizado para a tomada de decisão criando um *grafo de relacionamentos*. Um agente está localizado em cada nodo e as arestas representam as restrições do problema. Cada restrição ou relação funcional tem um custo associado, sendo que o custo é zero quando não há conflitos entre os dois nodos ligados por esta aresta. Dentro deste grafo, que pode ser um sub-grafo de um problema maior, um agente deve tomar o papel de mediador e tentar encontrar uma solução que reduza o custo de todos os agentes envolvidos na sessão de mediação. O algoritmo é composto de três estágios: inicialização, verificação da visão do agente e mediação.

Durante a execução do algoritmo, o agente mediador tenta minimizar o custo de seu sub-problema, do qual ele tem informações em uma lista denominada *good_list*. O mediador calcula o custo do sub-problema e tenta mudar os valores das variáveis que estão na sessão, com o objetivo de obter o valor ótimo. Toda a vez que isso não pode ser alcançado sem causar custo para os agentes que estão fora da sessão, o mediador cria conexões (*links*) com esses agentes, assumindo que eles estão envolvidos no custo de sua sub-estrutura. O processo continua até que cada um dos agentes tenha justificado o custo de seus sub-sistemas e garantido que a sua estrutura contém todas as restrições relacionadas ao custo do qual ele é parte.

4.3.1 Inicialização

Durante o processo de inicialização, cada agente inicializa suas variáveis: valor corrente (v_i^2), nome da variável (x_i), prioridade (p_i), domínio (D_i), relações funcionais (C_i), intenção de mediar (m_i), *good_list* e *agent_view*. O agente envia para cada um de seus vizinhos uma mensagem com suas variáveis correntes, menos suas duas listas (*good_list* e *agent_view*). As listas serão construídas a partir das mensagens de resposta recebidas por seus vizinhos.

Quando o agentes recebe as mensagens de resposta dos vizinhos, adiciona informação na *agent_view* e adiciona as variáveis à *good_list*, caso possível. A variável é somente adicionada à *good_list* caso haja um caminho direto entre o agente que tem a variável.

4.3.2 Verificação da visão do agente

Após o período de inicialização, o agente calcula o custo corrente do sub-grafo formado pelos agentes em sua *good_list*, F_i . Se o valor desse custo for maior que o custo ótimo F_i^* , ele conduz uma sessão de mediação *ativa* ou *passiva*. No início da simulação o custo ótimo tem valor zero.

²No algoritmo original, a variável é chamada de d_i

Algoritmo 4.2 OptApo: Escolhe Solução

seleciona uma solução s com o algoritmo *Branch and Bound* que:

1. minimize o custo para os agentes na *good_list*;
2. minimize o custo para os agentes fora da seção;

$F_i^* \leftarrow \text{custo}(s)$;

$F_i' \leftarrow F_i + \text{custo atual para os agentes fora da sessão}$;

$F_s' \leftarrow F_i^* + \text{custo de usar a solução } s \text{ para os agentes fora da sessão}$;

se *mediação* == **ativa** e $F_s' \leq F_i'$ **então**

$v_i \leftarrow v_i'$;

fim se

para todo $x_j \in \text{agent_view}$ **faça**

se $x_j \in \text{preferences}$ **então**

se $d_j' \in s$ viola um x_k e $x_k \notin \text{agent_view}$ **então**

envia(*init*, ($x_i, p_i, v_i, m_i, D_i, C_i, \text{caminho}_{i,k}$)) para x_k ;

adiciona x_k na *initList*;

fim se

se *mediação* == **ativa** e $F_s' \leq F_i'$ **então**

envia (*aceita!*, (d_j', x_i, p_i, v_i)) para x_j ;

atualiza a *agent_view* para x_j ;

senão se *mediação* == **ativa** e $F_s' > F_i'$ **então**

envia (*aceita!*, (d_j, x_i, p_i, v_i)) para x_j ;

fim se

senão se *mediação* == **ativa** **então**

envia (*valor?*, (x_i, p_i, v_i, m_i, c_i)) para x_j ;

fim se

fim para

mediação \leftarrow **nenhuma**;

verifica a visão do agente;

O agente pode executar uma mediação *ativa* caso não haja em sua *good_list* nenhum agente com prioridade maior ou igual a dele próprio. Existem duas diferenças entre mediação *ativa* e *passiva*, a primeira está no marcador (*flag*) de mediação (m_i), esse marcador impede que o paralelismo gere instabilidade no sistema, assim dois agentes não podem mediar dentro do mesmo sub-problema. A segunda diferença, está na intenção do agente: o agente com mediação passiva tem intenção de mudar o F_i^* e o agente com mediação ativa tem intenção de mudar F_i^* e F_i .

4.3.3 Mediação

No estágio de mediação, agentes que receberam uma requisição de mediação podem mandar respostas sobre suas restrições ou um pedido de espera. A avaliação das restrições consiste em avaliar cada um dos elementos do domínio e rotular com os nomes dos agentes os quais compartilham uma relação funcional de custo $f_i > f_i^*$ e enviar como uma mensagem para o mediador. após receber a resposta de todos os agentes que estão em sua *good_list*, o mediador executa o Algoritmo 4.2 para encontrar a solução. Na primeira parte ele executa uma busca do tipo *Branch and Bound* e encontra uma solução s , de custo F_i^* . Se está executando uma mediação ativa, compara o valor do custo atual com os custo resultante da utilização da solução s no sub-sistema estendido (agentes na *good_list* mais

os agentes que estão nas “preferências”).

Após calcular esses valores, o mediador envia mensagens para todos os agentes que não estejam em sua *agent_view* que tiveram os custos aumentados devido a utilização da solução *s*. Isso ocorre mesmo quando o mediador não usa a solução *s*. Esse passo faz com que o agente crie conexões com mais agentes, aumentando a *good_list*, e reduzindo a chance de ocorrer o mesmo erro na próxima mediação. Se a mediação era ativa, o mediador envia as mensagens do tipo “aceita!” para os agentes na sessão, como os novos valores de variáveis. Os agentes que receberam a mensagem “aceita!” adotam a solução proposta e atualizam sua *agent_view*.

4.4 Conclusão

Os mecanismos de resolução de problemas de otimização de restrições tradicionais são executados de forma centralizada. O modelo de resolução com mediação cooperativa é um meio termo entre a resolução distribuída e a centralizada.

No algoritmo OptApo, os agentes têm uma visão parcial do problema e interagem com seus vizinhos (diretos ou indiretos) tentando encontrar a melhor solução para problemas locais, gerando uma solução global melhor possível. Esse mecanismo de resolução será aplicado para a resolução do problema de coordenação de semáforos no modelo proposto que será apresentado na Seção 5.2 do Capítulo 5.

5 ABORDAGENS PROPOSTAS DE COORDENAÇÃO DE SEMÁFOROS

No presente Capítulo, duas abordagens de coordenação de semáforos são apresentadas. Cada uma utiliza um conceito de coordenação diferente: a primeira inspira-se nos conceitos de *Swarm Intelligence*, (OLIVEIRA; FERREIRA; BAZZAN, 2004b), apresentados no Capítulo 3 e a segunda utiliza conceitos de otimização de restrições, (OLIVEIRA; BAZZAN; LESSER, 2005), apresentados no Capítulo 4, e o modelo de resolução de problemas dessa natureza. Ambos modelos de coordenação são descentralizados, no entanto no primeiro modelo (inspirado em insetos sociais), a coordenação é completamente autônoma e a comunicação é indireta; já o segundo modelo utiliza uma coordenação que não é completamente descentralizada e há uma comunicação explícita entre os agentes inter-relacionados.

5.1 Modelo Utilizando Insetos Sociais

5.1.1 Descrição

No Capítulo 3 foram apresentados alguns modelos matemáticos que formalizam a hipótese de como a divisão do trabalho pode ocorrer em colônias de insetos sociais. Interações entre os insetos e o seu meio resultam em uma distribuição de tarefas que se adapta às necessidades da colônia. Utilizaremos um dos modelos que descreve a distribuição de tarefas associadas ao estímulo do meio e as tendências de cada indivíduo. Cada indivíduo possui um limiar específico de estímulo para executar cada tipo de tarefa, no entanto, esse limiar pode mudar com o tempo para que o indivíduo tenha uma melhor resposta diante às exigências da colônia.

Esses conceitos são utilizados nesta abordagem de modo que cada semáforo seja um agente com comportamento de inseto social. Cada um de seus planos semaforicos é visto como uma tarefa a ser executada e tem um estímulo associado que varia de acordo com as mudanças no meio. O estímulo de cada plano semaforico é relacionado com o rastro de feromônio deixado pelos veículos que estão trafegando nas ruas controladas. O agente percebe os rastros de feromônio e assim identifica o estímulo relacionado com cada plano. Por agirem como insetos sociais, os agentes possuem limiares associados a cada tarefa que indicam a propensão do agente em executá-la dado um certo estímulo.

5.1.2 Simulador Microscópico de Tráfego

Para as simulações deste modelo, foi utilizado o modelo de Nagel–Schreckenberg (NAGEL; SCHRECKENBERG, 1992) o qual é um modelo microscópico de tráfego originalmente inspirado em princípios de Autômatos Celulares (AC). Nesse modelo, cada

rua é dividida em células de tamanho fixo, que podem conter um veículo, assim, a rua é representada por uma matriz na qual os veículos podem estar posicionados. Cada veículo tem uma velocidade que é representada pelo número de células que ele pode percorrer por ciclo de simulação. O comportamento do veículo é representado por algumas regras de aceleração, desaceleração e movimentação. As regras estão listadas abaixo:

Regra I Aceleração: Se a velocidade v do veículo é inferior à $v_{máxima}$ e a distância até o próximo veículo é superior à $v + 1$, então ele deve acelerar, aumentando sua velocidade $v \leftarrow v + 1$;

Regra II Desaceleração: Caso a distância até o próximo veículo seja inferior ou igual a sua velocidade, isto é, $v \geq gap$ (gap é o termo empregado para designar a distância inter-veicular, isto é, a distância entre o veículo que se está analisando até o veículo mais próximo, neste caso, o imediatamente a frente) então o veículo deve reduzir sua velocidade: $v \leftarrow gap$;

Regra III Aleatoriedade: cada veículo pode, com um probabilidade $p_{desaceleração}$, reduzir sua velocidade em uma unidade, isto é, $v \leftarrow v - 1$;

Regra IV Movimentação: cada veículo avança v células.

Esse simples, mas válido modelo microscópico de tráfego pode ser implementado de modo eficiente para ser utilizado em simulações de tempo real e controle de tráfego.

Para a representação da rede, cada rua é descrita como uma composição de nodos, que representam os cruzamentos e arestas que representam os segmentos de via. A expressão aresta se refere a arestas dirigidas representando uma direção do movimento da rua, por exemplo uma rua com dois sentidos teria duas arestas com sentidos opostos.

No cenário de tráfego urbano mais elementos foram adicionados semáforos e tipos mais complexos de intersecções. A ferramenta de simulação, chamada ITSUMO – *Intelligent Transportation System for Urban Mobility*, (SILVA et al., 2005), contém elementos diversos como: faixas, veículos, injetores de veículos (*sources*), eliminadores de veículos (*sinks*), detectores e semáforos. A configuração topológica da rede e os parâmetros da simulação pode ser armazenados em um banco de dados ou em arquivos do tipo XML (*Extensible Markup Language*).

O simulador inicializa a rede (pelo banco de dados ou lendo o arquivo XML), inicia o cenário e atualiza os detectores a cada intervalo de tempo pré-definido. O visualizador 3D pode ser utilizado para visualizar a topologia, Figura 5.1. O comportamento dos agentes que controlam os semáforos teve de ser feito como um módulo a parte do simulador e se comunicava com ele através do banco de dados. O simulador ITSUMO possui módulos que simulam apenas agentes motoristas. Mais detalhes sobre esse simulador podem ser encontrados em: ANDRIOTTI (2001); ANDRIOTTI; BAZZAN (2002); ANDRIOTTI (2004); SILVA et al. (2004).

5.1.3 Cálculo do Estímulo

Utilizamos um modelo no qual os veículos liberam um rastro de feromônio por onde passam, esta metáfora é válida já que na maioria dos sistemas de controle de semáforo são instalados detectores que monitoram a quantidade de veículos próximos ao semáforo. O

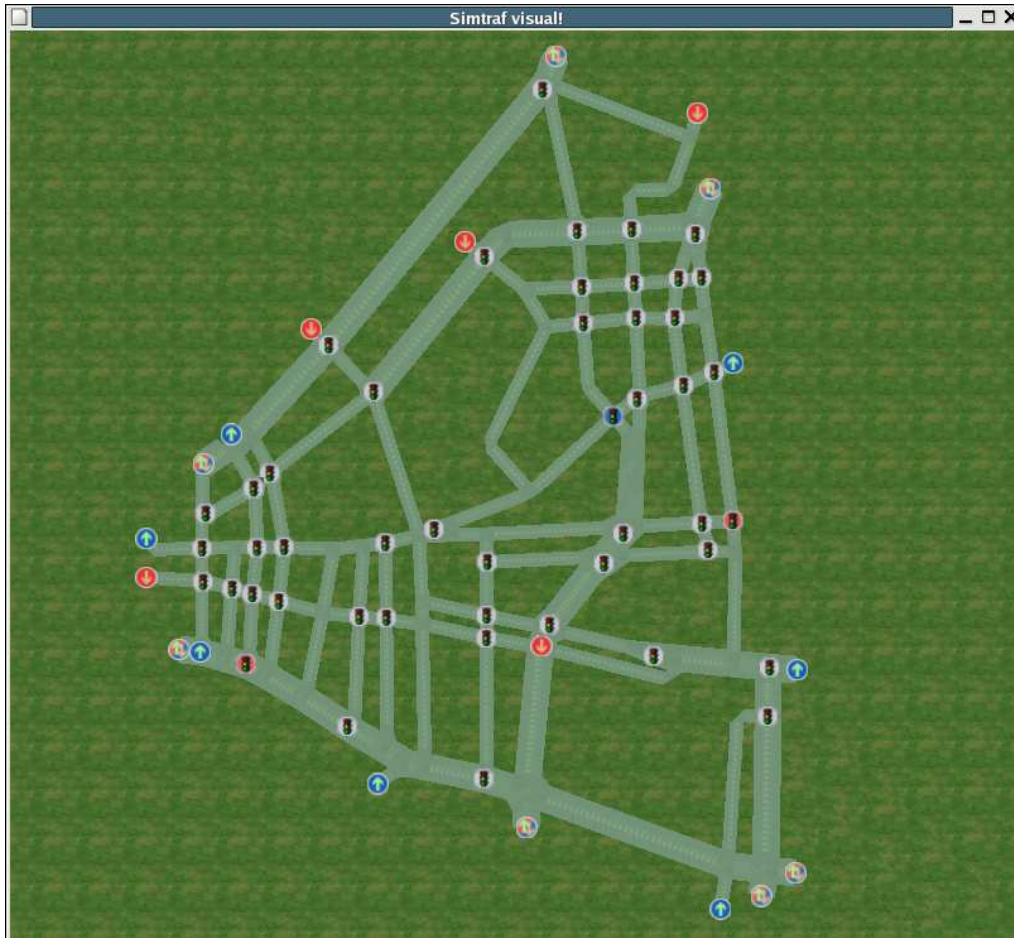


Figura 5.1: Rede analisada vista com o Visualizador 3D

rastros deixados pelo veículo tem uma taxa de dissipação (β) no tempo, desta forma o semáforo é mais sensível às mudanças no cenário ao longo do tempo. Esse rastro de feromônio pode ser visto como um tipo de comunicação primitiva chamada stigmergia (criada por Grassé em (GRASSÉ, 1959), do grego *stigma* = “marca” e *ergon* = “trabalho”), que pode ser definida como comunicação através da mudança no meio. Uma concentração maior de feromônio em uma direção específica pode indicar que o agente vizinho naquela direção mudou o seu plano ou a direção está recebendo uma quantidade maior de veículos.

O estímulo de um plano semaforico é calculado a partir dos rastros de feromônio acumulados em cada faixa controlada por ele. A densidade ($d_{l,t}$) do rastro de feromônio acumulado em uma faixa é o rastro acumulado na faixa l no tempo t . Enquanto os veículos estão parados eles continuam a liberar feromônio, sendo assim, uma faixa onde há muitos veículos parados tem um rastro de feromônio muito denso. Os rastros dissipam com o tempo a uma taxa β , assim a densidade varia com o tempo, diminuindo caso o fluxo de veículos diminua.

$$d_{l,t} = \frac{\sum_{t=0}^w \beta^{-t}(\delta_{l,t})}{\sum_{t=0}^w \beta^{-t}} \quad (5.1)$$

Onde:

$\delta_{l,t}$ densidade de veículos em uma faixa l no tempo t

w tamanho do intervalo de tempo

β taxa de dissipação do feromônio

A densidade do rastro de feromônio no tempo atual é calculada levando-se em conta as densidades (de veículos) dos tempos anteriores, como mostra a Equação 5.1. Em todas as simulações, a taxa de dissipação é mantida constante em 0,5, indicando que o feromônio perde a metade de sua intensidade a cada segundo.

$$\delta_{l,t} = \frac{\sum_{l=1}^L N_{l,t}}{L} \quad (5.2)$$

Onde:

L tamanho da faixa em número de células

N número de veículos

A densidade de veículos em uma faixa δ_l é calculada de acordo com a Equação 5.2. Se o horizonte de tempo T é 1 (como usado para o cálculo da densidade hormonal), então não é necessário o somatório em T . O valor da densidade δ sempre situa-se entre “0” e “1”, já que a célula não pode conter mais de um veículo.

O estímulo s de um plano j é calculado utilizando-se uma soma ponderada do feromônio acumulado em cada uma de suas fases. Cada fase tem um percentual de tempo ($\Delta_k = (tempo_{fim} - tempo_{início})/tempo_{ciclo}$), que indica o tempo de sinal verde que o plano dedica para aquela fase. Fases com maiores percentuais de tempo indicam o critério de prioridade do plano.

$$s_j = \sum_{k=0}^n (d_{in_{k,t}}) \Delta_k \quad (5.3)$$

Onde:

n número de fases do plano j

$d_{in_{k,t}}$ densidade de feromônio acumulado nas faixas de entrada da fase k no tempo t

Δ_k fatia de tempo da fase k

5.1.4 Alocação de Planos

A flexibilidade na troca de planos pelo agente, é consequência das mudanças induzidas pelo ambiente no estímulo e nos limiares do agente. Cada plano semafórico possui um estímulo associado de acordo com a direção que ele prioriza. Os agentes têm uma tendência maior de trocar de tarefa se essa direção têm um estímulo que ultrapassa seu limiar de resposta. A Equação 3.1, vista no Capítulo 3, define a tendência de um indivíduo executar uma tarefa como uma relação entre o estímulo e o limiar de resposta. A Equação 5.4 é uma adaptação da Equação 3.1, na qual o limiar de resposta é também relativo à tarefa j e não apenas ao indivíduo i .

$$T_{\theta_{ij}}(s_j) = \frac{s_j^2}{s_j^2 + \theta_{ij}^2} \quad (5.4)$$

Onde:

θ_{ij} limiar do indivíduo i em relação a tarefa j .

s_j estímulo associado à tarefa j .

5.1.5 Reforço

Utiliza-se o modelo de especialização no qual o limiar de resposta do indivíduo para a execução de uma tarefa é atualizado utilizando um mecanismo de aprendizado por reforço simples. Como visto no Capítulo sobre *Swarm Intelligence*, os limiares são atualizados (aumentando ou diminuindo) de acordo com um coeficiente de aprendizado ou um coeficiente de “esquecimento”.

De acordo com a entomologista Debora Gordon, (GORDON, 2000), insetos reais (mais especificamente, formigas) são diretamente influenciados pelo seu grau de sucesso na execução de uma determinada tarefa. Formigas bem sucedidas são motivadas a continuar na execução da mesma tarefa enquanto as mal sucedidas tendem a parar de executar a tarefa e buscar uma outra. Para adicionar essa propriedade que faz o inseto buscar ser bem sucedido, o modelo de aprendizado de (BONABEAU; THRAULAZ; DORIGO, 1999) (apresentado na Seção 3.3.1 Capítulo 3) foi estendido para incluir na parte da mudança da tendência um coeficiente que representa o aprendizado e o esquecimento ao mesmo tempo (quando o coeficiente é negativo o agente está esquecendo, quando é positivo ele está aprendendo). A Equação 5.5 define esta extensão.

$$\theta_{ij} = \theta_{ij} - \lambda \Delta t \quad (5.5)$$

Onde:

λ é o coeficiente de aprendizado/esquecimento

Δt um intervalo de tempo discreto

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{k=1}^n (d_k - \bar{d})^2} \quad (5.6)$$

Onde:

n número de seções de rua

d_k feromônio acumulado na seção k

\bar{d} feromônio médio acumulado em n seções de rua

O grau de sucesso de um indivíduo pode ser calculado pela Equação 5.7 ou pela Equação 5.8, na qual a diferença está no tipo de mudança que o meio produzirá no sucesso. Em ambas, um desvio padrão (σ), calculado pela Equação 5.6, elevado nas trilhas de feromônio indica um menor grau de sucesso.

$$\lambda = 1 - 2\sigma \quad (5.7)$$

$$\lambda = 2e^{(-5\sigma)} - 1 \quad (5.8)$$

Onde:

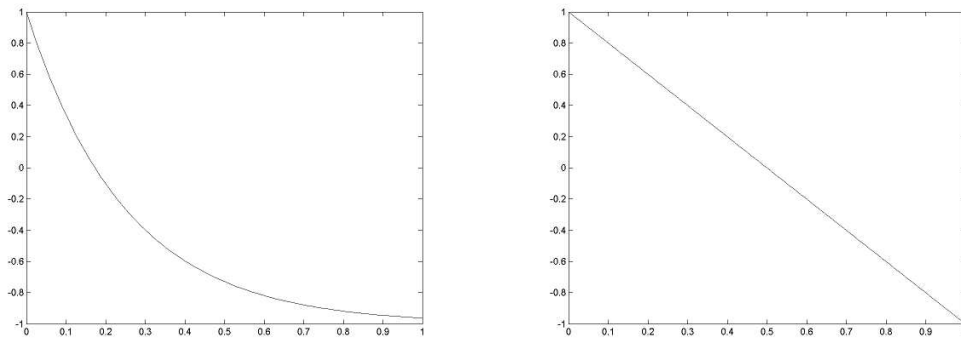


Figura 5.2: À direita, gráfico que mostra o decaimento linear do grau de sucesso e , à esquerda, o decaimento exponencial

σ é o desvio padrão do rastro de feromônio acumulado nas seções de rua controladas pelo semáforo.

Os gráficos que representam o comportamento do grau de sucesso do agente, podem ser observados na Figura 5.2. Pelo gráfico, pode-se observar que o modelo com decaimento exponencial assume uma taxa de “esquecimento” a partir do valor 0,2 para σ , enquanto na sua variante linear, o agente considera-se mal sucedido apenas quando o desvio padrão fica maior que 0,5.

O sistema tende a se manter estável e adaptado ao fluxo de tráfego mas com a capacidade de mudar para adaptar-se a novas mudanças no ambiente. Semáforos em uma mesma rua com fluxo de tráfego intenso em uma direção, tendem a adotar planos que priorizam essa direção e se coordenar com os demais vizinhos.

5.1.6 Cenário e Experimentos



Figura 5.3: Sub-rede analisada com a localização dos semáforos

O cenário é parte de uma rede real localizada na cidade de Porto Alegre. A Figura 5.3, gerada pelo módulo de visualização 3D do simulador ITSUMO, mostra as ruas que têm semáforos controlados por agentes.

A via principal, ou arterial, (no mapa real, Av. Independência seguida da Av. Mostardeiro) têm oito semáforos todos são controlados por agentes. Além desta via principal,

foram escolhidas duas ruas perpendiculares a esta, que são Av. Garibaldi (composta pelos nodos “0” a “3”) e Av. Goethe (nodos “5” a “9”).

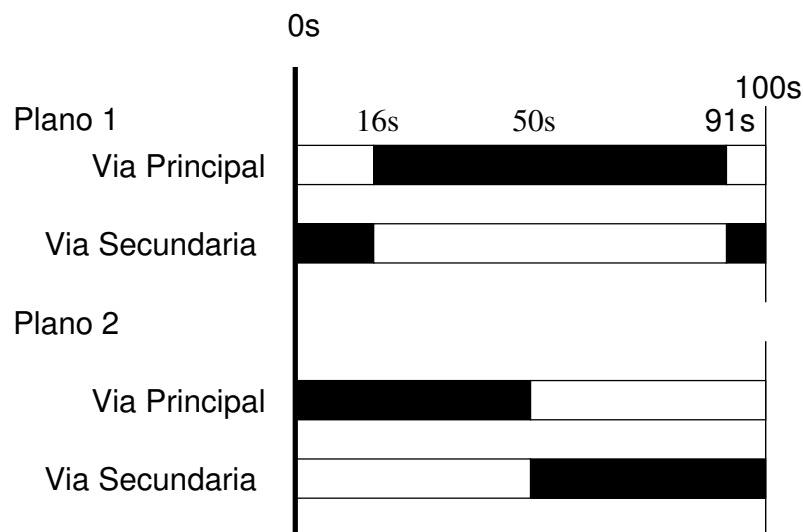


Figura 5.4: Planos semafóricos básicos (a linha em preto representa o tempo de verde)

As avenidas principais (Av. Independência e Av. Mostardeiro), que contêm os nodos 1, 4 e 6, mostrados no mapa, juntas têm 8 cruzamentos sinalizados e controlados por agentes, com planos de coordenação com os vizinhos e um plano neutro, no qual o agente não está coordenado e divide o tempo de ciclo de maneira uniforme entre todas as fases. Além da rua principal, duas ruas secundárias (Av. Goethe e Av. Garibaldi) também têm semáforos controlados por agentes. A Av. Garibaldi contém quatro semáforos controlados por agentes, sendo que um deles também faz parte da Av. Independência e está localizado no nodo “1”. A Av. Goethe possui 3 semáforos (nodos 6, 7 e 8) controlados por agentes.

Cada semáforo, controlado por um agente, tem um plano que prioriza a direção principal (Leste-Oeste para avenida principal ou Norte-Sul para as outras duas) e um que não prioriza nenhuma direção. Além desses dois planos, o semáforo pode ter planos para coordenação com outros semáforos em direções diferentes. Por exemplo, o semáforo localizado no nodo “1” possui três planos semafóricos diferentes:

Plano I: coordena na direção Leste-Oeste (prioriza Av. Independência);

Plano II: coordena na direção Norte-Sul (prioriza Av. Garibaldi);

Plano III: não coordena com outros semáforos (não prioriza nenhuma avenida).

Os planos coordenados ou sincronizados (por exemplo, o “Plano I” mostrado na Figura 5.4), são iguais (em termos de distribuição do tempo de verde entre as fases) para todos os semáforos de uma mesma rua, sendo apenas o tempo de defasagem (*offset*) de cada um diferente. O tempo de defasagem é o tempo que um veículo leva para percorrer a distância entre o semáforo anterior e o semáforo atual a uma certa velocidade. Com dois semáforos adjacentes coordenados, o veículo que parte do primeiro semáforo, com a velocidade de sincronização, não precisa parar no segundo semáforo, já que este semáforo foi programado para iniciar o seu tempo de verde atrasado em relação ao tempo de verde do semáforo anterior.

Os veículos são geralmente inseridos na rede através de injetores (*sources*) localizados nos nodos das bordas. No entanto, um nodo qualquer pode inserir veículos, caso se deseje.

Na Figura 5.3, podemos observar alguns pontos de inserção que aparecem como setas voltadas para cima (\uparrow) inseridas em um círculo azul. De modo similar, os veículos são removidos do cenário quando chegam em uma das bordas, na Figura 5.3 são representados como setas voltadas para baixo (\downarrow) inseridas em círculos vermelhos. A inserção ocorre em taxas diferentes, dependendo de sua localização.

Na Av. Independência, 36 veículos são inseridos por minuto em suas 3 faixas. Essa taxa de inserção foi definida colocando-se uma probabilidade de $3/5$ de um veículo ser inserido em cada segundo (passo) de simulação. As avenidas Goethe (sentido Norte-Sul) e Garibaldi possuem uma taxa de inserção de 24 veículos por minuto, com uma probabilidade de $2/5$ de um veículo ser inserido a cada segundo de simulação.

Além das inserções, cada cruzamento possui probabilidades de virada associadas. Desta forma, veículos provenientes da Av. Garibaldi podem virar a direita no nodo “1” e entrar na Av. Independência ou atravessar o cruzamento e permanecer na mesma direção. Neste cenário, na falta de informações reais, foram assumidos alguns valores que fazem com que os veículos tenham uma probabilidade maior de permanecer na mesma direção (a probabilidade é $1/10$ de mudar de direção).

O simulador ITSUMO permite a inserção de veículos com motoristas especiais com rotas fixas que coletam dados sobre as condições do tráfego. Para a análise desta simulação, um motorista especial foi colocado em cada uma das três avenidas em estudo para a coleta de dados. Todos os veículos desta simulação têm probabilidade zero de desaceleração espontânea, assim, não havendo congestionamento nem paradas em sinal, o veículo tende a permanecer com a velocidade máxima (no caso a velocidade utilizada foi de 3 células por segundo, sendo que a célula tem tamanho 4 metros, o equivalente a 12 metros por segundo).

5.1.7 Resultados

As simulações apresentadas nesta seção foram geradas usando o simulador discutido na Seção 5.1.2 com o cenário apresentado na Seção 5.1.6.

No início da simulação, a rede está vazia e é necessário algum tempo para que os veículos atinjam todas as partes da rede, sendo assim, os 100 ciclos iniciais da simulação são descartados dos resultados. Sendo a entrada de dados e a escolha dos planos pelos semáforos de natureza probabilística, todos os experimentos foram repetidos 10 vezes com as mesmas condições iniciais.

O objetivo dos experimentos foi comparar o tempo de percurso de cada motorista especial durante a sua rota, utilizando a abordagem proposta e comparando com a sincronização fixa e a falta de sincronização. Foram realizados quatro tipos de experimento:

Experimento tipo I mostra o tempo que os motoristas levam para percorrer as rotas em um cenário ideal e abstrato no qual não há semáforos e por consequência, não há desaceleração causada por filas. Neste cenário, o cruzamento é adimensional, logo veículos podem cruzá-lo sem colisões;

Experimento tipo II mostra um cenário no qual todos os semáforos da rota principal (Av. Independência e Av. Mostardeiro) estão sincronizados de maneira fixa e pré-determinada, executando planos que priorizam a via principal;

Experimento tipo III mostra um cenário no qual todos os semáforos estão executando planos que não priorizam nenhuma rota, isto é, planos que fornecem a mesma fatia de tempo para todas as fases;

Experimento tipo IV os semáforos são controlados por agentes com comportamento de inseto social de acordo com o modelo aqui proposto.

Nos experimentos de “tipo IV”, também avaliamos e comparamos as diferentes possibilidades da extensão do modelo (a função de sucesso), como definida na Seção 5.1.5. Sendo assim, a abordagem proposta foi simulada de quatro maneiras diferentes:

Modelo I sem reforço, os limiares não são atualizados no tempo;

Modelo II atualização do limiar de resposta utilizando a idéia original de atualização (utilizando as Equações 3.3 e 3.4, apresentadas na Seção 3.3.1 Capítulo 3), com coeficientes fixos de aprendizado e de esquecimento;

Modelo III atualização do limiar de resposta utilizando a função linear de sucesso, Equação 5.7;

Modelo IV atualização do limiar de resposta utilizando a função exponencial de sucesso, Equação 5.8;

Em todas as simulações foi adotado $\beta = 0,5$ e θ inicial de todas as tarefas e agentes igual a $0,5$. No modelo de atualização original (Modelo II), foi adotado $\xi = 0,5$ e $\rho = 0,05$.

Tabela 5.1: Tempo médio para completar o percurso em segundos

Experimento	Percurso		
	Independência e Mostardeiro	Garibaldi	Goethe
Sem semáforos	111,33	41,31	40,43
Sincronização fixa	130,04	277,99	248,87
Sem sincronização	239,80	141,63	330,24
Modelo I	277,97	129,34	157,50
Modelo II	274,13	135,95	188,44
Modelo III	294,58	123,49	174,99
Modelo IV	273,88	117,65	152,13

A Tabela 5.1.7 mostra os valores dos tempos médios do percurso para os motoristas especiais, coletados em simulações de 3h (10800 passos) que foram repetidas 10 vezes. O tempo médio foi medido com um motorista especial que percorria a mesma rota sempre que chegava ao ponto final. O tempo para completar uma rota sem o atraso causado pelos semáforos e sem desaceleração é mostrado na primeira linha. Este tempo é mostrado apenas como um comparativo (tempo mínimo possível) e não pode ocorrer em uma situação de tráfego real já que não são possíveis colisões, desaceleração ou paradas. A situação mais próxima do cenário real é mostrada na terceira linha da tabela, na qual não há sincronização. Neste caso, o tempo do percurso dos motoristas é alto pois há muitas paradas.

Quando é utilizada a coordenação fixa na rota principal, os motoristas que circulam por *esta rota* têm um melhor desempenho. Apesar do bom desempenho da rota principal, as ruas secundárias têm o fluxo bastante prejudicado. Para o fluxo de tráfego na Av.

Garibaldi ocorre a pior situação, já que o percurso demora em média 278 segundos, comparados com os 41 segundos na situação ideal. Como visto no Capítulo 2, a coordenação de uma rota arterial pode ser ineficiente para o desempenho geral da rede, visto que o volume de tráfego desta rota pode estar mais baixo que o volume das rotas adjacentes. Apenas abordagens adaptativas podem lidar com este tipo de problema. A abordagem apresentada mostra-se eficiente na percepção dos fluxos e adaptação dos planos para priorização dos maiores fluxos.

A abordagem proposta inspirada em insetos sociais, utilizando qualquer modelo de atualização dos limiares de resposta, prioriza a rota com as piores condições de tráfego (no caso, a Av. Goethe no sentido Norte-Sul). Os três diferentes modelos de atualização do limiar de resposta resultam em um comportamento muito similar em termos de priorização, isso ocorre em grande parte pela falta de mudanças no cenário, gerando praticamente o mesmo estímulo para cada plano em todo o período de simulação. Se as condições de tráfego mudam, o modelo de atualização dos limiares pode ter um impacto na mudança de tarefas (planos). Nos experimentos não foram inseridas mudanças no fluxo durante a simulação já que ainda não existe essa possibilidade no simulador ITSUMO.

5.1.8 Conclusões

A abordagem inspirada em insetos sociais mostra-se adequada para o controle de semáforos. A liberação de feromônio pelos veículos é uma metáfora adequada e factível com os detectores reais de movimentação de veículos. A comunicação por stigmergia mostra-se vantajosa pelo fato de não haver a necessidade de trocas de mensagens diretas. A abordagem foi realizada em um simulador microscópico de tráfego no qual os semáforos controlados por agentes foram adicionados externamente, comunicando-se com o ambiente através de detectores.

O tempo médio para percorrer uma rota foi medido utilizando-se três diferentes rotas com o objetivo de comparar as situações nas quais os semáforos não estão coordenados, estão coordenados de maneira fixa, e são livres para decidir quando e com que outros semáforos se coordenar. A última abordagem mostra-se mais flexível e dependente apenas de detectores instalados em cada cruzamento.

5.2 Coordenação Distribuída de Semáforos utilizando Mediação Cooperativa

Neste modelo, o problema de coordenação dos semáforos foi abordado como um meio-termo entre uma coordenação completamente autônoma com comunicação implícita (como o modelo da Seção 5.1) e uma solução centralizada clássica. A coordenação de semáforos é abordada como um problema de otimização em tempo real (*on-line*) para a utilização do algoritmo OptApo, apresentado na Seção 4.3. O objetivo deste modelo é duplo: propor um novo modelo de coordenação de semáforos que reduza a necessidade de intervenções externas ao sistema e testar o algoritmo OptApo em um ambiente dinâmico, já que este foi testado apenas no problema estático de coloração de grafos (MAILLER; LESSER, 2004).

5.2.1 Descrição

Assim como em (MAILLER; LESSER, 2004), aqui é considerado apenas o caso em que cada agente é associado a uma variável. No entanto, neste cenário, as variáveis do

problema são as direções de coordenação para cada semáforo e o domínio para todas as variáveis é dado pelos dois valores possíveis de coordenação: $D = \{\text{Norte-Sul/Sul-Norte (NS/SN), Leste-Oeste/Oeste-Leste (LO/OL)}\}$.

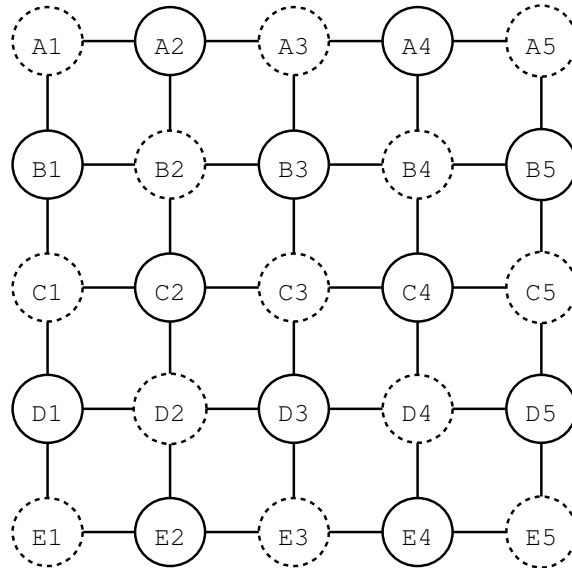


Figura 5.5: Uma rede com 25 intersecções (os nós com círculos pontilhados executam o plano SN/NS e os com linha cheia executam o plano LO/OL).

As restrições desse problema surgem com o fato de que cada semáforo, localizado em algum nó do grafo, pode apenas estar coordenado em um sentido a cada momento. Foi usado um mapa diferente do usado no modelo anterior (Seção 5.1) e também outro tipo de simulação. No modelo utilizado como cenário desta abordagem, o semáforo pode estar coordenado para os sentidos Norte e Sul ao mesmo tempo, visto que o mapa é uma grade (Figura 5.5) na qual as seções de rua têm tamanhos iguais. Coordenação aqui, indica que essa direção terá uma parcela maior de tempo de verde. Um conflito ocorre quando dois vizinhos querem priorizar direções diferentes.

Como uma medida de eficiência do sistema coordenado, geralmente busca-se otimizar uma combinação dos tempos de parada e o tempos de defasagem ou medidas associadas à ocupação das vias na rede. Nesta abordagem, utiliza-se uma função de custo que tem como parâmetros o número de veículos entrando na intersecção e o fato de o semáforo estar ou não coordenado com o seu vizinho, na direção a ser priorizada.

Algoritmo 5.1 Calcula o custo global F_i

```

se ( $VeiculosEntrando(NS) \geq (VeiculosEntrando(LO))$ ) então
  para todo  $j$  na direção Norte/Sul  $\in$  good_list do agente  $i$  faça
     $F_i \leftarrow F_i + f(x_i, x_j)$ 
  fim para
senão
  para todo  $j$  na direção Leste/Oeste  $\in$  good_list do agente  $i$  faça
     $F_i \leftarrow F_i + f(x_i, x_j)$ 
  fim para
fim se

```

Algoritmo 5.2 Calcula a relação de custo $f(x_i, x_j)$

```

se ( $v_i$  é a direção com o maior número de veículos na entrada) então
  se ( $v_i == v_j$ ) então
     $f(x_i, x_j) \leftarrow 0$ ;
  senão
     $f(x_i, x_j) \leftarrow \frac{VeiculosVindoDe(x_j)}{TotalDeVeiculosEntrando()};$ 
  fim se
senão
   $f(x_i, x_j) \leftarrow 2 * \frac{VeiculosVindoDe(x_j)}{TotalDeVeiculosEntrando()};$ 
fim se

```

A função do custo global de cada agente i , calculada pelo Algoritmo 5.1, depende da direção com maior número de veículos. Se há mais veículos sendo inseridos nos sentidos Norte e Sul, o agente fará o cálculo do seu custo global (F_i) considerando apenas a soma das relações funcionais com seus vizinhos nestas direções. Uma relação funcional é associada a dois vizinhos e as direções que estão priorizando no momento.

O custo de cada relação entre dois agentes ($f(x_i, x_j)$) é calculado utilizando-se o Algoritmo 5.2, considerando-se a direção com o maior número de veículos entrando e o estado atual de coordenação dos agentes. O custo da relação pode ter valores diferentes, de acordo com as seguintes situações possíveis:

Situação I se ambos os agentes na relação estão executando planos coordenados entre si, ou seja, $v_i = v_j$, e esse plano prioriza a direção em que entram o maior número de veículos, então, este relacionamento possui um custo zero ($f(x_i, x_j) = 0$);

Situação II se o agente i esta priorizando a direção com o maior número de veículos na entrada mas o agente vizinho j não está sincronizado nesta direção, então o custo dessa relação ($f(x_i, x_j)$) é dado pelo número de veículos vindos do nó x_j (calculado pela função $VeiculosVindoDe(x_j)$), dividido pela soma dos veículos de todos os agentes vizinhos que o nó x_i tem em sua *good_list* (calculado pela função $TotalDeVeiculosEntrando()$)

Situação III se a direção priorizada pelo agente x_i não é a direção com o maior número de veículos entrando, o custo será o dobro do custo da situação anterior, já que o agente não está utilizando o melhor plano para a situação corrente do tráfego.

O objetivo do problema de otimização é coordenar os semáforos para minimizar o custo global (soma de todos os custos dos relacionamentos). O custo de cada relação (f) é sempre um valor entre 0 e 2, como mostrado nas situações acima.

O cenário de tráfego e o algoritmo OptAPO foram implementados usando a ferramenta de simulação SeSAM (*Shell for Simulated Agent Systems*), (KLÜGL; PUPPE, 1998), que fornece uma plataforma genérica para modelagem e experimento para simulações baseadas em agentes. Nesta ferramenta o comportamento do agente é modelado como uma máquina de estados finitos (MEF). No caso do agente semáforo, há quatro estados possíveis: *Inicializa, Verifica a Visão do Agente, Media e Escolhe Solução*.

Inicialmente, são atribuídos externamente planos de coordenação para os agentes. A prioridade de mediação de cada agente é calculada pelo número de veículos que estão entrando na intersecção controlada pelo agente. Durante a inicialização, o agente inclui

todos os seus vizinhos diretos e a si mesmo na sua *good_list*. Depois da fase de inicialização, todos os agentes estão habilitados para ir ao estado *Verifica a Visão do Agente*, no qual avaliam suas intenções de mediação (ativa ou passiva) e decidem se conduzirão uma mediação passiva ou ativa. Os agentes que sinalizaram uma mediação ativa e têm a maior prioridade, vão para o estado *Media* e iniciam um processo de mediação. O agente mediador inicia a execução do estado *Escolhe Solução*, depois disso ele avalia todos os agentes que estão na sua *good_list* e tenham preenchido sua lista de preferências. No estado *Escolhe Solução* o mediador executa uma busca do tipo *branch-and-bound*, vista no Capítulo 4, Seção 4.2.1, para minimizar o custo para os agentes que não estão na sessão de mediação. Os agentes que estão na *good_list* do mediador e não estão envolvidos em outro processo de mediação são os agentes que estão na sessão. Após escolher a melhor solução e atribuir novos valores para os agentes que estão na sessão de mediação, retorna para o estado *Verifica a Visão do Agente*.

5.2.2 Exemplo

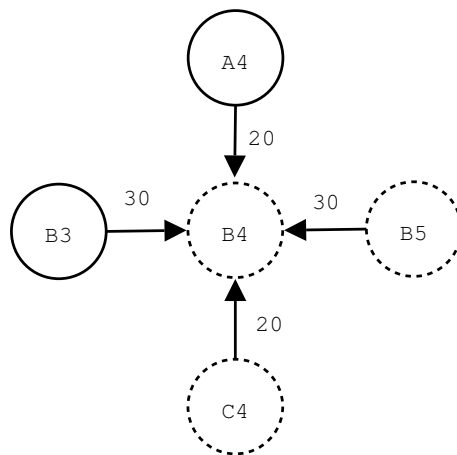


Figura 5.6: Uma configuração particular para o subsistema do nó B4.

Como exemplo, considere o grafo na Figura 5.5 que representa uma rede de tráfego com 25 nós (intersecções) e arestas associadas (relações funcionais). Os círculos pontilhados representam os agentes que estão executando o plano que prioriza o sentido Norte-Sul/Sul-Norte e círculos com linha representam agentes que estão com o plano de priorização do sentido Leste-Oeste/Oeste-Leste. A Figura 5.6 representa uma sub-rede com uma configuração particular de estados. Com o objetivo de simplificar a figura, as linhas de saída dos nós não estão representadas graficamente.

Considere o nó B4 na Figura 5.6, segundo o algoritmo OptApo, o valor de sua prioridade em relação aos outros nós é dado pelo número total de veículos vindos dos nós: A4, B3, B5 e C4. No caso da figura exemplo, a prioridade de B4 seria $p_{B4} = 100$ já que as entradas do nó B4 são: 20 veículos de A4, 20 veículos de C4, 30 veículos de B5 e 30 veículos de B3. Após a inicialização, os parâmetros para o agente B4 têm os seguintes valores:

- $p_i = 100$;
- $v_i = \text{NS/SN}$ (sentido Norte/Sul);

- $m_i = \mathbf{ativa}$ (devido a inicialização);
- $mediate = \mathbf{nenhuma}$;
- $good_list = \{A4, B3, B4, B5, C4\}$ (vizinhos diretos de B4 mais ele mesmo);
- $agent_view = \{A4, B3, B5, C4\}$ (vizinhos diretos de B4) ;

Usando o Algoritmo 5.1, o agente B4 compara a soma do número de veículos vindos das direções Norte e Sul com a soma dos veículos vindos das outras direções. No caso, há 40 veículos vindos do Norte e Sul e 60 do Leste e Oeste, indicando uma maior concentração de veículos no sentido NS/SN. A partir destes dados, o agente B4 considera a sua relação com os nós B3 e B5, não considerando os vizinhos A4 e C4 já que esses estão no sentido com menos veículos.

A relação de B4 com B3 e B5 se enquadram na “Situação III” (descrita na Subseção 5.2.1), visto que B4 não está priorizando a direção com o maior tráfego. Seguindo o Algoritmo 5.2:

- A relação entre B4 e B5 tem mesmo valor que a relação entre B4 e B3:

$$f(B4, B5) = f(B4, B3) = 2 * \frac{30}{100}.$$

- O valor do custo global para B4 é a soma dos custos das relações funcionais:

$$F_i = 2 * \frac{30}{100} + 2 * \frac{30}{100} = 1, 2.$$

- Como B4 tem o custo global com o valor maior que o valor ótimo, $F_i^* = 0$, B4 conduz uma sessão de mediação *ativa* ou *passiva*.

Sendo B4 o agente com a maior prioridade da sua lista, ele começa uma sessão de mediação ativa com os agentes A4, B3, B5 e C4; atribuindo o valor de **ativo** para o m'_i de B4. Para fins de simplificação do exemplo, consideramos que todos os agentes que não estão representados na Figura 5.6 estão priorizando o sentido NS/SN e todos têm: 30 veículos vindo da direção Leste, 30 veículos vindo da direção Oeste, 10 veículos vindo da direção Norte e 10 veículos vindo da direção Sul.

Na primeira tentativa para solucionar o problema, o agente tenta fazer uma mudança local no subsistema, com B4 mudando o plano para o sentido LO/OL. Essa mudança gera um custo diferente de zero, já que $f(B4, B5)$ enquadra-se na “Situação II” e B5 continuaria com NS/SN gerando um custo 0,3. Como a mudança local não leva a um custo ótimo, B4 envia a mensagem “avaliar?” para os agentes A4, B3, B5 e C4 e constrói uma tabela com a informação retornada pelos agentes na sessão (esses valores podem ser vistos na Tabela 5.2.2). Após receber uma mensagem “avaliar?” do agente B4, cada agente retorna informações sobre cada conflito gerado para cada um dos valores de coordenação. Por exemplo, o agente A4 informa que se seu valor d mudar para o plano NS/SN, o que levará a um conflito com os agentes A3 e A5 e que os custos $f(A4, A5)$ e $f(A4, A3)$ se enquadram na “Situação III” com um custo de 0,86. Essa informação é utilizada pelo agente mediador para minimizar o custo dos conflitos com agentes fora da sessão de mediação.

No próximo passo, o agente mediador (neste exemplo, o agente B4) executa uma busca *branch-and-bound* e encontra $F_i^* = 0$ para sua $good_list$. Usando essa solução, B4 muda a direção de B5, C4 e sua própria para LO/OL, deste modo os custos de $f(B4, B5)$ e $f(B4, B5)$ são iguais a zero. Uma situação de conflito entre C4 com C3 e C5 continua,

Tabela 5.2: Tabela de Preferências

nodo	Opções de Prioridade	Conflito	Custo
A4	NS/SN	A3	0,86
		A5	0,86
	LO/OL	A3	0,43
		A5	0,43
B3	NS/SN	B2	0,75
		B4	0,75
	LO/OL	B2	0,375
		B4	0,375
B5	NS/SN	B4	1,2
	LO/OL	B4	0,6
C4	NS/SN	C3	0,75
		C5	0,75
	LO/OL	C3	0,375
		C5	0,375

já que C3 e C5 não estão na *agent_view* do agente B4, porém com um custo reduzido (o custo original era 0,75 e com a solução encontrada, foi reduzido pela metade, ficando em 0,375, que pode ser observado na Tabela 5.2.2). Os conflitos causados por agentes fora do processo de mediação, fazem com que o agente B4 inclua os agentes C3 e C5 nas listas *agent_view* e *good_list*, já que B4 tem relação de vizinhança com C4. Essas adições criam relações artificiais (também chamadas de “links”) entre agentes e apenas acontece se o agente mediador está conduzindo uma mediação ativa e tem a maior prioridade. Depois da mediação de B4, o processo do OptApo continua até que cada um dos agentes tenha encontrado uma solução global com o menor número de restrições violadas.

5.2.3 Cenário e Experimentos

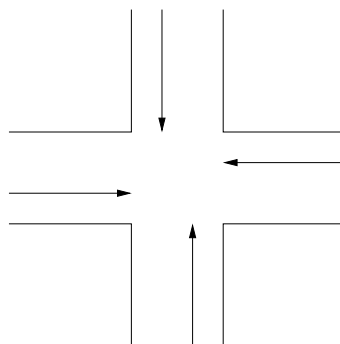


Figura 5.7: Movimentos de tráfego possíveis nos nodos com 8 faixas.

O cenário utilizado é uma grade quadrada de 5x5 com um semáforo (agente) em cada junção, representado na Figura 5.5, somando um total de 25 nodos e 40 arestas ou segmentos de rua (que representam 40 relações funcionais entre as variáveis) cada um desses segmentos têm a capacidade de 30 veículos por ciclo (em cada direção de tráfego). Cada nodo tem as movimentações possíveis mostradas pela Figura 5.2.3. O número de veículos

inseridos no início de cada rua (em nodos localizados nas bordas do mapa) é determinado por uma distribuição Normal com média μ e desvio padrão σ . Cada vez que um veículo atinge um nodo de borda ele é removido da simulação.

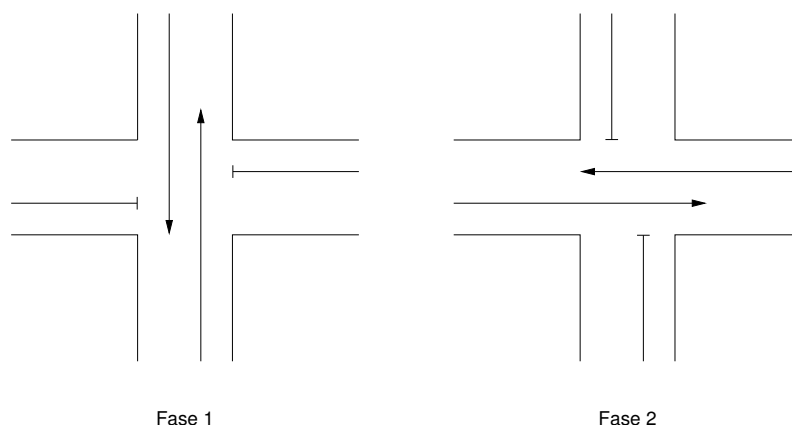


Figura 5.8: Movimentos de tráfego das fases dos semáforos neste cenário.

Geralmente, os semáforos têm um conjunto de planos semaforicos (para condições de tráfego diferentes ou diferentes períodos do dia). Nos experimentos, foram considerados apenas dois planos, cada um dando mais tempo de verde para uma direção. Esses planos semaforicos têm duas fases, os movimentos das duas fases podem ser vistas na Figura 5.2.3. Todos os planos semaforicos têm um tempo de ciclo de 90 segundos e duas fases: uma de 30s e outra de 60s. Como a menor fatia de tempo é de 30 segundos, o menor tempo considerado nas simulações é um terço de ciclo (30 segundos) e os gráficos dos resultados mostram esta unidade como um passo de simulação (*time step*). Restrições de velocidade fazem com que apenas 10 veículos (em cada direção) possam atravessar um cruzamento num intervalo de 30 segundos.

No início das simulações, os agentes A1, A3, A5, B2, B4, C1, C3, C5, D2, D4, E1, E3 e E5 (Figura 5.5) executam o plano $d = NS/SN$, enquanto os demais executam o plano $d = LO/OL$. Esta configuração inicial faz com que todos os agentes iniciem com vizinhos de planos diferentes do dele.

Foram simulados quatro casos:

Caso I o padrão do volume de tráfego é gerado uma vez e não muda durante a simulação;

Caso II uma mudança no volume de tráfego é inserida artificialmente no tempo 400 para testar o reinício das mediações;

Caso III duas mudanças no volume de tráfego são inseridas;

Caso IV situação com coordenação fixa (sem mediação);

5.2.4 Resultados

No caso I, não há mudanças no volume de tráfego durante toda a simulação, ou seja, os veículos são inseridos de acordo com uma única distribuição Gaussiana ($\mu = 8$ e $\sigma = 2$). Nessa simulação, espera-se que os agentes iniciem a mediação e eventualmente cheguem a uma configuração com o custo mínimo. O custo global (soma dos custos f de cada relacionamento) é mostrado na Figura 5.9. A simulação inicia com um custo alto (25) e durante os primeiros 120 passos de simulação, a mediação vai sendo executada e o

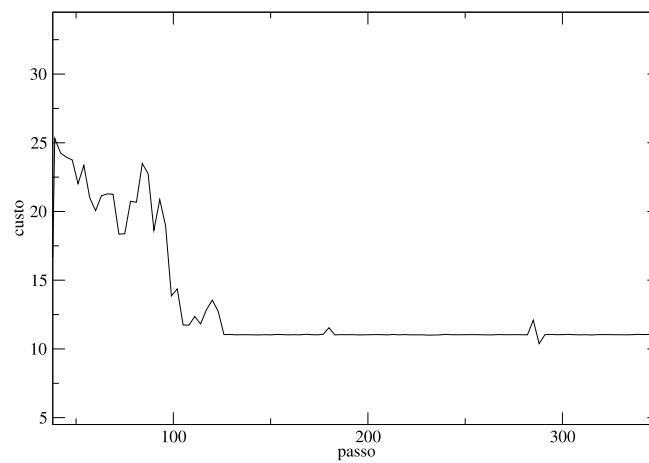


Figura 5.9: Cenário com volume de tráfego constante: custo ao longo da simulação.

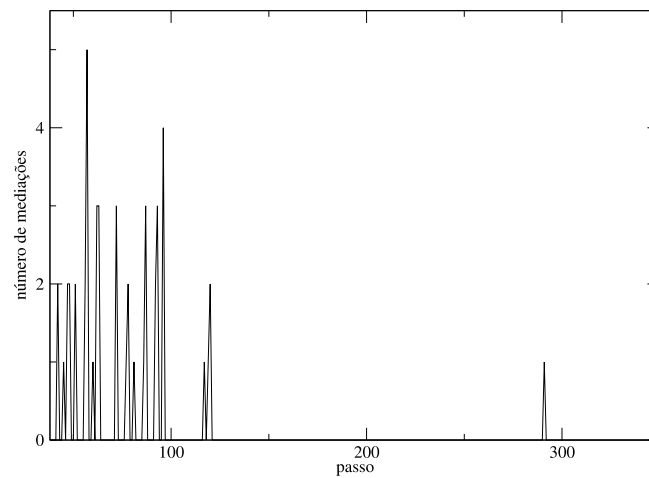


Figura 5.10: Cenário com volume de tráfego constante: número de mediações por ciclo.

custo vai diminuindo, exceto por alguns picos. Esses picos ocorrem devido à parada dos veículos no semáforo, aumentando a fila e conseqüentemente os custos. Durante o tempo de verde, as filas diminuem (não necessariamente acabam). Esse padrão ocorre devido ao tempo de ciclo do plano semafórico.

Os mediadores determinam os planos para cada agente e o custo estabiliza em torno de 12. O custo não pode ter um valor igual a zero, já que não é possível cada agente estar coordenado com todos os seus vizinhos ao mesmo tempo. Após o período de estabilização, apenas pequenas mudanças ocorrem no valor do custo, mas essas oscilações estão relacionadas à função Gaussiana de inserção de veículos e não causam novas mediações, já que o volume de tráfego não muda o suficiente para que seja necessária a mudança de

planos (como pode ser observado analisando o Algoritmo 5.1).

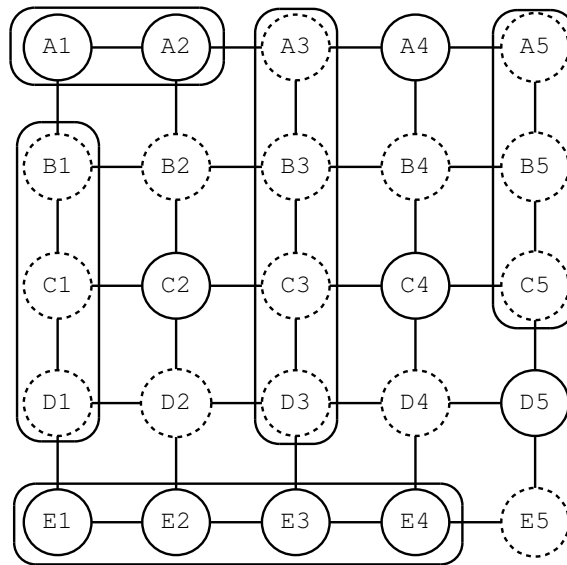


Figura 5.11: Grupos formados após a mediação.

Após a mediação, notam-se blocos de coordenação no mapa da Figura 5.11, a situação final difere muito da situação inicial mostrada na Figura 5.5: os nodos B1, C1, D1; A3, B3, C3, D3; e A5, B5, C5 têm planos coordenados para NS/SN, enquanto os nodos A1 e A2; e E1, E2, E3, E4 estão coordenados para o sentido LO/OL. O gráfico da Figura 5.10 mostra que poucos agentes conduziram mediações ativas, isso é um fato importante e positivo já que o número de passos necessários para alcançar um estado estável depende de quantos agentes necessitam mediar e se os mediadores têm informação suficientes sobre o sistema.

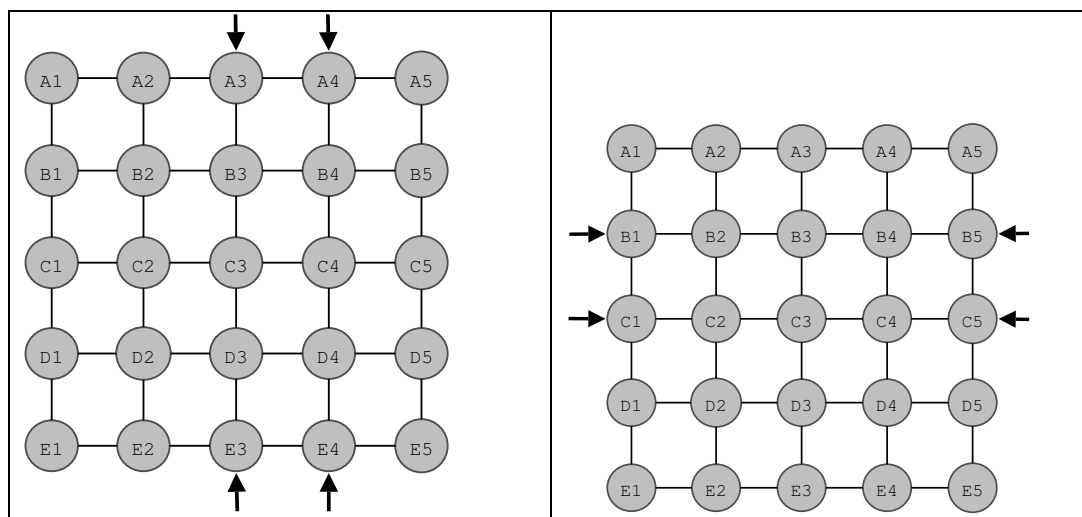


Figura 5.12: Mudança na inserção de veículos: Caso II (esquerda) e Caso III (direita).

No caso II, foi inserida artificialmente uma mudança no volume de tráfego no passo 400. Como mostra a Figura 5.12 (esquerda), as mudanças se localizam nas fontes dos nodos A3, A4, E3 e E4, os quais inserem mais veículos no sentido NS/SN. A nova distribuição gaussiana nesses nodos tem $\mu = 25$ e $\sigma = 1$. As demais fontes (demais nodos

localizados nas bordas do grafo) mudam seus valores para $\mu = 3$ e $\sigma = 1$. Sendo assim, espera-se que o início da simulação o comportamento seja similar ao observado no “Caso I” (quando o volume de tráfego é igual ao do “Caso I”). Após a mudança no passo 400, espera-se que o custo aumente novamente e que novos processos de mediação sejam iniciados com o objetivo de reduzir o custo.

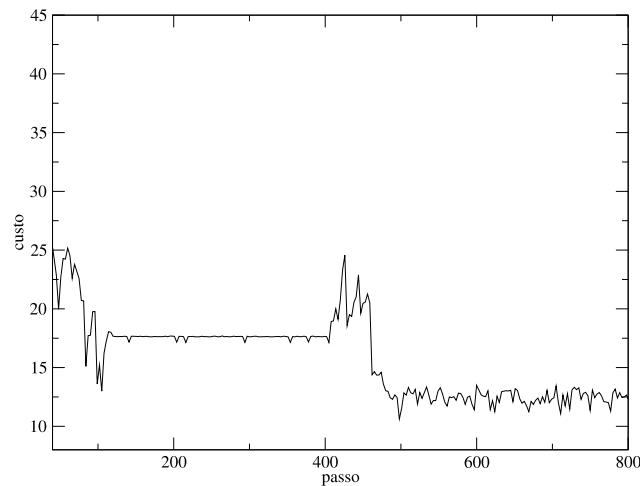


Figura 5.13: Situação com uma mudança no volume de tráfego no passo de simulação 400: custo ao longo da simulação.

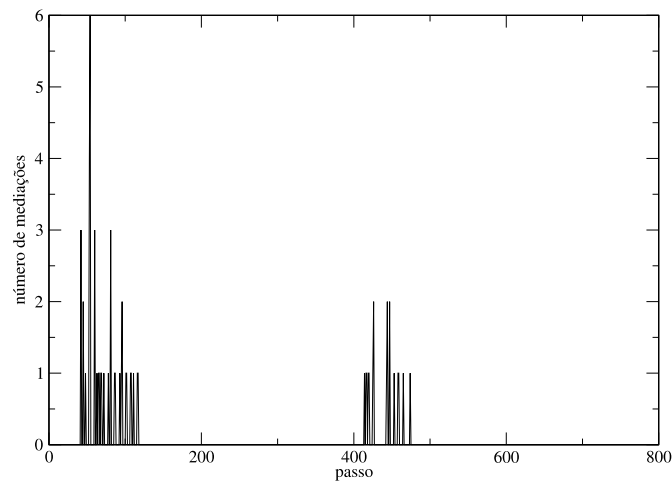


Figura 5.14: Situação com uma mudança no volume de tráfego no passo de simulação 400: número de mediações por ciclo.

De acordo com os resultados mostrados na Figura 5.13, os agentes encontraram uma solução com custo aproximadamente 18, para o fluxo de tráfego inicial. Como esperado,

após a mudança de fluxo no passo 400, o custo aumenta, fazendo com que os agentes iniciem novos processos de mediação. A nova solução é encontrada no passo 500, Figura 5.14, gerando um custo global 12. A nova solução faz com que as ruas “3” (agentes: A3, B3, ..., E3) e “4” (agentes: A4, B4, ..., E4) fiquem coordenadas para o sentido NS/SN, mostrando a adaptação às mudanças sofridas no número de veículos inseridos dos nodos A3, A4, E3 e E4.

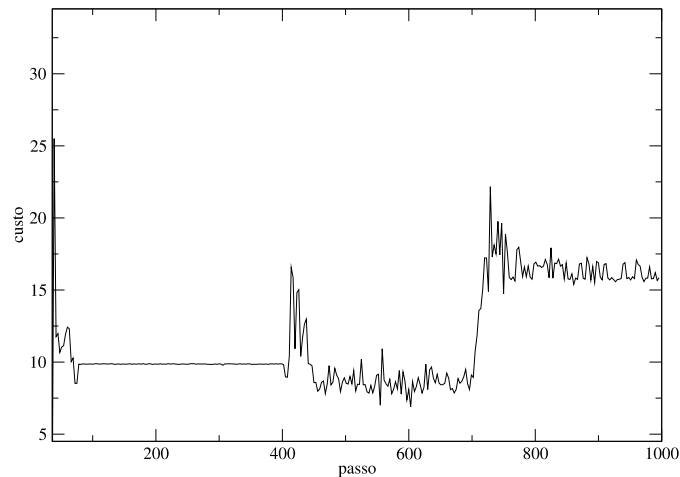


Figura 5.15: Situação com duas mudanças no volume de tráfego (nos passos 400 e 700): custo ao longo da simulação.

No caso III são inseridas duas mudanças no volume de tráfego, a primeira no passo 400 e a segunda no tempo 700. A primeira, ocorre do mesmo modo que no cenário anterior, o lado direito da Figura 5.12 mostra a segunda mudança: o número de veículos inseridos nos nodos B1, C1, B5 e C5 é determinado por uma distribuição Normal com $\mu = 25$ e $\sigma = 1$.

A Figura 5.15 mostra as alterações no custo causadas pelas duas mudanças no volume. Antes da primeira mudança, há uma estabilidade encontrada. Após a primeira mudança, verifica-se uma elevação no custo e o processo de mediação é reiniciado, Figura 5.16. Na segunda mudança, o custo eleva-se novamente, o processo de mediação reinicia e atinge-se uma solução na qual os agentes das ruas “B” (agentes: B1, B2, ..., B5) e “C” (agentes: B1, B2, ..., B5) estão coordenados para o sentido LO/OL.

O último experimento, caso IV, foi uma simulação com coordenação fixa (sem mediação ou mudanças de plano durante a simulação), a qual é mostrada na Figura 5.17 onde os nodos em cinza têm planos aleatórios e os demais estão coordenados para o plano LO/OL. Essa simulação serve como demonstração da ineficiência do método clássico no qual as ruas têm coordenação fixa e não há adaptação às mudanças no fluxo de tráfego. Durante os primeiros 400 ciclos o custo total é alto devido à situação inicial (igual a de todos os demais casos anteriores) e continua alto já que não há mudança nos planos.

Considere que os semáforos estão pré-programados para executar uma coordenação fixa priorizando LO/OL e envolvendo os agentes B1 a B5 e C1 a C5 (Figura 5.17, parte esquerda). No entanto, o volume de tráfego aumenta nos nodos A3, A4, E3 e E4 (nos quais a taxa de inserção foi modificada para $\mu = 25$ e $\sigma = 1$). Na Figura 5.18, nota-se

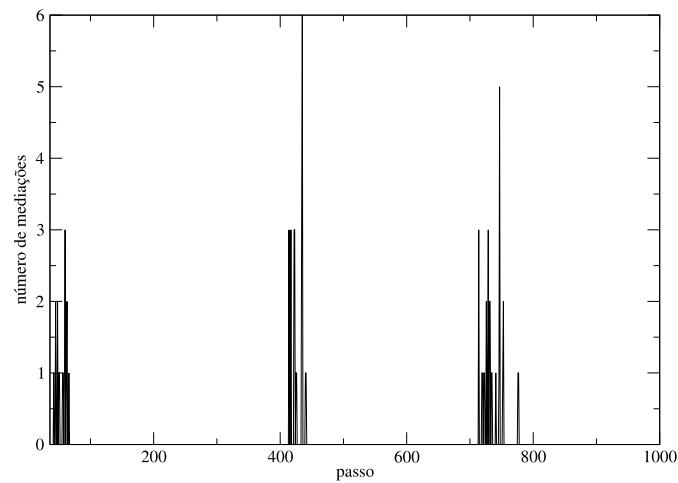


Figura 5.16: Situação com duas mudanças no volume de tráfego (nos passos 400 e 700): número de mediações por ciclo.

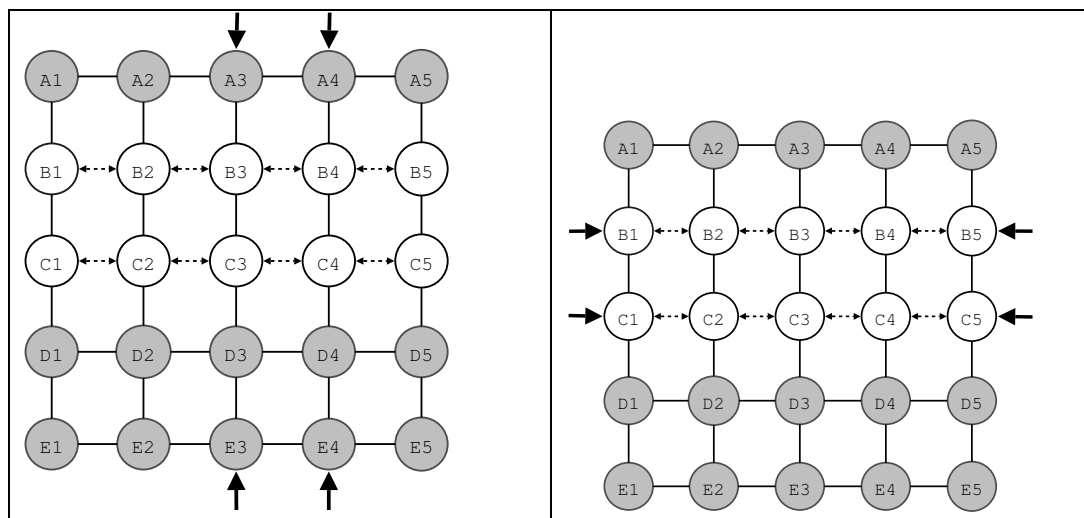


Figura 5.17: Caso com coordenação fixa: as setas em negrito indicam mudança na inserção de veículos para a variante 1 (esquerda) e variante 2 (direita).

que os custos aumentam mais ainda devido à coordenação fixa na direção LO/OL, que não está de acordo com essa situação de tráfego.

A coordenação fixa tem bom desempenho apenas onde a situação do tráfego está de acordo com sua programação, o que também é mostrado na mesma figura, após o passo 700. A partir do passo 700, o volume de tráfego está alto nos nós B1, C1, B5 e C5 e está de acordo com a coordenação fixa (Figura 5.17 a direita), havendo então a redução do custo total. No entanto, nota-se que esta redução é pior que em todos os casos onde há mediação, já que os demais semáforos executam planos aleatórios.

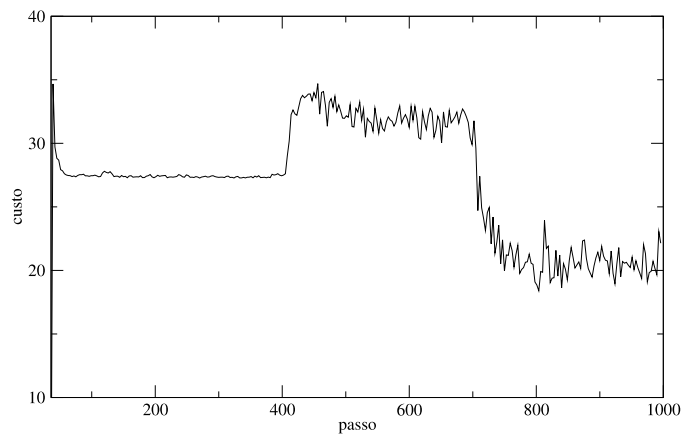


Figura 5.18: Caso com coordenação fixa.

5.2.5 Conclusões

Usando o algoritmo OptApo, a mediação cooperativa é executada pelo agente com mais informações sobre o sub-sistema (determinado pelo grafo de relacionamentos). Dentro desse sub-sistema, o mediador busca as mudanças que minimizarão o custo. Até o momento o algoritmo OptApo tinha sido testado apenas em ambientes estáticos (coloração de grafos), nos quais a solução era encontrada e o processo de mediação era terminado. No cenário do tráfego, a mediação além de reduzir os custos tem que lidar com as mudanças no padrão de tráfego, como foi mostrado nos experimentos. O processo de mediação demora alguns ciclos, especialmente nas primeiras mediações, já que a visão dos agentes ainda está muito limitada. Apesar do processo de mediação demorar alguns ciclos, ele já atua de forma a reduzir os custos.

5.3 Considerações sobre as abordagens

Ambas abordagens, apresentadas nas Seções 5.1 e 5.2, propõem soluções para coordenação de semáforos a fim de que haja uma melhoria na qualidade do tráfego na rede como um todo. O foco principal foi apenas a escolha de planos pré-programados que fossem mais adequados para determinada situação de tráfego e não sua otimização. O tipo de coordenação de ambas diferem, já que na primeira os agentes têm uma coordenação emergente e não há comunicação direta (por troca de mensagens), o que ocorre na segunda. A segunda abordagem tem uma coordenação explícita e possui troca de mensagens direta. Mesmo assim, não há um controle central agentes que possuem sempre o papel de controladores.

Qualitativamente, em ambas abordagens, quando os agentes são livres para decidir sua coordenação, o sistema comporta-se quase como se houvesse um agente externo controlando os demais. Os experimentos mostram que a sincronização é atingida sem monitoramento, indicando a eficiência de ambos os métodos.

6 CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho foram apresentados alguns conceitos básicos de coordenação de semáforos, *Swarm Intelligence* e problemas de otimização de restrições. Esses conceitos são importantes para a compreensão do funcionamento e dos objetivos das abordagens propostas.

Algumas abordagens para a redução do congestionamento nas vias urbanas foram apresentadas, sendo que a seleção de planos semaforicos foi o foco principal. Há uma clara necessidade de abordagens mais flexíveis onde as preferências de coordenação dos semáforos não precisem ser explicitamente programadas. Os experimentos realizados servem como validação inicial dos modelos e há uma necessidade de realizar mais experimentos, sob condições de tráfego mais próximas da real possível e em outros cenários. Esse trabalho é um passo adiante no trabalho de mais longo prazo para testar diferentes abordagens de coordenação de semáforos utilizando sistemas multiagentes.

Como trabalho futuro, pretende-se testar o modelo de coordenação por mediação cooperativa no simulador ITSUMO para poder fazer um comparativo tanto do desempenho computacional quanto melhoria nas condições de tráfego, em relação ao modelo inspirado em insetos sociais. Para simulações mais complexas, será preciso a implementação de alguns módulos ainda não existentes no simulador ITSUMO.

Além desses dois modelos, já está sendo analisada a implementação de um modelo que utilize processos de Markov parcialmente observáveis. A utilização de tais processos para controle de tráfego urbano é uma alternativa que vêm sendo avaliada, visto que os processos de decisão de Markov (*Markov Decision Processes*- MDP) servem como base para a solução de diversos problemas complexos e parcialmente observáveis (KALBLING; LITTMAN; CASSANDRA, 1998), que é o caso do problema da coordenação de semáforos.

REFERÊNCIAS

ABDALLAH, S.; DARWISH, N.; HEGAZY, O. Monitoring and Synchronization for Teamwork in GPGP. In: ACM SYMPOSIUM ON APPLIED COMPUTING, 2002. **Proceedings...** [S.l.: s.n.], 2002. p.288–293.

ANDRIOTTI, G. K. **Implementação de veículo tracejador para o simulador do projeto SISCOT**. 2001. Trabalho de Diplomação (Bacharelado em Ciência da Computação) — Instituto de Informática, UFRGS, Porto Alegre.

ANDRIOTTI, G. K. **Modelagem de motoristas e cenários de escolha de rota em simulações de tráfego veicular urbano**. 2004. Dissertação (Mestrado em Ciência da Computação) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre.

ANDRIOTTI, G. K.; BAZZAN, A. L. C. An Object-Oriented Microscopic Traffic Simulator. In: LATIN-AMERICAN CONFERENCE ON INFORMATICS, CLEI, 28., 2002, Montevideo, Uruguay. **Proceedings...** [S.l.: s.n.], 2002.

BAZZAN, A. L. C. Evolution of Coordination as a Metaphor for Learning in Multi-Agent Systems. In: WEISS, G. (Ed.). **DAI Meets Machine Learning**. Berlin: Springer-Verlag, 1997. p.117–136. (Lecture Notes in Artificial Intelligence, n. 1221).

BAZZAN, A. L. C. A Distributed Approach for Coordination of Traffic Signal Agents. **Autonomous Agents and Multiagent Systems**, [S.l.], v.10, n.1, p.131–164, March 2005.

BONABEAU, E.; SOBKOWSKI, A.; THERAULAZ, G.; DENEUBOURG, J.-L. Adaptive Task Allocation Inspired by a Model of Division of Labor in Social Insects. In: BIOC COMPUTING AND EMERGENT COMPUTATION, 1997. **Proceedings...** [S.l.]: World Scientific, 1997. p.36–45.

BONABEAU, E.; THERAULAZ, G.; DORIGO, M. **Swarm Intelligence: from natural to artificial systems**. [S.l.]: Oxford Univ Press, 1999.

CICIRELLO, V.; SMITH, S. Improved Routing Wasps for Distributed Factory Control. In: WORKSHOP ON ARTIFICIAL INTELLIGENCE AND MANUFACTURING: NEW AI PARADIGMS FOR MANUFACTURING, 2001. **Proceedings...** [S.l.: s.n.], 2001.

DECKER, K. S.; LESSER, V. R. Quantitative Modeling of Complex Computational Task Environments. In: NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, 11., 1993, Washington. **Proceedings...** [S.l.]: AAAI Press, 1993.

- DECKER, K. S.; LESSER, V. R. Designing a Family of Coordination Algorithms. In: INTERNATIONAL CONFERENCE ON MULTI-AGENT SYSTEMS, 1., 1995, San Francisco, CA. **Proceedings...** [S.l.]: AAAI Press, 1995.
- DIAKAKI, C.; PAPAGEORGIOU, M.; ABOUDOLAS, K. A multivariable regulator approach to traffic-responsive network-wide signal control. **Control Engineering Practice**, [S.l.], v.10, n.2, p.183–195, February 2002.
- FREUDER, E. C.; WALLACE, R. J. Partial constraint satisfaction. **Artificial Intelligence**, [S.l.], v.58, n.1–3, p.21–70, 1992.
- GOLOMB, S. W.; BAUMERT, L. D. Backtrack Programming. **J. ACM**, [S.l.], v.12, n.4, p.516–524, 1965.
- GORDON, D. The organization of work in social insect colonies. **Nature**, [S.l.], v.380, p.121–124, 1996.
- GORDON, D. **Ants at Work**: how an insect society is organized. [S.l.]: W.W. Norton & Company, 2000.
- GORDON, D.; PEREIRA, H. A trade-off in task allocation between sensitivity to the environment and response time. **Journal of Theoretical Biology**, [S.l.], v.208, p.165–184, 2001.
- GRASSÉ, P. P. La Reconstruction du nid et les Coordination Inter-Individuelles chez *Bellicositermes Natalensis* et *Cubitermes* sp. La théorie de la Stigmergie: essai d'interprétation du comportement des termites constructeurs. **Insectes Sociaux**, [S.l.], v.6, p.41–80, 1959.
- HORLING, B.; BENYO, B.; LESSER, V. Using Self-Diagnosis to Adapt Organizational Structures. In: INTERNATIONAL CONFERENCE ON AUTONOMOUS AGENTS, 5., 2001, Montreal. **Proceedings...** New York: ACM Press, 2001. p.529–536.
- KAELBLING, L. P.; LITTMAN, M. L.; CASSANDRA, A. R. Planning and acting in partially observable stochastic domains. **Artif. Intell.**, [S.l.], v.101, n.1-2, p.99–134, 1998.
- KLÜGL, F.; PUPPE, F. The Multi-Agent Simulation Environment SeSAM. In: WORKSHOP SiWis UNIVERSITY PADERBORN, 1998, Paderborn. **Proceedings...** [S.l.: s.n.], 1998.
- KUMAR, V. Algorithms for Constraints Satisfaction problems: a survey. **AI Magazine**, [S.l.], v.13, n.1, p.32–44, 1992.
- LEITE, J. G. M. **Engenharia de Tráfego**: métodos de pesquisa, características de tráfego, interseções e sinais luminosos. São Paulo: Companhia de Engenharia de Tráfego - CET, 1980.
- LESSER, V. Reflections on the Nature of Multi-Agent Coordination and Its Implications for an Agent Architecture. **Autonomous Agents and Multi-Agent Systems**, [S.l.], v.1, p.89–111, January 1998.
- LOWRIE, P. The Sydney Coordinate Adaptive Traffic System – Principles, Methodology, Algorithms. In: INTERNATIONAL CONFERENCE ON ROAD TRAFFIC SIGNALLING, 1982, Sydney, Australia. **Proceedings...** [S.l.: s.n.], 1982.

MAILLER, R.; LESSER, V. A Mediation Based Protocol for Distributed Constraint Satisfaction. In: INTERNATIONAL WORKSHOP ON DISTRIBUTED CONSTRAINT REASONING, 4., 2003, Acapulco, Mexico. **Proceedings...** [S.l.: s.n.], 2003. p.49–58.

MAILLER, R.; LESSER, V. Solving Distributed Constraint Optimization Problems Using Cooperative Mediation. In: INTERNATIONAL JOINT CONFERENCE ON AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS, 3., 2004, New York. **Proceedings...** New York: IEEE Computer Society, 2004. p.438–445.

MALONE, T.; CROWSTON, K. The Interdisciplinary Study of Coordination. **ACM Computing Surveys**, [S.l.], v.26, n.1, p.87–119, 1994.

NAGEL, K.; SCHRECKENBERG, M. A cellular automaton model for freeway traffic. **Journal de Physique I**, [S.l.], v.2, p.2221, 1992.

OLIVEIRA, D.; BAZZAN, A. L. C.; LESSER, V. Using Cooperative Mediation to Coordinate Traffic Lights: a case study. In: INTERNATIONAL JOINT CONFERENCE ON AUTONOMOUS AGENTS AND MULTI AGENT SYSTEMS, AAMAS, 4., 2005. **Proceedings...** New York: IEEE Computer Society, 2005.

OLIVEIRA, D.; FERREIRA, P. R.; BAZZAN, A. Reducing Traffic Jams with a Swarm-based Approach for Selection of Signal Plans. In: INTERNATIONAL WORKSHOP ON ANT COLONY OPTIMIZATION AND SWARM INTELLIGENCE, ANTS, 4., 2004, Berlin, Germany. **Proceedings...** Berlin: Springer-Verlag, 2004. p.416–417. (LNCS, v.3172).

OLIVEIRA, D.; FERREIRA, P. R.; BAZZAN, A. L. C. A Swarm Based Approach for Task Allocation in Dynamic Agents Organizations. In: INTERNATIONAL JOINT CONFERENCE ON AUTONOMOUS AGENTS AND MULTI AGENT SYSTEMS, AAMAS, 3., 2004, New York, USA. **Proceedings...** New York: IEEE Computer Society, 2004. v.3, p.1252–1253.

ROBERTSON, D. I.; BRETHERTON, R. D. Optimizing Networks of Traffic Signals in Real Time – The SCOOT Method. **IEEE Transactions on Vehicular Technology**, [S.l.], v.40, n.1, p.11–15, February 1991.

ROBISON, G. E. Regulation of Division of Labor in Insect Societies. **Annual Review of Entomology**, [S.l.], v.37, p.637–665, 1992.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence: a modern approach**. [S.l.]: Prentice-Hall, 1995.

SCHULZ, D. J.; HUANG, Z.-Y.; ROBINSON, G. E. Effects of colony food shortage on behavioral development in honey bees. **Behavioral Ecological Sociobiology**, [S.l.], n.42, p.295–303, 1998.

SILVA, B. C.; BAZZAN, A. L. C.; ANDRIOTTI, G. K.; OLIVEIRA, D. ITSUMO: an intelligent transportation system for urban mobility. In: OPTIMIZATION OF URBAN TRAFFIC SYSTEMS, 2005, Guadalajara, Mexico. **Proceedings...** Berlin: Springer-Verlag, 2005. n.3473, p.224–235. (Lecture Notes in Computer Science – Innovative Internet Computing Systems 2004).

SILVA, B. C.; BAZZAN, A. L. C.; OLIVEIRA, D.; FERREIRA, P. R.; LOPES, F.; ANDRIOTTI, G. K.; PAPAGEORGIOU, T. M. Desenvolvimento de um Sistema Livre para Simulação de Tráfego Baseada em Modelos Microscópicos. In: WORKSHOP SOBRE SOFTWARE LIVRE, WSL, 5.; FÓRUM INTERNACIONAL DE SOFTWARE LIVRE, FISL, 5., 2004. **Anais...** [S.l.: s.n.], 2004.

THERAULAZ, G.; BONABEAU, E.; DENEUBOURG, J. Response Threshold Reinforcement and Division of Labour in Insect Societies. In: ROYAL SOCIETY OF LONDON SERIES B – BIOLOGICAL SCIENCES, 1998. **Proceedings...** [S.l.: s.n.], 1998. v.265, p.327–332.

TRANSYT-7F. **TRANSYT-7F User's Manual**. [S.l.]: Transportation Research Center, University of Florida, 1988.

WAGNER, T.; GURALNIK, V.; PHELPS, J. A key-based coordination algorithm for dynamic readiness and repair service coordination. In: INTERNATIONAL JOINT CONFERENCE ON AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS, 2., 2003, New York. **Proceedings...** New York: ACM Press, 2003.

WAGNER, T.; LESSER, V. R. Design-to-Criteria Scheduling: real-time agent control. In: SPRING SYMPOSIUM ON REAL-TIME AUTONOMOUS SYSTEMS, AAAI, 2000, Stanford, CA. **Proceedings...** [S.l.: s.n.], 2000. p.89–96.

WILSON, E. O. **Sociobiology: the new synthesis**. [S.l.]: Harvard Univ Press, 2000.

APÊNDICE A PLANOS SEMAFÓRICOS DO CENÁRIO UTILIZADO NO MODELO DE INSETOS SOCIAIS

Semáforo 0

Plano 1	($T_{\text{Offset}}: 0s$)	Plano 2	($T_{\text{Offset}}: 16s$)
0-75	Esquerda – Direita	0-75	Cima - Baixo
76-100	Cima - Baixo	76-100	Esquerda – Direita

--	--

Figura A.1: Estágios e duração das fases do Semáforo 0

Semáforo 1

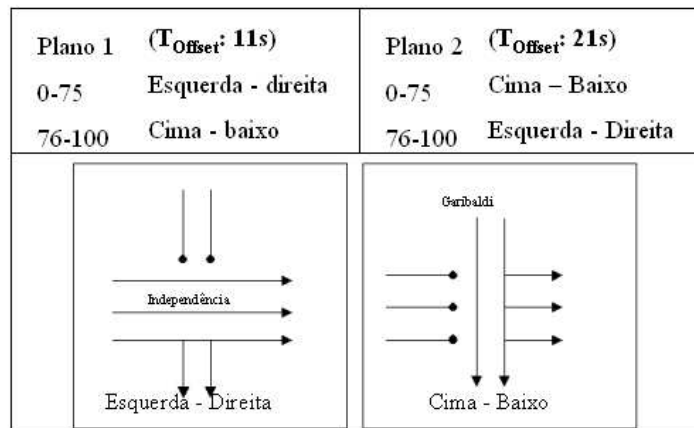


Figura A.2: Estágios e duração das fases do Semáforo 1

Semáforo 2

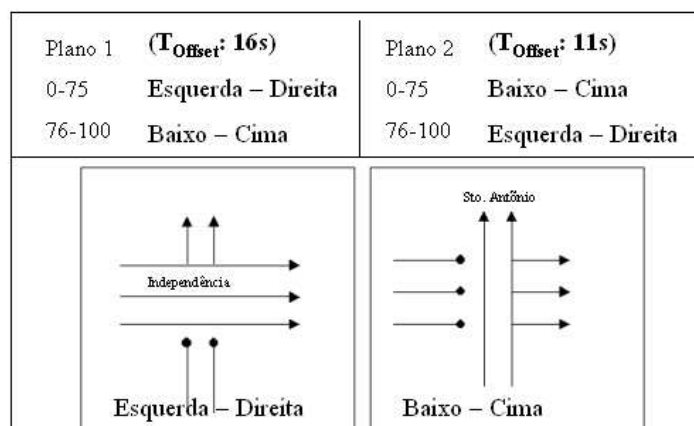


Figura A.3: Estágios e duração das fases do Semáforo 2

Semáforo 3

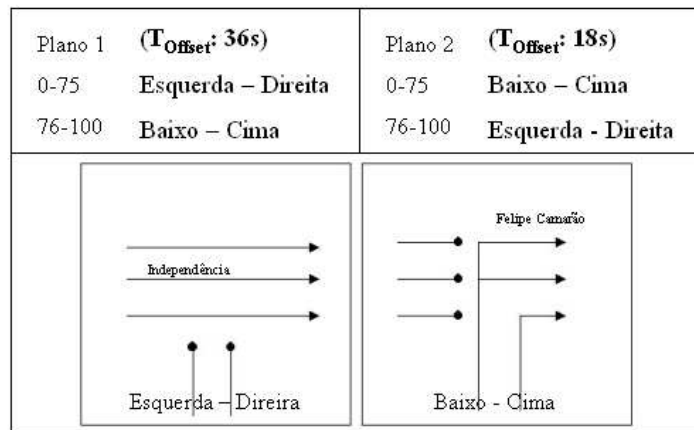


Figura A.4: Estágios e duração das fases do Semáforo 3

Semáforo 4

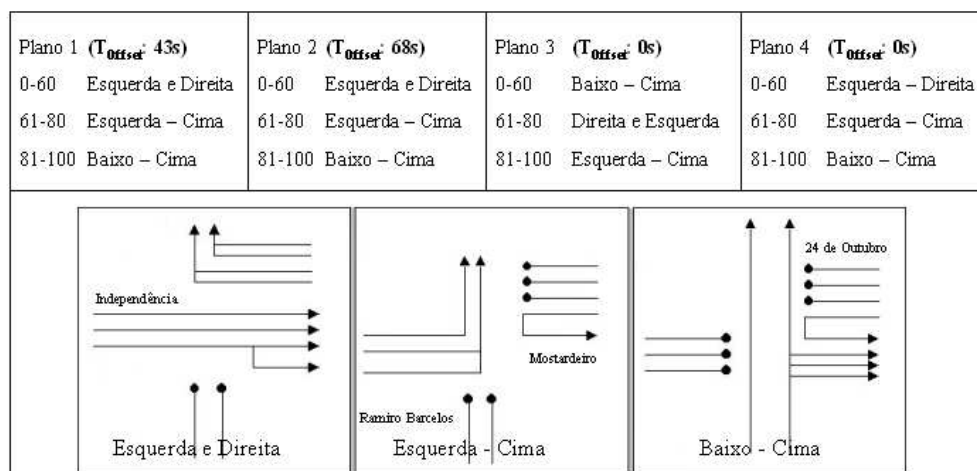


Figura A.5: Estágios e duração das fases do Semáforo 4

Semáforo 5

Plano 1 (T_{Offset} : 41s)	Plano 2 (T_{Offset} : 0s)	Plano 3 (T_{Offset} : 29s)
0-75 Esquerda – Direita	0-75 Cima e Baixo	0-75 Cima e Baixo
76-100 Cima e Baixo	76-100 Esquerda – Direita	76-100 Esquerda – Direita

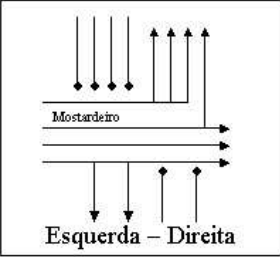
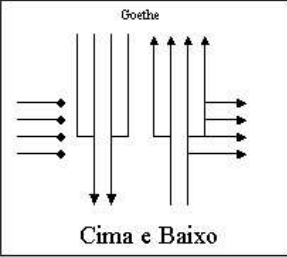
 <p>Esquerda – Direita</p>	 <p>Cima e Baixo</p>
---	--

Figura A.6: Estágios e duração das fases do Semáforo 5

Semáforo 6

Plano 1 (T_{Offset} : 56s)	Plano 2 (T_{Offset} : 59s)
0-75 Esquerda – Direita	0-75 Cima – Baixo
76-100 Cima – Direita	76-100 Esquerda – Direita

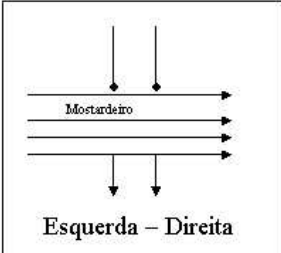
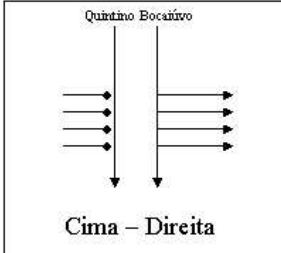
 <p>Esquerda – Direita</p>	 <p>Cima – Direita</p>
---	--

Figura A.7: Estágios e duração das fases do Semáforo 6

Semáforo 7

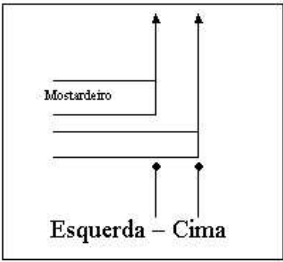
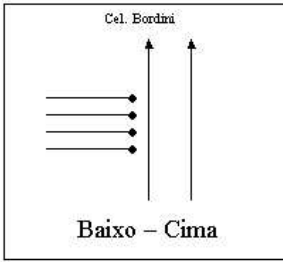
Plano 1 (T_{Offset}:62 s)	Plano 2 (T_{Offset}: 28s)
0-75 Esquerda – Cima	0-75 Baixo – Cima
76-100 Baixo – Cima	76-100 Esquerda – Cima
	

Figura A.8: Estágios e duração das fases do Semáforo 7

Semáforo 8

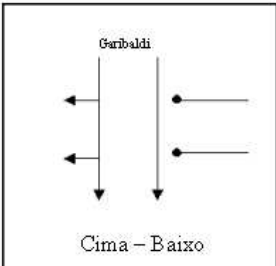
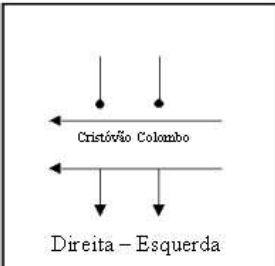
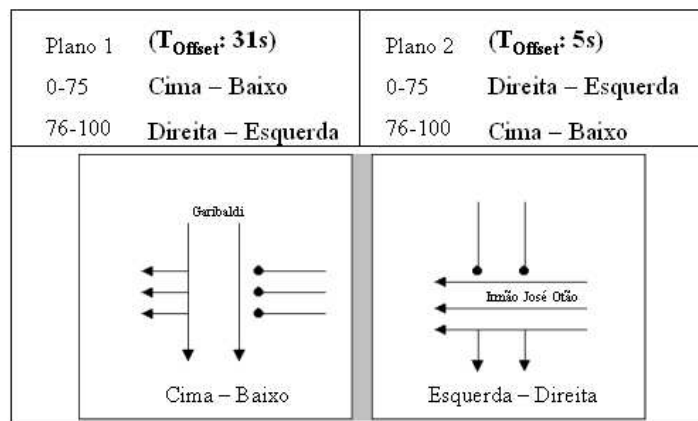
Plano 1 (T_{Offset}: 0s)	Plano 2 (T_{Offset}: 8s)
0-75 Cima – Baixo	0-75 Direita – Esquerda
76-100 Direita – Esquerda	76-100 Cima – Baixo
	

Figura A.9: Estágios e duração das fases do Semáforo 8

Semáforo 9



e

Figura A.10: Estágios e duração das fases do Semáforo 9

Semáforo 10

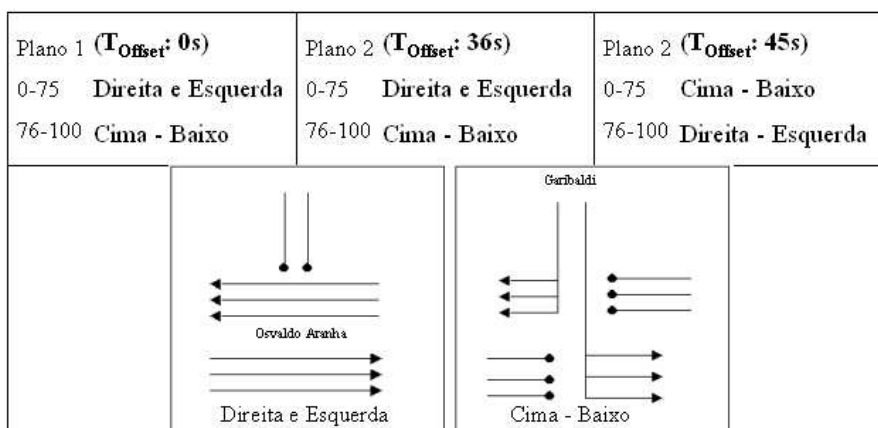


Figura A.11: Estágios e duração das fases do Semáforo 10

Semáforo 11

Plano 1 (T_{Offset} : 21s)	Plano 2 (T_{Offset} : 8s)	Plano 3 (T_{Offset} : 22s)
0-75 Direita – Esquerda	0-75 Cima e Baixo	0-75 Cima e Baixo
76-100 Cima e Baixo	76-100 Direita – Esquerda	76-100 Direita – Esquerda

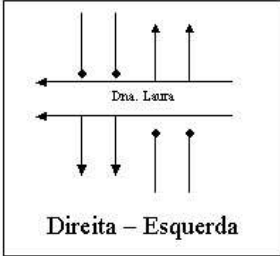
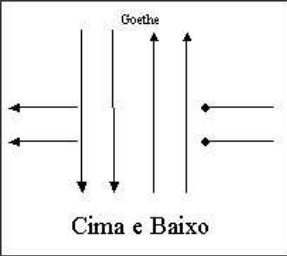
 <p>Direita – Esquerda</p>	 <p>Cima e Baixo</p>
---	--

Figura A.12: Estágios e duração das fases do Semáforo 11

Semáforo 12

Plano 1 (T_{Offset} : 12s)	Plano 2 (T_{Offset} : 25s)	Plano 3 (T_{Offset} : 4s)
0-75 Esquerda – Direita	0-75 Cima e Baixo	0-75 Cima e Baixo
76-100 Cima e Baixo	76-100 Esquerda – Direita	76-100 Esquerda – Direita

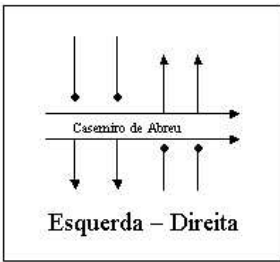
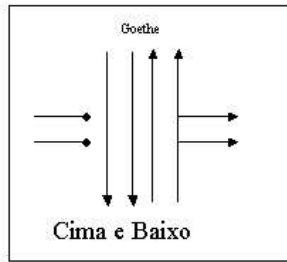
 <p>Esquerda – Direita</p>	 <p>Cima e Baixo</p>
---	--

Figura A.13: Estágios e duração das fases do Semáforo 12

Semáforo 13

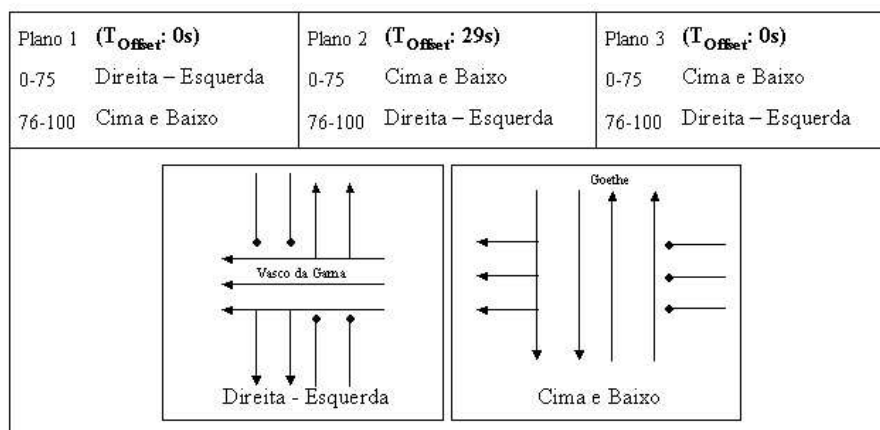


Figura A.14: Estágios e duração das fases do Semáforo 13

APÊNDICE B ALOCAÇÃO DINÂMICA DE TAREFAS INSPIRADA EM INSETOS SOCIAIS

B.1 Introdução

A estrutura organizacional de um sistema multiagente é um dos aspectos mais significativos para seu sucesso, (LESSER, 1998). A organização dos agentes depende dos objetivos do sistema, na percepção do ambiente e na configuração das atividades dos agentes e suas interações. Um problema é definir qual estrutura organizacional é mais adequada às necessidades do sistema.

Uma maneira simples de resolver esse problema é a definição estática da organização, que significa saber as necessidades do sistema e modelar uma organização apropriada. Sendo essa solução feita antes do início da resolução do problema, em um sistema instável ela se torna ineficiente.

Os SMAs precisam lidar com mudanças dinâmicas no sistema como variação no número de agentes, mudanças no ambiente e mudanças nos objetivos. O problema é como gerar a sua estrutura organizacional a partir de uma situação particular. A maioria dos trabalhos nesta área é restrita à adaptação de alguns aspectos específicos da organização ou da geração da estrutura. Cada uma dessas abordagens se mostra eficiente em cenários específicos mas não são uma solução genérica ao problema.

Uma abordagem que pretende ser mais generalista, baseada nas ferramentas TÆMS, *General Partial Global Planning* – GPGP e *Design-to-Criteria* – DTC e um sistema de auto-análise, foi proposta por (HORLING; BENYO; LESSER, 2001). Essa é uma abordagem de alto nível baseada na especificação prévia dos objetivos da organização, planejamento e escalonamento de tarefas (ou alocação de tarefas). Essa ferramenta também mostra bons resultados mas a questão sobre a eficiência da organização em termos de comunicação ainda está em aberto.

A abordagem proposta tem o objetivo de adaptar organizações em SMA utilizando mecanismos de atribuição de tarefas dos insetos sociais. Devido a sua independência de domínio, utiliza-se TÆMS (DECKER; LESSER, 1993) para representar as atividades dos agentes. O TÆMS será apresentado na Seção B.2, juntamente com o GPGP e o escalonador DTC. Na abordagem proposta utiliza-se uma estrutura de tarefas TÆMS para modelar as atividades e objetivos do sistema. O modo como as tarefas são alocadas a determinado pelo modelo de alocação inspirado em insetos sociais.

B.2 TÆEMS, GPGP e DTC

As ferramentas TÆEMS (DECKER; LESSER, 1993), GPGP (DECKER; LESSER, 1995) e DTC (WAGNER; LESSER, 2000) vêm sendo utilizadas como linguagem com domínio independente para a descrição de tarefas associadas a agentes, planejamento e atribuição de tarefas a agentes. A descrição TÆEMS permite a construção de um modelo de tarefas no qual a relação entre as tarefas é apresentada e fornecendo maneiras de modelar cenários onde as tarefas têm características associadas como tempo limite, custo, qualidade, etc.

As atividades do agentes são representadas como um grafo onde os nodos podem representar métodos executáveis com características de qualidade, custo e duração associadas. A qualidade de um grupo de tarefas depende de quais sub-tarefas e métodos são executadas. Por exemplo, a qualidade pode ser calculada com uma função de acumulação de qualidade (*Quality Accumulation Function* – QAF) como $sum()$ que indica que todas as tarefas na estrutura precisam ser executadas.

Utilizando essas ferramentas, é possível construir uma estrutura de tarefas para a resolução de um problema. A estrutura completa é chamada modelo objetivo do ambiente e é inacessível aos agentes. Cada agente tem uma visão limitada do problema que contém tarefas e relacionamentos.

Além disso, existem relações de coordenação utilizadas para modelar o efeito de uma tarefa sobre outra dentro do mesmo problema. GPGP e DTC fazem uma análise do processo modelado em TÆEMS e decidem a ordem de execução das tarefas para cada agente, de acordo com as restrições fornecidas (tempo limite, limitação de recursos, etc.).

B.3 O modelo de alocação de tarefas inspirado em insetos sociais

O modelo proposto é inspirado na alocação de tarefas dos insetos sociais e é aplicado a resolução de problemas dinâmicos modelados em TÆEMS. A seguir será apresentado o funcionamento deste modelo.

B.3.1 Estímulo

Como mencionado na Seção B.2, um método é um elemento em uma estrutura de tarefas TÆEMS que representa o que o agente pode fazer. Todos os métodos (tarefas que o inseto deve executar) na estrutura de tarefas tem distribuições probabilísticas de qualidade (q_j), custo (c_j) e duração (d_j) associadas. Esses valores são utilizados pelo agente para calcular o estímulo s_j de um método j , utilizando a Equação B.1.

$$s_j = \varphi * (\alpha * \hat{q}_j - \beta * \hat{c}_j - \gamma * \hat{d}_j + \beta + \gamma) + (1 - \varphi) * x_j \quad (\text{B.1})$$

Onde:

s_j estímulo associado ao método j ;

\hat{q}_j qualidade esperada e normalizada do método j ;

\hat{c}_j custo esperado e normalizado do método j ;

\hat{d}_j duração esperada e normalizada do método j ;

x_j estímulo associado à QAF relacionada ao método j ;

$\alpha, \beta, \gamma, \varphi$ constantes empregadas para atribuir diferentes prioridades para qualidade, custo e duração (a soma dessas constantes deve ser 1). Nos experimentos, as constantes α, β e γ têm valores iguais e $\varphi = 0,5$.

Tabela B.1: Constantes relacionadas a QAF

QAF	κ
SeqMax, Max, SeqMin, Min	0
SeqSum, Sum	0,01
ExactlyOne	-0,01

O estímulo s_j para cada método j é recalculado cada vez que um agente executa esse método. Essa atualização do estímulo é modelada para modelar a sucessão tarefas, discutida por Bonabeau et al. (BONABEAU; THRAULAZ; DORIGO, 1999). Em colônias insetos sociais, a execução de uma tarefa aumenta o estímulo das tarefas relacionadas a ela. Por exemplo, criar uma pilha de entulhos na entrada do ninho gera uma necessidade de limpeza. Na abordagem proposta, a execução de um método influencia o estímulo associado com todos os métodos da mesma tarefa TÆMS de acordo com as QAF relacionadas a essas tarefas.

$$x_j = x_j + \kappa \quad (\text{B.2})$$

Onde κ é a constante relacionada a QAF, como definida na Tabela B.1.

Considerando uma tarefa TÆMS com M métodos, quando um método k qualquer de M é executado, todos os demais métodos relacionados a ele por uma QAF terão o estímulo atualizado, utilizando a Equação B.2. Essa influência é recursiva para cada método da mesma tarefa. A constante κ é associada a cada tipo de QAF de modo a representar as diferenças no estímulo entre as diferentes QAFs.

Na Equação B.1 utiliza-se a constante φ para diferenciar a influência do estímulo associado à execução do método (qualidade, custo e duração) do estímulo relacionado à sucessão de tarefas (relacionado às QAFs).

A abordagem proposta foi desenvolvida com tendo em vista ambientes dinâmicos onde a estrutura de tarefas TÆMS pode ser modificada em tempo de execução: métodos podem ser acrescentados ou removidos, o número de agentes pode mudar e os relacionamentos entre sub-tarefas podem mudar. A ferramenta atualmente utilizada com o TÆMS, chamada DTC, não é capaz de lidar com esse tipo de mudança e serve para resolução de problemas estáticos.

B.3.2 Polietismo

A divisão do trabalho é um aspecto bem estudado por ser um aspecto muito importante do comportamento dos insetos dentro da colônia (ROBISON, 1992). A especialização orientada pela idade do indivíduo é chamada, na biologia, de polietismo temporal. Em abelhas (*Apis mellifera*) esse é o aspecto principal que define a divisão do trabalho, fazendo com que abelhas jovens trabalhem dentro da colméia e as mais velhas trabalhem na parte externa atuando na defesa ou na coleta de pólen.

Theraulaz et al. (THERAULAZ; BONABEAU; DENEUBOURG, 1998) apenas sugerem que uma extensão do modelo deveria ser feita para adicionar o polietismo, em seu modelo original utilizam duas constantes para modificar o limiar do agente executar o método. Para modificar o limiar de resposta de modo a modelar o polietismo temporal,

o modelo original de (THERAULAZ; BONABEAU; DENEUBOURG, 1998) foi modificado, incluindo-se variáveis relativas a idade do agente e do método. Quando o agente i está executando o método j , o limiar é atualizado do seguinte modo:

$$\theta_{ij} = \theta_{ij} - \frac{a_i}{A_i} * \frac{\mathcal{A} - m_j}{\mathcal{A}} \quad (\text{B.3})$$

Onde:

a_i idade do agente i ;

A_i idade máxima para o agente i ;

m_j idade do método j ;

\mathcal{A} idade do método mais antigo disponível.

O limiar de resposta θ_{ij} de um indivíduo i quando *não* está executando o método é calculado utilizando-se:

$$\theta_{ij} = \theta_{ij} + \frac{a_i}{A_i} * \frac{m_j}{\mathcal{A}} \quad (\text{B.4})$$

Nesse modelo de especialização, todos os agentes iniciam como o mesmo valor de θ_{ij} . O limiar de resposta do agente é modificado quando um método é executado por ele. Para o agente se especializar em escolher um certo tipo de método, é necessário que ele o selecione previamente diversas vezes. Sendo assim, é necessário executar o modelo diversas vezes (turnos). Em cada turno de execução, a abordagem gera uma possibilidade diferente de alocação de tarefas. Dada a natureza probabilística do modelo, as alocações podem ser diferentes. Quando a estrutura para de modificar-se, considera-se que o modelo estabilizado e analisa-se a resposta.

Nas equações B.3 e B.4, considera-se a idade dos agentes (proporcional ao limite de execução da tarefa) e dos métodos. A idade do método aumenta até que ele seja completamente terminado. Um agente pode sobreviver por vários turnos ou apenas um (deixando um método incompleto). Os agentes tendem a ter limiares de resposta maiores para métodos antigos e limiares mais baixos para métodos novos no sistema. A idéia é especializar os agentes mais velhos na execução de uma gama maior de tarefas que os agentes mais jovens, de modo similar ao encontrado na natureza.

B.4 Experimentos

As simulações apresentadas nesta seção foram geradas usando uma ferramenta de simulação desenvolvida em JAVA e utilizando-se as bibliotecas da interface de programação TÆMS.

Na abordagem proposta, adota-se a Equação 3.1 para o cálculo da distribuição da tendência de um agente executar cada método dado um estímulo e um limiar de resposta. Cada método na estrutura de tarefas TÆMS tem uma probabilidade de ser executado por cada agente.

Os resultados apresentados são médias em mil repetições de cada experimento (com condições iniciais iguais de execução).

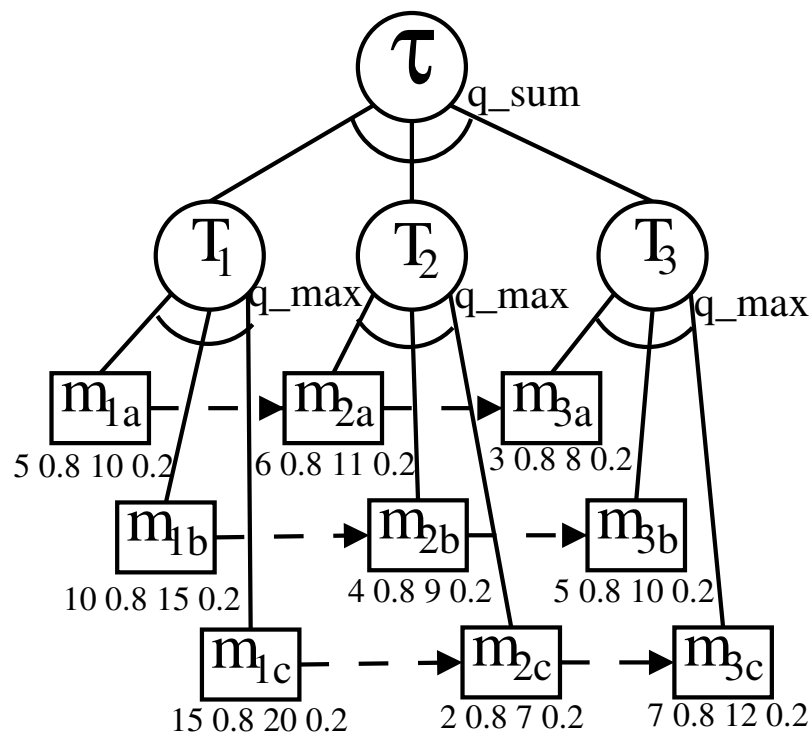


Figura B.1: Estrutura de tarefas TÆMS.

B.4.1 Cenário I

Utiliza-se a estrutura de tarefas TÆMS mostrada na Figura B.1 como base para as simulações apresentadas. Essa estrutura poderia representar, por exemplo, um problema típico de escalonamento de máquinas de propósitos múltiplos (CICIRELLO; SMITH, 2001), onde o autor associa um pequeno número de máquinas (2 a 4) com vespas. Essa estrutura também pode estar relacionada com o serviço de manutenção de aeronaves discutido em (WAGNER; GURALNIK; PHELPS, 2003).

A tarefa T_1 é o primeiro estágio da produção, tarefas do tipo a , b ou c chegam e devem ser processadas. Se a tarefa é do tipo a , algum agente deve executar a tarefa m_{1a} , a qual habilitará a tarefa m_{2a} . Note-se que neste cenário há muitos relacionamentos de habilitação de tarefas que deixa sua execução menos flexível.

A Figura B.1 mostra apenas a distribuição da duração de cada método. Todos os métodos possuem as mesmas distribuições que determinam o custo e a qualidade: a distribuição do custo é (0 0,8 1 0,2), a distribuição da qualidade é (5 0,8 4 0,2). A princípio, os tempos limites são 25 unidades de tempo.

O DTC foi executado sobre o mesmo problema, com o objetivo de comparar o escalonamento produzido pela abordagem proposta com o escalonamento produzido pelo programa padrão, o escalonamento gerado pelo DTC é visto na Tabela B.2. Essa tabela apresenta para cada método os tempos de início de de fim, qualidade e custo. Os valores não apresentados na tabela: a qualidade total é 14,35; custo total = 0,6 e a duração total é 17,00.

Apesar do DTC lidar com distribuições de probabilidade de qualidade, custo e duração vindos da estrutura TÆMS, ele não utiliza esses valores de modo probabilísticos (como por exemplo, sorteio do valor por roleta). Ao contrário, ele calcula a média para cada distribuição. No modelo proposto, utilizamos o mesmo método do DTC quando vamos

Tabela B.2: Escalonamento DTC

método	início	fim	qualidade	custo	duração
m_{1a}	0,0	6,0	4,8	0,2	6,0
m_{2a}	6,0	13,0	4,8	0,2	7,0
m_{3a}	13,0	17,0	4,76	0,2	4,0

comparar os resultados com ele, nos demais casos, utilizamos o modo probabilístico.

Utilizando a abordagem proposta com a variante não probabilística (calculando a média) é gerado o escalonamento mostrado na Tabela B.3. A abordagem produz mil resultados (como mencionado anteriormente), o resultado mostrado na tabela é o mais frequente e foi produzido 32,7% das vezes. A qualidade total é 14,4, o custo total é de 0,6 e a duração total é 17,00, indicando que o resultado mais frequente é semelhante ao resultado gerado pelo DTC.

A mesma estrutura de tarefas foi escalonada com o tratamento probabilístico das distribuições de qualidade, custo e duração. A Tabela B.4 mostra o resultado mais frequente e foi produzido 22,3% das repetições. A qualidade total é igual a 15,00, o custo total zero e a duração total 24,00. Todos três os métodos escalonados na primeira variante também são apresentados na segunda, no entanto, devido a natureza probabilística da variação da duração, mais métodos podem ser executados dentro de um mesmo tempo limite. No total, 41,5% dos resultados, escalonam pelo menos os 3 métodos que o DTC escalonou, dentro do tempo limite.

Tabela B.3: Melhor escalonamento (primeira variação).

método	início	fim	qualidade	custo	duração
m_{1a}	0,0	6,0	4,8	0,2	6,0
m_{2a}	6,0	13,0	4,8	0,2	7,0
m_{3a}	13,0	17,0	4,8	0,2	4,0

Tabela B.4: Melhor escalonamento (segunda variação).

método	início	fim	qualidade	custo	duração
m_{1a}	0,0	5,0	5,0	0,0	5,0
m_{2a}	5,0	11,0	5,0	0,0	6,0
m_{3a}	11,0	14,0	5,0	0,0	3,0
m_{1b}	14,0	24,0	5,0	0,0	10,0

A abordagem proposta não pretende ser melhor ou igual ao DTC porque destina-se a ambientes dinâmicos. Os agentes podem se adaptar às mudanças no ambiente sem a necessidade de comunicação e de atribuições externas. Os ambientes dinâmicos serão agora apresentados e discutidos.

B.4.2 Cenário II

Com o objetivo de medir o desempenho da abordagem proposta em ambientes dinâmicos, foram escalonados 4 diferentes estruturas TÆMS que apareciam aleatoriamente com a mesma probabilidade. A primeira estrutura (TS_1) é mostrada na Figura B.1. As demais são variações de TS_1 :

TS_2 não tem relações de habilitação entre as sub-tarefas;

TS_3 tem somente as sub-tarefas T_1 e T_2 ;

TS_4 o tempo limite foi alterado para 30.

Além dessas mudanças, a distribuição da qualidade em TS_2 é (15 0,8 10 0,2), em TS_3 (50 0.8 40 0.2) e em TS_4 (100 0.8 90 0.2).

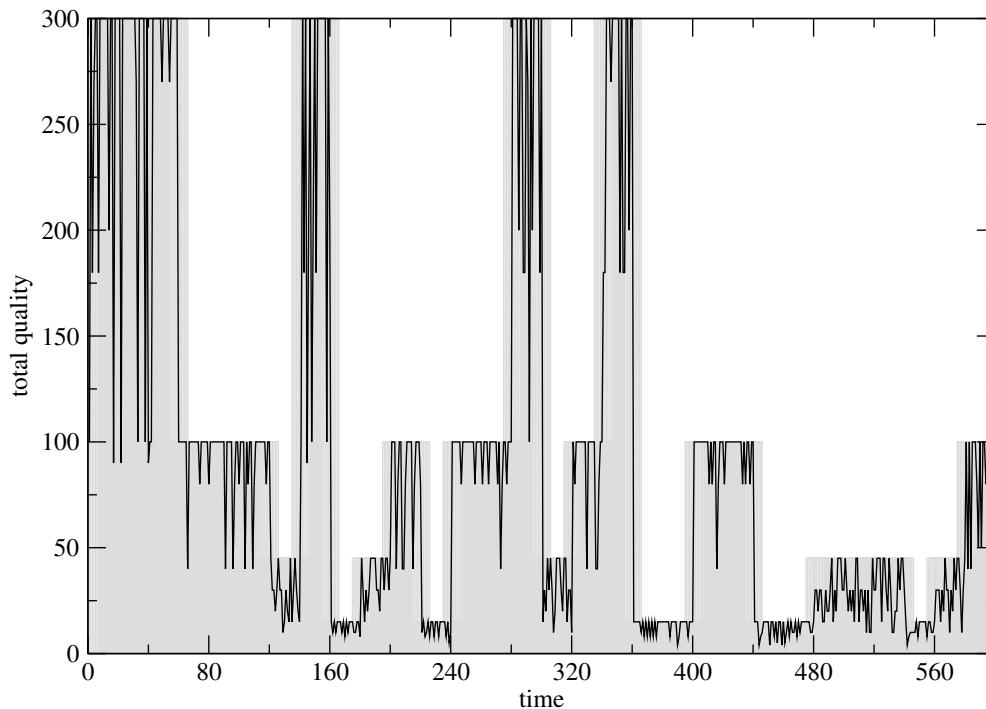


Figura B.2: Mudanças na qualidade pelo tempo (linha), as mudanças no ambiente são mostradas como barras cinza.

A Figura B.2 mostra as mudanças ocorridas no ambiente e como os agentes se adaptam a elas. No início (primeiros 50 passos) os agentes ainda estão em fase de adaptação. As barras cinza na figura indicam qual a estrutura TÆMS está sendo escalonada, quando a barra vai até 300, indica que TS_4 está sendo escalonada (isso devido às distribuições de qualidade). Quando a barra vai até a qualidade 100, 50 ou 15, a estrutura escalonada é TS_3 , TS_2 ou TS_1 respectivamente.

A Figura B.2, mostra que, quando a estrutura muda, a qualidade também muda de acordo. Esses resultados mostram que modificando a estrutura de tarefas dinamicamente em tempo de execução, há uma pequena perturbação no desempenho em relação a qualidade dos resultados, mostrando que os agentes adaptam-se bem às mudanças.

B.4.3 Cenário III

Com o mesmo objetivo da simulação do “Cenário II”, agora mudamos dinamicamente o número de agentes disponíveis para a execução das tarefas. Neste cenário, a estrutura

TÆMS básica, mostrada na Figura B.1, é utilizada como sub-tarefa de uma tarefa maior que contém três sub-tarefas idênticas e cuja QAF raiz é $sum()$, gerando uma estrutura de tarefas com 27 métodos. O número de agentes varia de 1 a 27. Para lidar com a natureza probabilística do problema, são executadas 100 repetições com as mesmas condições iniciais cada vez que mudamos o número de agentes, totalizando 2700 repetições.

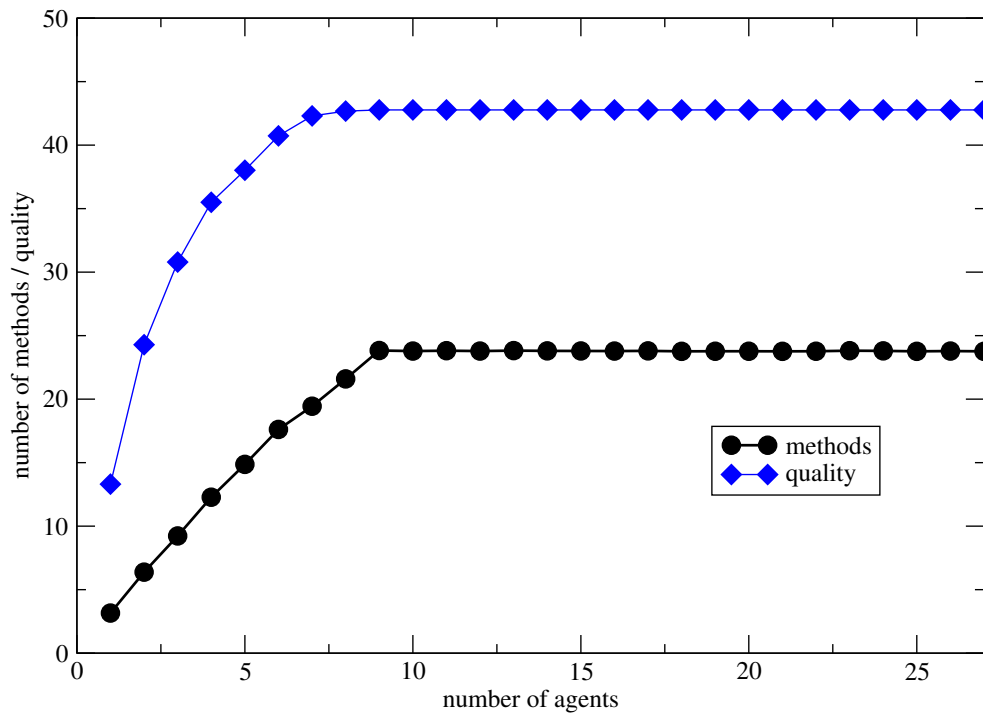


Figura B.3: Número de métodos e qualidade para número de agentes variável.

A Figura B.3 mostra a influência do número de agentes sobre o número de métodos e sobre a qualidade alcançada. Nove métodos dos 27 totais não possuem restrições para seu início. Quando aumenta-se o número de agentes, aumenta o número de métodos que são executados até chegar a 9 agentes. Adicionando mais que nove agentes não muda a qualidade total, devido as restrições da estrutura de tarefas. A maior qualidade possível (em torno de 40) é atingida quando o sistema é resolvido por 9 agentes.

B.5 Conclusões

A abordagem apresentada lida com seleção de ações e problemas de sequenciamento, buscando ser uma solução eficiente em situações onde o ambiente se modifica e demanda diferentes organizações de tarefas. Em outras abordagens, a adaptação em ambientes dinâmicos depende de um comportamento de aprendizado, normalmente com comunicação ou coordenação externa.

A solução proposta utiliza um paradigma baseado nas colônias de insetos sociais, onde há uma abundância das evidências do sucesso ecológico, apesar da falta aparente da coor-

denação explícita. Insetos sociais adaptam-se às mudanças no ambiente e às necessidades da colônia e têm como pontos chave a plasticidade na divisão do trabalho.

O objetivo é mostrar que tal abordagem pode ser utilizada para alocar tarefas a agentes em SMA, quando as organizações mudam dinamicamente. Como mostrado em (BONABEAU et al., 1997), não há nenhuma maneira padrão de avaliar o desempenho dos algoritmos que tratam dos ambientes dinâmicos, porque os problemas padrão (por exemplo “Caixeiro Viajante”) são problemas de natureza estática. Foram utilizados o GPGP e DTC como comparativo, embora este comparativo seja limitado aos casos estáticos. Quanto às situações dinâmicas, podemos apenas discutir a vantagem qualitativa da abordagem apresentada. Nos cenários apresentados, apesar dos agentes levarem algum tempo para atingir a estabilidade, a adaptação é alcançada antes do tempo limite imposto pela tarefa. Como apresentado em (ABDALLAH; DARWISH; HEGAZY, 2002), os mecanismos GPGP não lidam com sincronismo nem com redundância de tarefas (dois agentes executando a mesma tarefa). PO exemplo, no Cenário III, mais de um agente podem estar executando a mesma tarefa, mas somente um agente pode estar executando um mesmo método.

Em suma, há um meio-termo entre a coordenação explícita que conduz à resultados precisos e a coordenação implícita pela adaptação, que gera resultados menos precisos ou de menor qualidade. A abordagem proposta não é a melhor abordagem para ambientes estáticos, mas mostra-se eficiente em ambientes dinâmicos. A eficiência do método está relacionada ao cenário que ele destina-se a resolver.