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

TAYLOR DE OLIVEIRA ANTES

# Organização Hierárquica com Agregação de Estados em Aprendizado Multiagente: uma Aplicação em Controle Semafórico

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

Orientador: Profª. Drª Ana L. C. Bazzan
Co-orientador: Prof. Dr. Anderson R. Tavares

Porto Alegre
2021

**CIP — CATALOGAÇÃO NA PUBLICAÇÃO**

De Oliveira Antes, Taylor

Organização Hierárquica com Agregação de Estados em Aprendizado Multiagente: uma Aplicação em Controle Semafórico / Taylor De Oliveira Antes. – Porto Alegre: PPGC da UFRGS, 2021.

87 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2021. Orientador: Ana L. C. Bazzan; Co-orientador: Anderson R. Tavares.

I. Bazzan, Ana L. C.. II. Tavares, Anderson R.. III. Título.

## AGRADECIMENTOS

Agradeço primeiramente a vida, essa dádiva que nos foi dada com suas infinitas possibilidades e a qual deveríamos aproveitar da melhor maneira possível.

À minha mãe e ao meu pai pelo amor, paciência e apoio. Meus heróis e exemplos de força e de determinação.

Obrigado a UFRGS pelas oportunidades de graduação e pós-graduação. Obrigado a todos os envolvidos com o Instituto de Informática e, principalmente, aos seus grandes professores, verdadeiras inspirações para todo e qualquer aluno que deseja se tornar um profissional da área.

Obrigado à minha orientadora Ana Bazzan e ao meu co-orientador Anderson Tavares pela orientação, pela paciência e pelas ideias debatidas que tornaram esse trabalho possível.

Aos companheiros de MASLab, pelas experiências e convivência, alegrias e risadas compartilhadas. Em especial a Lucas Alegre, pelo trabalho o qual serviu de inspiração para os modelos locais deste trabalho.

Ao CNPq pelo auxílio financeiro fundamental para a realização desse trabalho. Às instalações do MASLab e da UFRGS e ao RU que forneceu e fornece combustível para diversas mentes brilhantes do meio acadêmico.

E por fim, agradeço a Ela, inteligente, bela e engraçada, que me apoiou durante todo esse percurso, tornando cada dia especial. Te amo.

**SUMÁRIO**

## LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| CS | Controle Semafórico |
| TSC | *Traffic Signal Control* |
| ITS | Sistemas Inteligentes de Transporte, em inglês, Intelligent Transportation Systems |
| ATMS | Sistemas Avançados de Gerenciamento de Tráfego, em inglês, Advanced Traffic Management Systems |
| RL | Aprendizado por reforço, em inglês, *Reinforcement Learning* |
| ML | Aprendizado de máquina, em inglês, *Machine Learning* |
| MARL | Aprendizado por reforço multiagente, em inglês, *Multiagent Reinforcement Learning* |
| IL | Aprendizado independente, em inglês, *Independent learning* |
| MDP | Processo de decisão de Markov, em inglês, *Markov decision process* |
| MMDP | Processo de decisão de Markov multiagente, em inglês, *multiagent Markov decision Process* |
| TraCI | *Traffic control interface* |
| OD | Origem-destino |
| A-CATs | *Actor-critic adaptive traffic signal* |
| ATSC | *Adaptative traffic signal control* |
| GCNN | *Graph Convolutional Neural Networks* |
| LSTM | Memória longa de curto prazo, em inglês, *Long short-term memory* |
| VHO | Organização hierárquica baseada em vetores, em inglês, *Vector based hierarchical organization* |

## LISTA DE SÍMBOLOS

| | |
|---|---|
| $L_i$ | conjunto de faixas de uma interseção $i$ |
| $\vec{e}_{I,i}$ | vetor resultante de entrada de uma interseção $i$ |
| $\vec{s}_{I,i}$ | vetor resultante de saída de uma interseção $i$ |
| $\vec{e}_l$ | vetor de faixa $l$ que entra em uma interseção |
| $\vec{s}_l$ | vetor de faixa $l$ que sai de uma interseção |
| $R$ | conjunto dos agentes região |
| $R_a^h$ | agente região $a$ de nível hierárquico $h$ |
| $R^{-a}$ | conjunto dos agentes região subordinados a um agente $a$ |
| $\vec{e}_{R,a}$ | vetor resultante de entrada de uma região $a$ |
| $\vec{s}_{R,a}$ | vetor resultante de saída de uma região $a$ |
| $I$ | conjunto dos agentes interseção |
| $I^{-a}$ | conjunto dos agentes interseção subordinados a um agente $a$ |
| $S^{-a}$ | conjunto dos agentes subordinados a um agente $a$, um $R^{-a}$ ou um $I^{-a}$ |
| $p$ | informação processada de um agente subordinado |
| $\mathcal{A}^{+a}$ | recomendação do agente supervisor de um agente $a$ |
| $o_l$ | ocupação de uma faixa $l$ |
| $f_l$ | fila de uma faixa $l$ |
| $\alpha_I$ | taxa de aprendizagem de um agente interseção |
| $\gamma_I$ | fator de desconto de um agente interseção |
| $\varepsilon_I$ | taxa de exploração de um agente interseção |
| $\alpha_R$ | taxa de aprendizagem de um agente região |
| $\gamma_R$ | fator de desconto de um agente região |
| $\varepsilon_R$ | taxa de exploração de um agente região |

## LISTA DE FIGURAS

## LISTA DE TABELAS

## LISTA DE ALGORITMOS

## RESUMO

Controle semafórico é uma possível solução para o sério problema de aumento de congestionamento nas áreas urbanas. Técnicas de aprendizado por reforço multiagente (MARL) têm mostrado resultados significativos na otimização de controladores semafóricos, visto que distribuem o controle global do tráfego entre agentes locais responsáveis pelos controladores. Assim, cada agente local tem uma visão parcial do ambiente e otimiza sua política baseado em suas observações. Contudo, o tráfego que passa por uma interseção não depende apenas de influências locais, mas de informações da rede de transporte como um todo. Do ponto de vista computacional, realizar o controle semafórico de uma rede de transporte de forma centralizada é uma tarefa de difícil execução devido à grande quantidade de variáveis envolvidas; enquanto de forma descentralizada, é possível não atingir o melhor desempenho do sistema, visto que os agentes buscam melhorar individualmente. O método proposto nesta dissertação baseia-se em utilizar uma organização hierárquica para aumentar a visão dos agentes locais e coordená-los com o objetivo de melhorar o desempenho do sistema. O método é inspirado em algumas técnicas de aprendizado por reforço que utilizam uma organização hierárquica. Contudo, diferencia-se dessas técnicas por apresentar uma metodologia hierárquica mais flexível em relação às interações entre os agentes de diferentes níveis. Na metodologia proposta, uma organização hierárquica com um número arbitrário de níveis é apresentada. Agentes supervisores, de nível $l$, são responsáveis por um conjunto de agentes subordinados, de nível $l-1$. Os subordinados transmitem uma abstração de suas observações do ambiente para seus supervisores. Os supervisores utilizam essas abstrações para aprender uma recomendação de alto nível a qual guiará o aprendizado dos seus subordinados para um melhor desempenho coletivo. Na aplicação para controle semafórico, a rede de transporte é dividida em regiões de diferentes níveis hierárquicos, cada região sendo controlada por um agente. Logo, quanto mais alto o nível hierárquico do agente região, mais ampla é sua visão do tráfego na rede de transporte. Na base da hierarquia se encontram os agentes dos controladores semafóricos, localizados em cada interseção. Os resultados dos experimentos, realizados em uma rede sintética em *grid*, mostram que a metodologia proposta de aprendizado por reforço com organização hierárquica tem melhor desempenho quando comparada a um método de tempo fixo e a um método com aprendizado por reforço sem organização hierárquica.



**Palavras-chave:** Aprendizado por reforço. Organização hierárquica. Controle semafórico.

**Hierarchical Organization with State Aggregation in Multiagent Learning: an application in Traffic Signal Control**

## ABSTRACT

Traffic signal control is a possible solution to the serious problem of congestion increase in urban areas. Multi-agent reinforcement learning (MARL) techniques have shown significant results in the traffic signal controllers' optimization, since they distribute the global traffic control among local agents responsible for the controllers. Thus, each local agent has a partial view of the environment and optimizes its policy based on its observations. However, traffic passing through an intersection does not depend only on local influences, but on information from the transport network as a whole. From a computational point of view, carrying out the traffic control of a transport network in a centralized way is difficult task due to the large number of variables involved; while in a decentralized way, it is possible not to achieve the best performance of the system, since the agents seek to improve individually. The method proposed in this dissertation uses a hierarchical organization to increase the local agents' vision and coordinate them in order to improve the performance of the system. The method is inspired by reinforcement learning techniques that use a hierarchical organization. However, it differs from these techniques in that it presents a more flexible hierarchical methodology in relation to the interactions between agents at different levels. In the proposed methodology, a hierarchical organization with an arbitrary number of levels is presented. Supervisor agents, of level $l$, are responsible for a set of subordinate agents, of level $l-1$. The subordinates transmit an abstraction of their environment' observations to their supervisors. The supervisors use these abstractions to learn a high-level recommendation that will guide their subordinates' learning to a better collective performance. In the traffic control application, the transportation network is divided into regions of different hierarchical levels, each region being controlled by an agent. Therefore, the higher the hierarchical level of the region agent, the broader his view of traffic on the transport network. At the bottom of the hierarchy are the traffic signal controller agents, located at each intersection. The results of the experiments, carried out in a synthetic grid network, show that the proposed reinforcement learning approach with hierarchical organization outperforms a fixed time method and a reinforcement learning method without hierarchical organization.

**Keywords:** Reinforcement Learning. Hierarchical Organization. Traffic Signal Control.

# 1 INTRODUÇÃO

Este capítulo tem como objetivo apresentar o contexto no qual a dissertação se insere, discutindo a motivação (Seção 1.1), a proposta realizada no trabalho (Seção 1.2) e a organização dos capítulos (Seção 1.3).

## 1.1 Motivação

O congestionamento de veículos se tornou um obstáculo nas cidades ao redor do mundo devido à urbanização e ao crescimento populacional, causando graves problemas como o aumento de acidentes no tráfego e o aumento da poluição pelo alto consumo de combustível. Conforme as áreas urbanas se expandiram, a demanda por mobilidade aumentou e o problema de gerenciar essa grande quantidade de veículos se tornou uma importante área de pesquisa, uma vez que a mobilidade está diretamente relacionada à economia e à qualidade de vida dos cidadãos. Como, na maioria dos casos, a expansão da infraestrutura de transporte é de difícil execução, o uso mais eficiente da infraestrutura existente se mostra uma solução mais atraente nesse contexto.

A área de sistemas inteligentes de transporte (ITS - *intelligent transportation systems*) envolve a aplicação das tecnologias de informação, comunicação, controle e eletrônica na área de sistemas de transporte. Dentro de ITS, a área de sistemas avançados de gerenciamento de tráfego (ATMS - *advanced traffic management systems*) foca nas tecnologias relacionadas a dispositivos de controle de tráfego, gerenciamento de situações de emergência, monitoramento e comunicação entre as diferentes partes do sistema relacionados à segurança. Controle semafórico (TSC - *traffic signal control*) é a área de ATMS focada no gerenciamento e controle de semáforos, sendo uma das soluções mais tradicionais para otimizar o tráfego de veículos, aproveitando melhor a infraestrutura existente.

Controle semafórico é uma solução prática para o problema de tráfego, visto que ele atenua o congestionamento e melhora o fluxo de veículos. Tradicionalmente, controladores semafóricos usam tempos fixos (WEBSTER, 1958) ou otimizam a alocação de tempos semafóricos previamente calculados (LOWRIE, 1982; HUNT et al., 1981) para gerenciar o fluxo de tráfego. Com o avanço tecnológico, técnicas de aprendizado de máquina (ML - *machine learning*) vêm apresentando diversas aplicações na área de ITS. Técnicas como a de aprendizado por reforço (RL - *reinforcement learning*) têm mostrado resultados promissores na área de controle semafórico (ASLANI; MESGARI; WIERING,

2017; CHU et al., 2019; NISHI et al., 2018). Em aprendizado por reforço para controle semafórico, os agentes (controladores) aprendem como agir a partir de interações com o sistema de tráfego buscando responder de melhor maneira às variações na demanda de veículos em tempo real.

As técnicas de aprendizado por reforço para controle semafórico podem ser divididas em duas metodologias: aprendizado por reforço centralizado e aprendizado por reforço multiagente (MARL - *multiagent reinforcement learning*). O aprendizado por reforço centralizado para controle semafórico consiste em coletar todas informações de tráfego de uma rede de transporte para formar um estado global e aprender uma maneira de agir, chamada de política, a partir dos diferentes estados globais. Normalmente, esse método não é realizado em grandes redes de transporte uma vez que o espaço de estados e ações para o aprendizado cresce exponencialmente com o número de controladores na rede, problema conhecido como maldição da dimensionalidade (*curse of dimensionality*).

As técnicas de MARL normalmente se encaixam em aprendizado independente (IL - *independent learning*) ou em uma otimização centralizada de agentes coordenados. A otimização centralizada tem o mesmo problema de dimensionalidade que o método centralizado. No aprendizado independente, cada interseção da rede é controlada por um agente que possui suas informações locais e aprende sua política de forma independente, interpretando o aprendizado (que leva à mudança de comportamento) dos outros agentes como mudanças nas dinâmicas do ambiente. O aprendizado independente pode ser implementado em grandes redes de transporte, porém, como o agente otimiza a sua política baseado apenas nas suas observações locais, ele busca o melhor desempenho de maneira individual. Contudo, no controle semafórico, o tráfego não depende apenas das informações locais de uma interseção individual, mas sim da combinação das informações das diferentes interseções da rede de transporte. Assim, buscando uma solução escalável com um melhor desempenho na otimização do fluxo de tráfego na rede de transporte como um todo, utilizamos uma organização hierárquica para aumentar a visão dos agentes que controlam as interseções, dividindo a rede de transporte em regiões, de diferentes níveis hierárquicos, que influenciarão no aprendizado dos agentes que pertencem a essas regiões.

Do ponto de vista computacional, realizar o controle semafórico de uma rede de transporte de forma centralizada é uma tarefa de difícil execução devido à grande quantidade de variáveis envolvidas; enquanto, de forma descentralizada, é possível não atingir o melhor desempenho do sistema, visto que os agentes buscam melhorar individualmente. Assim, uma organização hierárquica é utilizada como uma possível solução para coorde-

nar os diferentes agentes com o objetivo de melhorar o desempenho do sistema como um todo.

A utilização de uma organização hierárquica é pouco explorada em aprendizado por reforço para controle semafórico devido ao desafio de se definir os modelos (estados, ações e recompensas) e as informações relevantes dos agentes em diferentes níveis hierárquicos. Este trabalho apresenta uma organização hierárquica mais flexível que trabalhos anteriores, em relação às interações entre os agentes de diferentes níveis.

## 1.2 Proposta

Esta dissertação apresenta uma estrutura genérica que organiza o aprendizado por reforço dos agentes de forma hierárquica e uma aplicação desse método para controle semafórico usando cálculos com vetores. A organização hierárquica é inspirada nas ideias do aprendizado hierárquico (DIETTERICH, 1999), aprendizado feudal (DAYAN; HINTON, 1993) e aprendizado holônico (GERBER; SIEKMANN; VIERKE, 1999). Os agentes são organizados em um número arbitrário de níveis hierárquicos onde cada agente supervisor, de nível $l$, possui agentes subordinados de nível $l-1$. Os agentes subordinados transmitem abstrações de suas informações para seus supervisores, logo, quanto maior o nível hierárquico do supervisor, mais abrangente será sua visão. Os agentes supervisores usam as informações de seus subordinados para criarem seus estados e aprenderem uma recomendação que influenciará o aprendizado dos seus subordinados, aumentando a visão deles sobre o problema em questão. Assim, ao usar essa organização hierárquica, os agentes supervisores podem guiar seus subordinados a um melhor desempenho não apenas individual, mas coletivo.

Na aplicação para controle semafórico, a rede de transporte é dividida em regiões de diferentes níveis hierárquicos, cada região sendo controlada por um agente. Na base da hierarquia se encontram os agentes interseção, que controlam os semáforos. Nos diferentes níveis hierárquicos, agentes região, de nível $l$, podem supervisionar um conjunto de agentes interseção ou um conjunto de agentes região, de nível $l-1$. Os agentes subordinados transmitem informações sobre suas observações locais aos agentes supervisores. Os supervisores usam as informações de seus subordinados para aprender uma recomendação: uma indicação para favorecer o tráfego em uma direção sobre a região que controlam. Essa recomendação é passada aos subordinados, que a consideram ao aprenderem suas políticas. A organização hierárquica ajuda a guiar o aprendizado dos agentes, usando

indicações de diferentes níveis regionais, para um melhor desempenho coletivo enquanto mantém escalabilidade. Na aplicação, as informações e indicações passadas são calculadas usando vetores para modelar os fluxos de tráfego. Os vetores representam em seus ângulos os sentidos dos fluxos e em suas magnitudes as quantidades de veículos em cada um. Assim, os agentes região podem, de uma maneira simples, considerar as informações de todos seus subordinados em primeiro nível e não apenas tratar sua região como uma caixa preta, i.e, considerar apenas as informações dos limites da região controlada.

Os experimentos realizados mostram que a utilização da organização hierárquica proposta apresenta um melhor desempenho do que outros métodos de controle semafórico: um método de tempo fixo e um método de aprendizado por reforço sem organização hierárquica.

Assim, as principais contribuições do trabalho realizado nesta dissertação são:

- Uma estrutura genérica escalável para organizar o aprendizado por reforço de maneira hierárquica, onde os agentes em diferentes níveis são guiados pelos seus supervisores a um melhor desempenho coletivo;
- Um método que utiliza operações com vetores para aprendizado por reforço em controle semafórico baseado na estrutura apresentada.

## 1.3 Organização dos Capítulos

Os próximos capítulos dessa dissertação estão organizados da seguinte forma:

- Capítulo 2: apresenta a fundamentação teórica, terminologia e conceitos de base para o entendimento desta dissertação;
- Capítulo 3: apresenta uma discussão sobre trabalhos relacionados à dissertação, discutindo trabalhos realizados em controle semafórico no contexto geral, trabalhos que utilizam aprendizado por reforço em controle semafórico e trabalhos que utilizam aprendizado com uma organização hierárquica em controle semafórico;
- Capítulo 4: discute a abordagem de aprendizado por reforço com organização hierárquica utilizada no presente trabalho. É apresentada a organização hierárquica de forma genérica e uma aplicação para controle semafórico usando vetores;
- Capítulo 5: apresenta o cenário estudado, os modelos utilizados e os resultados dos experimentos realizados para avaliar o desempenho da abordagem proposta;

- Capítulo 6: apresenta uma revisão geral da dissertação, apontando conclusões, contribuições e perspectivas de continuidade deste trabalho.

# 2 FUNDAMENTAÇÃO TEÓRICA

Esta dissertação apresenta uma abordagem de aprendizado por reforço com organização hierárquica dos agentes para controle semafórico. Este capítulo apresenta a terminologia e os conceitos básicos dos diferentes tópicos abordados ao decorrer da dissertação, com o intuito de embasar o leitor menos familiarizado com alguns desses temas. O capítulo começa com uma breve revisão sobre agentes autônomos e sistemas multiagente (Seção 2.1), seguido por discussões sobre aprendizado por reforço e aproximação de função (Seções 2.2 e 2.3), aprendizado por reforço multiagente e aprendizado com organização hierárquica (Seções 2.4 e 2.5), controle semafórico e simulação de tráfego (Seções 2.6 e 2.7).

## 2.1 Agentes autônomos e sistemas multiagente

Não existe uma definição de agente universalmente aceita, porém há uma convergência das diferentes definições para um conceito de autonomia. Um agente, segundo Franklin e Graesser (1997), é uma entidade capaz de perceber seu ambiente, por meio de sensores, e de agir sobre esse ambiente em prol de um objetivo próprio. Segundo Wooldridge (2009), um agente é um sistema computacional, situado em um ambiente, capaz de realizar ações autônomas sobre o ambiente para atingir seus objetivos, definidos pelo seu projetista.

Um sistema multiagente é composto por múltiplos agentes que interagem entre si (WOOLDRIDGE, 2009). Cada agente atua sobre o ambiente e possui, portanto, uma "esfera" de influência sobre o ambiente em que está situado. As esferas de influência de diferentes agentes podem se sobrepor, podendo gerar relações entre eles. Essas relações podem ser de caráter colaborativo, quando os objetivos de diferentes agentes estão alinhados, ou de caráter competitivo, quando os objetivos de diferentes agentes se opõem.

Para uma leitura mais detalhada sobre agentes autônomos e sistemas multiagentes, o leitor pode consultar (WOOLDRIDGE, 2009).

## 2.2 Aprendizado por Reforço

A ideia de aprender interagindo com o ambiente é provavelmente a primeira a ocorrer quando pensamos na natureza do aprender. Em muitas ocasiões, não temos um professor explícito, mas uma conexão sensorial com o ambiente. Aprendemos, por essa interação, a relacionar causa e efeito das nossas ações e, então, o que fazer para atingir um objetivo; por exemplo, quando pequenos, choramos para sermos amamentados. Na versão computacional de aprendizado por reforço, acontece exatamente esse fato: aprendemos interagindo com o ambiente. O agente aprende a interagir ao tomar ações baseadas no seu estado, identificado por seus sensores, e ao receber uma recompensa - um sinal numérico - pela ação executada. O mapeamento de estado para ação é chamado de *política*. Para o agente, não é dito explicitamente quais ações realizar. Ao invés disso, ele deve descobrir quais ações retornam a maior recompensa, experimentando-as. A ideia é que o agente interaja com o ambiente e aprenda a política que maximize suas recompensas. Além de lidar com a recompensa imediata, o agente deve lidar com o problema de decisão sequencial, uma vez que suas ações podem afetar seu estado futuro e recompensas subsequentes. Assim, os pontos principais do aprendizado por reforço são as suas características de experimentação, tentativa-e-erro, das ações e a maximização das recompensas tanto imediatas quanto subsequentes.

Um problema de aprendizado por reforço geralmente pode ser modelado como um processo de decisão de Markov (MDP - *Markov decision process*). Um MDP é composto por uma tupla $(S,\ A,\ T,\ R)$, onde $S$ é o conjunto de estados do ambiente; $A$ é o conjunto de ações; $T$ é a função de transição de estados: $S \times A \rightarrow \Psi(S)$, onde $\Psi(S)$ é uma distribuição de probabilidades sobre $S$; e $R$ é a função de recompensa esperada: $S \times A \rightarrow \mathbb{R}$. Nesse contexto, o agente interage com o ambiente seguindo uma política $\pi$ e tenta aprender a política ótima $\pi^*$, que mapeia cada estado $s \in S$ para uma ação $a \in A$ de forma que a utilidade futura seja maximizada. A utilidade de cada par estado-ação é baseada nas recompensas imediatas e subsequentes que o agente recebe ao interagir com o ambiente. Seja $r_{t+1}, r_{t+2}, r_{t+3}, \cdots$, a sequência de recompensas recebidas após o instante $t$ e $\gamma \in [0, 1]$ o fator de desconto, a utilidade $G_t$, chamada de *retorno*, é definida pela Eq. 2.1:

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \tag{2.1}$$

O fator de desconto $\gamma$ determina a importância das recompensas futuras. Um fator de desconto com valor igual a 0 fará com que o agente considere apenas a recompensa imediata, $r_{t+1}$. E, conforme seu valor aumenta, mais o agente considera as recompensas distantes no futuro.

Dentre os diferentes algoritmos de aprendizado por reforço, o *Sarsa* (RUMMERY; NIRANJAN, 1994) é amplamente utilizado. No Sarsa, um agente aprende a utilidade de cada par estado-ação, chamada *valor-Q*. O *valor-Q* é uma função de valor representada por $Q(s, a)$ e reflete a utilidade esperada do agente ao realizar uma dada ação $a$ em um dado estado $s$ e seguir sua política $\pi$ a partir disso. Sendo $\mathbb{E}_\pi$ o valor esperado ao seguir uma política $\pi$; $G_t$ o retorno; $S_t$ o estado atual do agente no instante $t$ e $A_t$ a ação tomada pelo agente no instante $t$; o valor $Q(s, a)$ é definido pela Eq. 2.2:

$$Q(s,a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| S_t = s, A_t = a\right] \tag{2.2}$$

O *valor-Q* pode ser aprendido diretamente a partir de uma tupla de experiência do agente $(s, a, r, s', a')$, que significa que, em um estado $s$, o agente realizou a ação $a$, recebendo a recompensa $r$, entrando em um próximo estado $s'$ e escolhendo a próxima ação $a'$. Assim, sendo $\alpha \in [0, 1]$ a taxa de aprendizagem e $\gamma \in [0, 1]$, o fator de desconto, sua atualização é feita através da Eq. 2.3:

$$Q(s,a) \leftarrow (1-\alpha)\, Q(s,a) + \alpha\,(\, r + \gamma\, Q(s',a')\,) \tag{2.3}$$

A taxa de aprendizagem determina o quanto a nova interação adquirida ajusta o *valor-Q* previamente aprendido. Uma taxa de aprendizagem de 0 fará com que o agente não aprenda com as novas interações e considere apenas o valor $Q(s, a)$ inicialmente definido, enquanto um taxa de aprendizagem de 1 fará o agente considerar apenas a interação mais recente.

Os diferentes *valores-Q* são armazenados em uma tabela chamada de *tabela-Q* e, com os *valores-Q* calculados, um agente, em um estado $s$, precisa selecionar qual ação executar dentre as ações possíveis desse estado. Para isso, geralmente, é utilizada uma estratégia que equilibre a exploração (ganho de conhecimento) e o aproveitamento (uso de conhecimento). Um estratégia popular é a $\varepsilon$-*greedy*, que consiste em escolher uma ação aleatória (exploração) com a probabilidade $\varepsilon$ ou escolher a melhor ação conhecida, i.e., com maior *valor-Q* (aproveitamento), com probabilidade $1 - \varepsilon$.

Para uma discussão mais detalhada sobre aprendizado por reforço recomenda-se a leitura de (SUTTON; BARTO, 2018).

## 2.3 Aproximação de Função

Os métodos de aproximação de função podem ser usados para lidar com espaços de estado arbitrariamente grandes. Em muitas tarefas que gostaríamos de aplicar aprendizado por reforço, o conjunto de estados é enorme. O problema com conjuntos de estados grandes não é apenas a memória necessária para grandes tabelas de valores Q, mas o tempo e os dados necessários para preenchê-las com precisão. Nesses casos, quase todo estado encontrado estará sendo visto pela primeira vez e, para realizar decisões, precisamos generalizar informações de estados encontrados anteriormente, que sejam similares ao atual. Assim, nosso objetivo é encontrar uma boa solução para aproximar a variedade de estados.

O tipo de generalização normalmente utilizada com aprendizado por reforço é chamada de *aproximação de função*. O método recebe esse nome pois generaliza uma aproximação de uma dada função, por exemplo, a função de *valor-Q*, a partir de um conjunto de exemplos reduzido. A aproximação de função que utilizaremos nos agentes desse trabalho é chamada de *aproximação linear de funções*.

Na aproximação linear de funções, a aproximação é feita a partir de uma função linear sobre vetores de pesos. Para cada estado $s$ existe um vetor $x(s) = (x_1(s), x_2(s), ..., x_f(s))$ chamada de vetor de *features* do estado $s$. Cada componente do vetor $x_i(s) \in x(s)$ é uma função $x_i : S \rightarrow \mathbb{R}$ que indica uma característica do estado. Para cada ação $a$ existe um vetor de pesos $w_a$ com o mesmo número de componentes que o vetor de $features$. Assim, o valor de Q(s,a) é aproximado pelo produto interno definido pela Eq. 2.4:

$$Q(s, a) = w_a \, x(s) = \sum_{i=1}^{n} w_{a,i} \, x_i(s) \tag{2.4}$$

Nessa perspectiva, ao invés de gerenciarmos uma tabela de valores Q(s,a) para cada par estado-ação, gerenciamos apenas os vetores de pesos de cada ação. Assim, a cada iteração, atualizamos o vetor de pesos da ação realizada. Sendo $\delta$ o erro de diferença temporal, definida pela Eq. 2.5:

$$\delta = r + \gamma \, Q(s', a') - Q(s, a) \tag{2.5}$$

Cada peso $w_{a,i}$ do vetor de pesos da ação realizada, $w_a$, é atualizado segundo a Eq. 2.6:

$$w_{a,i} \leftarrow w_{a,i} + \alpha\delta x_i(s) \tag{2.6}$$

Para uma visão mais detalhada sobre aproximação de função o leitor pode consultar os Capítulos 9 a 11 de (SUTTON; BARTO, 2018).

## 2.4 Aprendizado por Reforço Multiagente

O aprendizado por reforço em sistemas multiagente ou aprendizado por reforço multiagente (MARL - *multiagent reinforcement learning*) é mais complexo, visto que agora vários agentes aprendem no mesmo ambiente. Cada agente deve adaptar-se não somente ao ambiente, mas também ao comportamento dos outros agentes presentes. Nesse contexto, a adaptação de um agente em específico faz com que os outros tenham que mudar seus comportamentos, gerando, assim, uma nova adaptação do agente anterior. Logo, o aprendizado é muito mais dinâmico e complexo para ser resolvido usando apenas técnicas de aprendizado por reforço monoagente, como, por exemplo, o previamente citado Sarsa (Seção 2.2).

Um problema de aprendizado por reforço multiagente pode ser modelado como um processo de decisão de Markov multiagente (MMDP - *multiagent Markov decision process*), que é uma generalização do MDP. Um MMDP, também chamado de jogo estocástico (SG - *stochastic game*), é constituído de: um conjunto de agentes $N = \{1, \cdots, N\}$; um conjunto de estados do ambiente $S$; uma coleção de conjuntos de ações possíveis $A = \{A_1, \cdots, A_n\}$ dos diferentes agentes que pertencem a $N$; uma função de transição $T : S \times A_1 \times \cdots \times A_n \rightarrow \Psi(S)$ e uma função de recompensa por agente $i \in N$, $R_i : S \times A_1 \times \cdots \times A_n \rightarrow \mathbb{R}$. A função de transição retorna o próximo estado baseado nas ações combinadas que cada agente realizou no estado anterior. A função de recompensa mostra que a recompensa de cada agente depende não somente de sua ação, mas também das ações dos outros agentes.

Um grande desafio dos problemas de aprendizado por reforço multiagente modelados como MMDP é a escalabilidade. Como o espaço de estados-ações cresce exponencialmente de acordo com o número de agentes e as ações disponíveis para eles (problema conhecido como a maldição da dimensionalidade), é impraticável resolver MMDPs

em larga escala. Para resolver essa questão, alguns problemas de MARL são tratados modelando-se o processo de aprendizado de cada agente como um MDP. Assim, cada agente aprende de maneira descentralizada sem considerar a adaptação dos outros agentes; a mudança de política dos outros agentes é apenas considerada como uma mudança na dinâmica do ambiente. Nessa abordagem, os agentes são chamados de aprendizes independentes (CLAUS; BOUTILIER, 1998). A desvantagem dessa abordagem é que os agentes, sem considerar a adaptação dos outros, podem convergir para políticas ótimas individuais, podendo não ser a melhor solução para o problema como um coletivo de agentes.

No presente trabalho, nos focaremos em aprendizado com organização hierárquica, mas o leitor interessado pode encontrar uma discussão mais detalhada sobre o assunto em (BUŞONIU; BABUSKA; SCHUTTER, 2008), no qual são apresentadas as principais técnicas de MARL e os principais desafios da área.

## 2.5 Aprendizado com Organização Hierárquica

Como visto na seção anterior, um dos grandes desafios do aprendizado por reforço multiagente é a escalabilidade. A fim de atenuar esse problema, diferentes técnicas podem ser usadas, entre elas, a utilização de uma organização hierárquica. Normalmente, na literatura, quando procuramos pela utilização de uma hierarquia em aprendizado por reforço, encontramos três diferentes abordagens: o aprendizado por reforço hierárquico (*hierarchical reinforcement learning*), o aprendizado por reforço feudal (*feudal reinforcement learning*) e o aprendizado por reforço para sistemas multiagente holônicos (*holonic multi-agent systems*).

No aprendizado por reforço hierárquico, um problema alvo modelado como um MDP é decomposto em uma hierarquia de MDPs menores, i.e, uma tarefa é dividida em outras sub-tarefas menores. Assim, tendo essas sub-tarefas e seus respectivos objetivos definidos, as sub-tarefas são alocadas a diferentes agentes. Os agentes, no processo de aprendizado das sub-tarefas, não precisam se preocupar com o espaço estado-ação completo do problema original. Para uma visão mais detalhada do aprendizado por reforço hierárquico, seus conceitos, vantagens e desvantagens, é recomendada a leitura de (DIETTERICH, 1999), trabalho monoagente que é utilizado como base para os avanços no assunto até hoje.

No aprendizado por reforço feudal (DAYAN; HINTON, 1993), os agentes são

separados em uma hierarquia de gerentes e subgerentes, onde os gerentes aprendem a atribuir tarefas aos subgerentes e os subgerentes aprendem a satisfazer essas tarefas. Os gerentes possuem controle total sobre os subgerentes, podendo atribuir tarefas a eles, recompensá-los ou puni-los. Nessa hierarquia, cada gerente precisa saber o estado do sistema apenas segundo seu nível de granularidade, i.e., os agentes consideram as informações do sistema pertinentes ao seu nível na hierarquia, não considerando informações dos níveis abaixo ou acima do seu. Outro ponto importante, é que os subgerentes são recompensados quando satisfazem os sub-objetivos atribuídos a eles, mesmo que esses sub-objetivos não satisfaçam o objetivo do gerente. Por outro lado, quando os subgerentes não satisfazem seus sub-objetivos, não recebem recompensa, mesmo que as ações realizadas satisfaçam o objetivo do gerente. Dessa maneira, subgerentes aprendem a satisfazer seus sub-objetivos, mesmo quando o seu gerente erra nessas atribuições, e cabe ao gerente atribuir sub-objetivos corretos aos seus subgerentes.

O paradigma holônico é uma abordagem utilizada para reduzir a complexidade de sistemas ao definir diferentes níveis de abstração. O conceito holônico foi desenvolvido pelo filósofo Koestler (1979) para explicar a evolução de sistemas sociais e biológicos. Um *holon*, de acordo com Koestler, é definido simultaneamente como um todo e uma parte de um todo, logo pode ser constituído de outros holons. Um holon deve cumprir 3 condições: ser estável, ter capacidade de ser autônomo e capaz de ser cooperativo. Em outras palavras, um holon é: uma unidade autônoma que age de acordo com suas próprias diretrizes comportamentais; uma unidade superordenada que respeita os componentes das partes que a transcende; uma unidade subordinada, na medida que faz parte de um todo. Assim, hólons são estruturas similares e podem estar conectados numa holarquia. A Fig. 2.1 apresenta um exemplo de holarquia com quatro níveis. No exemplo, podemos notar dois tipos de comunicação: intra-níveis (entre holons de mesmo nível, horizontal) e inter-níveis (entre holons de níveis diferentes, vertical). Uma leitura mais aprofundada sobre holons, seus conceitos e suas aplicações no contexto de sistemas de transporte pode ser encontrada em (TCHAPPI et al., 2020).

No presente trabalho, entretanto, optamos por não decompor o problema em problemas menores, não dar controle total aos agentes de níveis mais altos e não possuir agentes independentes uns dos outros. Usando os conceitos citados como inspiração, organizamos os agentes em uma organização hierárquica com um número arbitrário de níveis, tendo agentes supervisores de diferentes níveis encarregados de subordinados. Na organização proposta, há uma transferência de informações dos subordinados ao seus

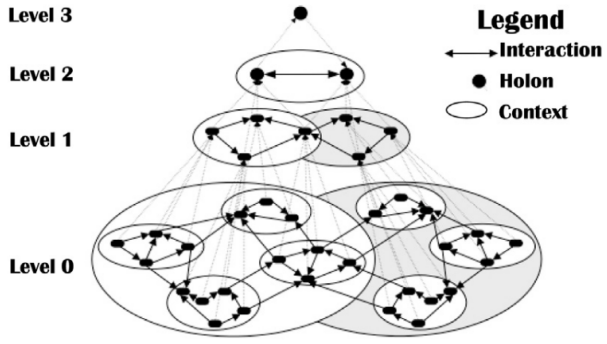



Figura 2.1 – Exemplo de holarquia (GAUD, 2007)

supervisores e os supervisores fazem uma recomendação para seus subordinados. Essa recomendação pode ou não ser seguida pelos subordinados, implicando ou não em um incentivo nas suas recompensas. Assim, agentes de níveis hierárquicos mais altos, tendo uma visão mais ampla do problema por considerarem as informações de seus subordinados, podem guiar eles a ações que, alinhadas com a recomendação, promovem um melhor desempenho coletivo.

## 2.6 Sistemas de transporte e Controle Semafórico

Controle semafórico é um dos métodos mais tradicionais de controle de tráfego. Como dito na Seção 1.1, é necessário uma utilização mais eficiente de uma rede de transporte visto que alterá-la é muito custoso. Existem três tipos de controle semafórico: tempo-fixo, semi-atuado e atuado. Contudo, precisamos, previamente, explicar alguns conceitos sobre sistemas de transporte para entendermos melhor sobre controle semafórico.

### 2.6.1 Sistemas de transporte

Sistemas de transporte podem ser modelados contendo duas partes: oferta e demanda. A oferta representa a infraestrutura da rede de transporte (rede viária, controladores semafóricos, detectores, entre outros) e pode ser representada como um grafo $G = (V, A)$, onde $V$ é o conjunto de vértices, representando as intersecções, e $A$ é o conjunto de arestas, representando os segmentos de vias, como ilustrada na Fig. 2.2.

Os seguintes elementos são componentes de redes viárias e são ilustrados na Fig. 2.3:

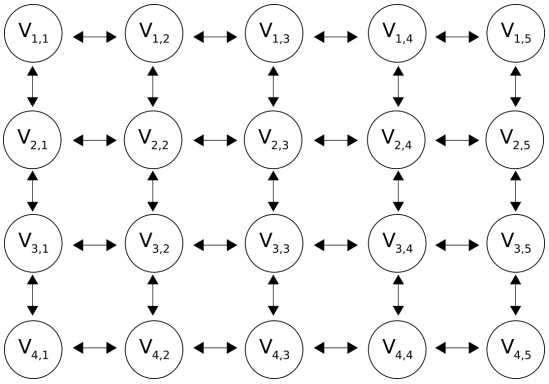



Figura 2.2 – Exemplo de grafo de uma rede de transporte em grid 4x5 - as arestas bidirecionais representam duas arestas em sentidos opostos

- Faixa de trânsito - espaço para circulação de veículos num único sentido de fluxo;
- Pista - conjunto de faixas de trânsito;
- Via - conjunto de pistas que podem permitir fluxo em sentido único ou duplo;
- Interseção - locais onde ocorrem cruzamento entre vias, podem haver conflitos entre os sentidos de fluxo dos veículos.

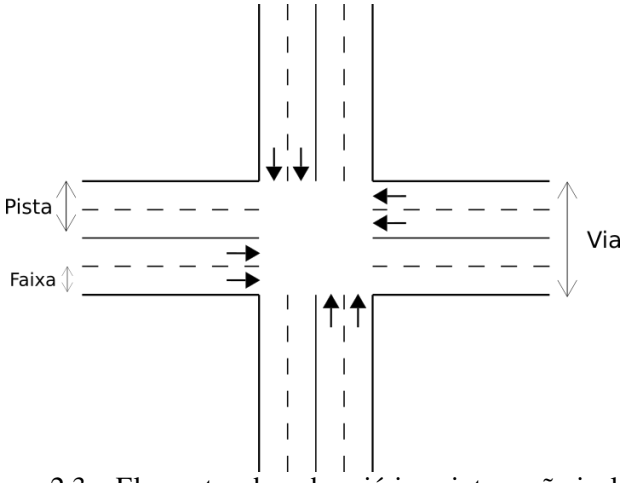



Figura 2.3 – Elementos de redes viárias - interseção isolada

A demanda representa os usuários do sistema de transporte, incluindo veículos particulares, transporte público, pedestres, ciclistas, entre outros. Esses usuários podem ser modelados como entidades que viajam de um vértice de origem $o \in V$ para um vértice de destino $d \in V$ na rede viária. Assim, cada viajante está associado a um par OD (origem-destino) e a uma rota (caminho) pelo qual viajará. No presente trabalho, os viajantes são formados por veículos particulares cujos pares OD e rotas são fixos.

### 2.6.2 Controle Semafórico

O controle semafórico é um método de sinalização que ajuda a gerenciar o tráfego nas interseções de uma rede viária. De acordo com Roess, Prassas e McShane (2011), o componente principal de um controlador semafórico é seu ciclo: esquema de temporização das diferentes fases do controlador. O tempo de ciclo é o tempo necessário para a rotação completa de todas as suas fases. Uma fase é uma configuração de sinais que permite o fluxo de tráfego em determinadas faixas da interseção e, por consequência, proíbe o fluxo em outras. A Fig. 2.4 ilustra diferentes fases em uma interseção e a Fig. 2.5 ilustra um ciclo e suas fases.

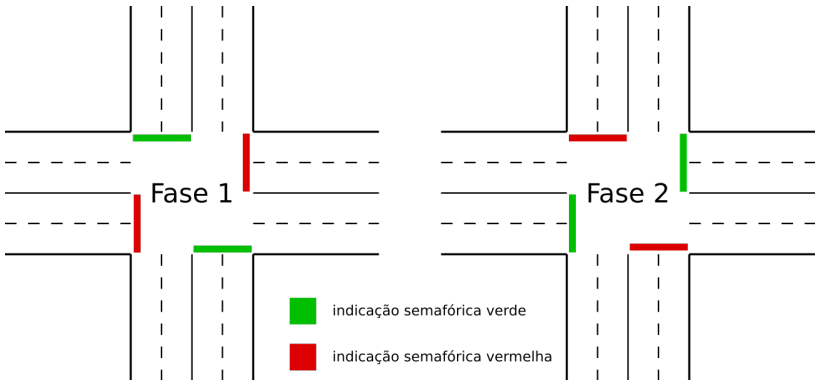



Figura 2.4 – Exemplo das fases de uma interseção. Fase 1 - libera o tráfego na direção Norte-Sul. Fase 2 - libera tráfego na direção Leste-Oeste

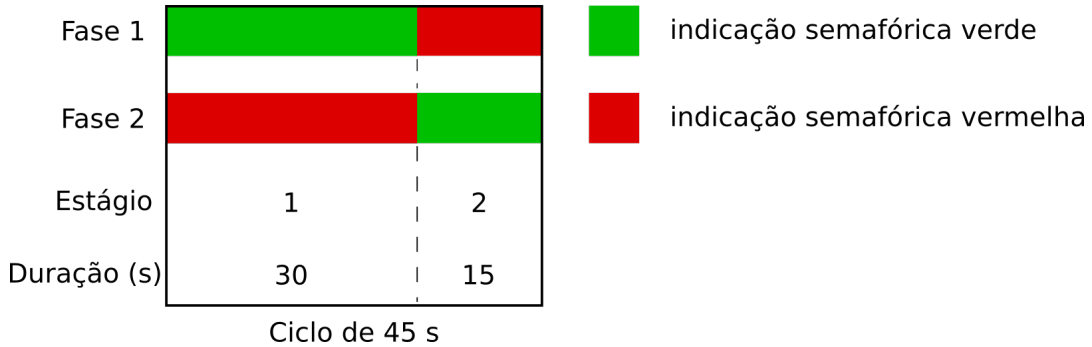



Figura 2.5 – Exemplo do ciclo de 45 segundos composto por duas fases, 1 e 2. Fase 1 sendo a primeira fase do ciclo com 30 segundos de verde e a fase 2 com 15 segundos de verde

Os diferentes tipos de controle semafóricos estão relacionados aos tempos de ciclo e a divisão das fases (tempo associado a cada fase) do controlador. São eles:

- Tempo fixo - o tempo de ciclo e a divisão das fases são fixos. Uma vez feita a programação do controlador, as mesmas configurações ocorrerão na interseção até que outras configurações sejam feitas manualmente ou outra configuração de duração fixa seja selecionada de acordo com o horário do dia. É um tipo não responsivo,

não se ajustando ao tráfego, contudo, não utiliza detectores, sendo menos custoso de ser implementado nas redes de transporte;

- Semi-Atuado - o tempo de ciclo é fixo e o tempo de cada fase é variável. O uso de detectores permite o gerenciamento do tempo de cada fase, oferecendo mais flexibilidade. Os tempos das fases podem ser encurtados ou alongados dependendo das detecções no fluxo de tráfego;

- Atuado - o ciclo, o tempo de ciclo e as durações das fases são variáveis. São definidos tempos máximos e mínimos de cada fase e a duração da fase é determinada pelo número de veículos que passam pelos detectores. O primeiro veículo a passar garante que o tempo mínimo será dado a determinada fase, e as detecções subsequentes estendem a duração da fase até o limite máximo.

No presente trabalho, os controladores semafóricos são atuados, possuindo ciclo e fases de tempos variáveis. Os controladores e diferentes agentes descritos nos Capítulos 4 e 5 buscam, por meio de aprendizado por reforço, encontrar as melhores fases e suas respectivas durações para as diferentes situações simuladas nos experimentos.

## 2.7 Simulação de Tráfego

A utilização de simuladores de tráfego é necessária, visto que os sistemas de transportes são sistemas complexos e realizar testes ou experimentos em situações reais é custoso, demandam tempo e podem apresentar riscos. A complexidade dos sistemas de transporte é devida às interações entre suas múltiplas entidades de comportamentos variados (veículos de diferentes tipos, pedestres, controladores de tráfego, entre outras) e, na simulação de tráfego, os modelos podem ser classificados conforme o nível de detalhe da representação do sistema. Entre os modelos de simulação, destacam-se:

- Macroscópico - o modelo de maior abstração. O modelo analisa o sistema de tráfego utilizando modelos matemáticos de mecânica de fluidos. Assim, não existem modelos para cada veículo simulado, mas fluxos de tráfego que informam volumes, densidades e velocidades. Estes modelos são computacionalmente eficientes e úteis para predições de valores mais grosseiros (menor granularidade);

- Mesoscópico - o modelo intermediário. Modela as entidades com razoável nível de detalhes, mas abstrai suas interações. Analisa as entidades em grupos, nos quais

elas são consideradas homogêneas, utilizando, por exemplo, dinâmicas de pelotão de veículos;

- Microscópico - o modelo com maior nível de detalhes das entidades e suas interações. Considera as dinâmicas de tráfego de cada indivíduo. São modelos mais complexos e exigem mais parâmetros de configuração.

No presente trabalho é utilizado um modelo de simulação microscópico visto que é necessário simular os veículos em alto nível de detalhe para realizar o aprendizado dos controladores semafóricos e dos demais agentes hierárquicos, detalhados no Capítulo 4. Para isso, nos experimentos, utilizamos o simulador de tráfego microscópico SUMO, *Simulation of Urban Mobility* (LOPEZ et al., 2018).

# 3 TRABALHOS RELACIONADOS

Este capítulo discute trabalhos que também lidam com controle semafórico como forma de melhorar o fluxo de tráfego. Começamos elencando brevemente as técnicas utilizadas nesse contexto (Seção 3.1). Concentramos a discussão em trabalhos que utilizam aprendizado por reforço em controle semafórico (Seção 3.2). Por fim, comparamos trabalhos que utilizam aprendizado em uma organização hierárquica em controle semafórico como o trabalho proposto nesta dissertação (Seção 3.3).

## 3.1 Técnicas Utilizadas em Controle Semafórico

Controle semafórico, sendo uma ferramenta importante para o gerenciamento de tráfego, vem sendo estudado desde a metade do século passado (ZHAO; DAI; ZHANG, 2012). Nessa perspectiva, técnicas clássicas de engenharia de tráfego foram desenvolvidas para amenizar o problema de tráfego, entre elas podemos citar: método de Webster (WEBSTER, 1958), SCATS (LOWRIE, 1982) e SCOOTS (HUNT et al., 1981) .

Com o avanço da tecnologia, o controle semafórico também sofreu mudanças, técnicas que utilizam inteligência computacional foram desenvolvidas para esse contexto. Técnicas como sistemas *Fuzzy* (PAPPIS; MAMDANI, 1977), redes neurais (CHOY; SRINIVASAN; CHEU, 2006), computação evolutiva (CEYLAN; BELL, 2004), entre outras, que estão além do escopo deste trabalho. Caso o leitor tenha curiosidade sobre o assunto, recomenda-se a leitura do *survey* (ZHAO; DAI; ZHANG, 2012).

Das diferentes técnicas utilizadas em controle semafórico, nos focaremos nos trabalhos que utilizam aprendizado por reforço.

## 3.2 Aprendizado por Reforço em Controle Semafórico

Na área de aprendizado por reforço, existem inúmeras maneiras de modelar o problema de controle semafórico. Caso o leitor tenha curiosidade sobre diferentes modelagens, recomenda-se a leitura dos seguintes *surveys*: (WEI et al., 2019b), (YAU et al., 2017) e (BAZZAN, 2009). Discutiremos, nesta seção, trabalhos que modelam os controladores semafóricos de maneira semelhante ao modelo de controladores utilizado nesta dissertação. Como os controladores semafóricos serão a base da organização hierárquica

proposta, os trabalhos discutidos nesta seção não serão comparados de forma qualitativa, apenas serão discutidas suas semelhanças com o modelo local proposto.

Em (ASLANI; MESGARI; WIERING, 2017), foi realizado um trabalho com controladores adaptativos usando um aprendizado ator-crítico no semáforos, chamados de controladores A-CATs (*actor-critic adaptive traffic signal*). O trabalho foi realizado com o objetivo de analisar o desempenho do aprendizado ator-crítico discreto e contínuo, e analisar os efeitos de diferentes interrupções de tráfego (cruzamento de pedestres, congestionamento, ruído dos sensores) no comportamento desses controladores. É feita uma simulação do centro da cidade de Teerã e os controladores são testados em 6 diferentes cenários a fim de se encontrar a configuração mais robusta para os A-CATs.

Em (ASLANI et al., 2018), o trabalho busca comparar a robustez, o desempenho e a velocidade de aprendizado de sete diferentes algoritmos de aprendizado por reforço em controle semafórico: versões discretas e contínuas de Q-Learning, Sarsa e ator-crítico e uma versão chamada *residual actor-critic*, que utiliza gradiente descendente. Novamente, é utilizada uma simulação de Teerã por conter uma variedade de interrupções e um espaço de estado-ação com grande dimensionalidade. São feitos quatro experimentos, resultando em um melhor desempenho do método com ator-crítico contínuo.

Em (CHU et al., 2019), o trabalho apresenta um algoritmo de MARL descentralizado e escalável no contexto de ATSC (*adaptive traffic signal control*) para o agente estado-da-arte de aprendizado por reforço profundo (*deep reinforcement learning*): ator-crítico de vantagem (*advantage actor-crtic*, A2C). São realizados experimentos em uma rede de *grid* e em uma rede baseada na cidade de Mônaco, mostrando um melhor desempenho do algoritmo apresentado sobre outros algoritmos estado-da-arte de MARL descentralizados.

Em (MANNION; DUGGAN; HOWLEY, 2016), os autores avaliam três diferentes modelos de aprendizado por reforço com controle semafórico baseados em Q-Learning. O primeiro tendo como recompensa o tamanho da fila de carros; o segundo tendo como recompensa o tempo de espera no semáforo; o último tendo como recompensa uma função que é a soma sem pesos das recompensas do primeiro e do segundo modelo. Ao final, o algoritmo que possui o melhor desempenho, nas condições de experimentação (fluxo fixo), é o segundo.

Em (NISHI et al., 2018), os autores apresentam um controle semafórico baseado em redes grafo-neurais convolucionais (GCNN, *graph convolutional neural networks*). Essa abordagem permite a extração automática das características geométricas das vias

utilizando múltiplas camadas de redes neurais, diferentemente dos métodos convencionais, onde a extração dessas características são feitas de maneira manual. Os resultados demonstram que essa abordagem gera políticas duas vezes mais rápido e pode se adaptar melhor às trocas de demanda no tráfego.

Em (WEI et al., 2018), os autores apresentam o IntelliLight, que é uma abordagem de deep RL (aprendizado por reforço profundo) para controle semafórico. Diferente das redes neurais convencionais, o modelo utiliza uma sub-estrutura chamada *Phase Gate* que faz o processo de aprendizado ser diferente para cada fase. Nos experimentos, o desempenho do método é superior aos métodos de estado-da-arte citados.

Em (WEI et al., 2019b), o trabalho propõe um modelo de aprendizado por reforço que utiliza *graph attention networks* para facilitar a comunicação entre os semáforos. O modelo, chamado CoLight, incorpora as influências espaciais e temporais das interseções vizinhas no aprendizado de um controlador semafórico. Nos experimentos, é demonstrado que o modelo, utilizando essa comunicação, alcança melhor desempenho do que outros métodos estado-da-arte.

Em (ZHENG et al., 2019a), o trabalho propõe um modelo de Deep Q-Learning chamado de FRAP que se baseia em dois princípios. O primeiro, da competição entre fases, diz que quando dois sinais estão em conflito, deve ser dado prioridade para aquele que possuir maior demanda (movimento de tráfego). O segundo, da invariância, diz que o controle de sinais deve ser invariável à simetrias como rotação e *flipping* (trocas de sentidos simétricas). Baseado nesses princípios, o aprendizado de diferentes movimentos de tráfego e fases ocorrem ao mesmo tempo, aumentando a eficiência no aprendizado e o desempenho.

Em (ZHENG et al., 2019b), os autores propõem um método denominado LIT, uma abordagem de aprendizado por reforço baseado na teoria e métodos de controle semafórico clássicos de área de transporte. LIT, ou Light-Intellight, é baseado no trabalho de (WEI et al., 2018), utilizando uma diferente modelagem de estado e recompensa embasado na teoria de transporte. Nos experimentos, o modelo alcança desempenho superior a outros trabalhos citados.

A Tab. 3.1 compara as definições dos modelos utilizados nos diferentes trabalhos apresentados de aprendizado por reforço em controle semafórico, colocando em evidência se os trabalhos: utilizam as informações de tamanho de fila e volume de veículos, fase atual e sua duração em suas definições de estado; se suas ações são escolher a próxima fase do controlador semafórico; e se suas recompensas dependem do tamanho da fila e do

tempo de espera dos veículos.

| Trabalho | Estado | | | | Ação | Recompensa | | Simulador |
|---|---|---|---|---|---|---|---|---|
| | Tamanho da fila | Volume | Fase | Duração da fase | Escolher próxima fase | Tamanho da fila | Tempo de espera | |
| (ASLANI; MESGARI; WIERING, 2017) | ✓ | ✓ | ✓ | | | ✓ | | AIMSUN |
| (ASLANI et al., 2018) | ✓ | ✓ | ✓ | | | ✓ | | AIMSUN |
| (CHU et al., 2019) | ✓ | | | | ✓ | ✓ | ✓ | SUMO |
| (MANNION; DUGGAN; HOWLEY, 2016) | ✓ | | ✓ | ✓ | | ✓ | ✓ | SUMO |
| (NISHI et al., 2018) | ✓ | | | | ✓ | | ✓ | SUMO |
| (WEI et al., 2018) | ✓ | ✓ | ✓ | | | | | SUMO |
| (WEI et al., 2019a) | ✓ | | | | ✓ | ✓ | | SUMO |
| (ZHENG et al., 2019a) | ✓ | | ✓ | | ✓ | ✓ | | SUMO |
| (ZHENG et al., 2019b) | ✓ | | ✓ | | ✓ | ✓ | | SUMO |
| Modelo local desta dissertação | ✓ | | | | ✓ | | ✓ | SUMO |

Tabela 3.1 – Tabela comparativa dos trabalhos de aprendizado por reforço em controle semafórico

Observando a Tab. 3.1, é possível notar que o modelo local desta dissertação utiliza características semelhantes às de trabalhos recentes na área para realizar o aprendizado nos controladores semafóricos. Como o modelo local dos controladores é um módulo da organização hierárquica proposta nesta dissertação, são incentivadas, como trabalhos futuros, possíveis substituições deste modelo com modelos semelhantes - como os discutidos nesta seção.

## 3.3 Aprendizado com Organização Hierárquica em Controle Semafórico

Como visto no Seção 2.4, o aprendizado por reforço multiagente (MARL) pode ser modelado como um jogo estocástico. Nessa perspectiva, temos o problema da maldição

da dimensionalidade, que se refere ao crescimento exponencial do espaço de estados-ações discreto devido ao número de estados e variáveis ações (dimensões) dos diferentes agentes. Uma das diferentes técnicas usadas para simplificar esse problema é utilizar uma organização hierárquica: organizar o conjunto de agentes sobre uma ordem de prioridade, criando graus sucessivos de poderes e responsabilidades.

A seguir, discutiremos trabalhos que utilizam uma organização hierárquica no contexto de controle semafórico. As Tabelas 3.2 e 3.3 fazem o resumo e comparam os trabalhos citados à esta dissertação respectivamente.

Em (ABDOOS; MOZAYANI; BAZZAN, 2013) é utilizada uma organização chamada holonic multi-agent system para modelar uma rede de transporte. A hierarquia é dividida em super-holons e seus subordinados, sub-holons. Neste trabalho, os agentes em níveis superiores (super-holons, regiões) recebem uma abstração dos estados dos agentes inferiores (sub-holons, interseções) por meio de mensagens e podem, assim, definir uma ação a tomar. Essa ação superior restringe o conjunto de ações dos agentes em nível inferior. Após os agentes inferiores tomarem suas ações, os super-holons recebem a recompensa do ambiente e a repassam para seus subordinados. Os experimentos mostram que o método apresentado tem um desempenho melhor do que uma abordagem que utiliza apenas Q-Learning em nível local. Segundo o artigo, é possível estender os níveis hierárquicos se um novo design levar os novos níveis em consideração.

Em (ABDOOS; MOZAYANI; BAZZAN, 2014) uma hierarquia de dois níveis baseada em Q-Learning é apresentada. Em nível inferior, os agentes aprendem por Q-Learning clássico. Em nível superior, os agentes utilizam aproximação de função com *tile coding* ((SUTTON; BARTO, 2018) - Seção 9.5.4), visto que eles recebem as informações de estados dos seus agentes subordinados, o que caracteriza um espaço de estados impraticável de ser representado de maneira tabular (*tabelas-Q*). As ações dos agentes superiores restringem as ações dos inferiores. Os experimentos mostram um melhor desempenho comparado a utilizar apenas Q-Learning em nível local. Devido à grande quantidade de informações do segundo nível, é difícil aumentar a quantidade de níveis hierárquicos.

A hierarquia apresentada em (BAZZAN; OLIVEIRA; SILVA, 2010) é constituída por agentes supervisores e agentes supervisionados. Os supervisores, durante o período de treinamento de seus agentes, coletam informações sobre as ações tomadas e recompensas recebidas de seus supervisionados. Após o treinamento, essas informações coletadas são utilizadas pelo supervisor para sugerir aos seus agentes qual a melhor ação a ser tomada. Nessa abordagem, não há aprendizado no nível superior, apenas uma coleta de

informações. Como as informações do nível superior escalam de maneira exponencial com o número de agentes subordinados - visto que os superiores levam em consideração as informações de estado, ação e recompensa - não há possibilidade de aumentar os níveis hierárquicos devido ao custo computacional.

A rede de transporte é dividida em 3 níveis hierárquicos em (CHOY; SRINIVASAN; CHEU, 2003): controladores de interseção, controladores de zona e controladores de região. Cada controlador é um agente implementado com redes neurais. O interessante dessa hierarquia é que cada agente em nível inferior, além de passar seu estado, passa um fator de cooperação para o nível superior, que utiliza esse dado no cálculo de sua ação. A recompensa recebida do ambiente é propagada do nível superior ao inferior, ajustando os pesos das redes neurais.

Em (FRANCE; GHORBANI, 2003) e (ROOZEMOND, 2001) as hierarquias possuem dois níveis: em (FRANCE; GHORBANI, 2003), os agentes superiores recebem as ações de seus subordinados, as modificam e as enviam novamente ao nível inferior; em (ROOZEMOND, 2001), os agentes superiores alocam papéis e regras aos seus subordinados. Contudo, em ambos os artigos, não é discutido como os agentes superiores atuam, apenas que fazem suas decisões buscando o ótimo do sistema.

O trabalho apresentado em (MA; WU, 2020) é o mais semelhante ao proposto nesta dissertação. Nesse trabalho, os autores estendem o algoritmo de aprendizado por reforço profundo MA2C (CHU et al., 2019) com ideias de *feudal learning* (DAYAN; HINTON, 1993). A rede de transporte é dividida em regiões e os agentes divididos em trabalhadores, que controlam as interseções, e gerentes, que controlam as regiões. Nessa hierarquia, os trabalhadores comunicam suas observações para os seus gerentes, que abstraem informações dessas observações e definem um objetivo. Esse objetivo é comunicado aos trabalhadores subordinados para que eles ajam levando-o em consideração. As recompensas de cada agente são definidas sobre diferentes informações das interseções e das regiões, e os agentes trabalhadores recebem um aumento na sua recompensa conforme uma relação entre seu estado e objetivo do seu gerente. As principais diferenças desse trabalho ao apresentado na dissertação, além do algoritmo de aprendizado, são as definições dos agentes que controlam regiões, as definições do incentivo e o número de níveis hierárquicos utilizados nos experimentos.

Recentemente, o trabalho realizado em (ABDOOS; BAZZAN, 2021) apresenta uma hierarquia de dois níveis que utiliza memória longa de curto prazo (LSTM - *long short-term memory*) para prever o tráfego no nível superior. A rede de transporte é di-

vidida em regiões e, usando previsão de tráfego, os agentes região tentam encontrar a melhor ação conjunta para os agentes locais (controladores) dentro da sua região. Os agentes locais utilizam um mecanismo de limite para escolher entre seguir a ação conjunta recomendada ou seguir sua própria política. Os experimentos realizados mostram um menor tempo de espera dos veículos usando a hierarquia proposta quando comparada a um método de tempo fixo e a um método de aprendizado por reforço.

A Tab. 3.2 faz o resumo dos trabalhos mencionados que utilizam aprendizado por reforço com organização hierárquica em controle semafórico destacando: o algoritmo usado nos agentes que controlam as interseções; o algoritmo usado nos agentes de diferentes níveis hierárquicos; as informações utilizadas pelos agentes hierárquicos; a interação entre os agentes de diferentes níveis; o número de níveis hierárquicos no experimento e as redes utilizadas nos experimentos.

| Trabalho | Algoritmo dos agentes nas interseções | Algoritmo dos agentes hierárquicos | Informações dos agentes hierárquicos | Interação entre os agentes de diferentes níveis | Número de níveis hierárquicos no experimento | Redes utilizadas no experimento |
|---|---|---|---|---|---|---|
| (ABDOOS; MOZAYANI; BAZZAN, 2013) | Q-Learning | Holonic Q-Learning | Densidade de veículos nas faixas dos agentes subordinados | Nível superior restringe ações e repassa recompensa | 2 | Rede sintética de 50 interseções |
| (ABDOOS; MOZAYANI; BAZZAN, 2014) | Q-Learning | Aproximação de Função + tile coding | Fila média de veículos dos agentes subordinados | Nível superior restringe ações | 2 | Grid 3x3 |
| (BAZZAN; OLIVEIRA; SILVA, 2010) | Q-Learning | Algoritmo de observação de recompensas | Média das recompensas dos agentes subordinados | Supervisor sugere ações | 2 | Grid 8x8 |
| (CHOY; SRINIVASAN; CHEU, 2003) | Redes Neurais | Redes Neurais | Estado dos agentes subordinados e fator de cooperação | Níveis mais altos calcula política com maior recompensa e prioridade | 3 | Singapore Central Business District |
| (FRANCE; GHORBANI, 2003) | - | - | Ações dos agentes subordinados | Nível superior calcula melhor ação para os subordinados | 2 | Grid 3x2 |
| (ROOZEMOND, 2001) | - | - | - | Nível superior aloca papéis e regras aos agentes subordinados | - | - |
| (MA; WU, 2020) | MA2C | FMA2C | Filas dos agentes que limitam a região controlada | Objetivo superior e aumento na recompensa dos subordinados | 2 | Grid 4x4 e cenário real (Moscou) |
| (ABDOOS; BAZZAN, 2021) | Q-Learning | LSTM para predição de tráfego | Filas e tempo de delay nas faixas dos agentes subordinados, e a ação tomada pelos subordinados | Nível superior escolhe a melhor ação conjunta dos subordinados | 2 | Grid 4x4 |
| Esta Dissertação | Sarsa com aproximação de função | Organização Hierárquica baseada em vetores (VHO - Capítulo 4) | Resultantes dos agentes subordinados | Recomendação dos supervisores de níveis mais altos com influência na recompensa dos subordinados | 3 | Grid 4x4 |

Tabela 3.2 – Resumo dos trabalhos de aprendizado por reforço com organização hierárquica

A Tab. 3.3 compara os trabalhos que utilizam organização hierárquica nos seguintes pontos:

- Se os agentes em níveis mais altos não precisam conhecer o conjunto de ações dos agentes subordinados;
- Se os agentes de níveis mais altos consideram as informações de todos seus subordinados;
- Se é possível aumentar os níveis hierárquicos, além dos citados nos trabalhos, sem novos projetos ou complicações;
- Se os critérios para a escolha da ação dos agentes de níveis mais altos são detalhados de maneira clara;
- Se há aprendizado nos diferentes níveis hierárquicos, além do nível local.

| **Trabalho** | **Agentes de nível mais alto não precisam ter conhecimento das ações dos seus subordinados** | **Agentes de nível mais alto levam em consideração informações de todos os subordinados** | **Possibilidade de número arbitrário de níveis** | **Procedimento de escolha da ação do agente de nível mais alto é detalhado** | **Aprendizado nos diferentes níveis** |
|---|---|---|---|---|---|
| (ABDOOS; MOZAYANI; BAZZAN, 2013) | X | V | X | V | V |
| (ABDOOS; MOZAYANI; BAZZAN, 2014) | X | V | X | V | V |
| (BAZZAN; OLIVEIRA; SILVA, 2010) | X | V | X | V | X |
| (CHOY; SRINIVASAN; CHEU, 2003) | X | V | X | X | V |
| (FRANCE; GHORBANI, 2003) | - | X | X | X | X |
| (ROOZEMOND, 2001) | - | X | X | X | X |
| (MA; WU, 2020) | V | X | V | V | V |
| (ABDOOS; BAZZAN, 2021) | X | V | X | V | V |
| Esta dissertação | V | V | V | V | V |

Tabela 3.3 – Tabela comparativa dos trabalhos de aprendizado por reforço com organização hierárquica

Observando a Tab. 3.3, é possível notar que o modelo de organização hierárquica proposta para controle semafórico desta dissertação completa algumas lacunas deixadas por trabalhos realizados na área.

Na maioria dos trabalhos que utilizam organização hierárquica para RL em controle semafórico, os agentes de nível mais alto precisam ter conhecimento das ações dos seus subordinados, pois definem quais ações seus subordinados irão realizar, e levam as informações de todos seus subordinados em consideração para encontrar as melhores ações coletivas. Essas características dificultam o potencial de escalabilidade desses trabalhos, visto que quanto mais níveis hierárquicos, mais informações devem ser consideradas pelos agentes.

O diferencial da organização hierárquica proposta nesta dissertação é utilizar métodos de abstração tanto na coleta das informações dos subordinados, quanto na recomendação de suas ações, para que os agentes de níveis mais altos possam considerar as informações de todos seus subordinados e influenciar em suas ações, mesmo sem conhecê-las. As abstrações permitem que isso seja feito de maneira simples e facilitam o potencial de escalabilidade do método.

# 4 ABORDAGEM PROPOSTA

Este capítulo detalha uma estrutura genérica com organização hierárquica para aprendizado por reforço e uma aplicação dessa organização para o controle semafórico em redes de transporte. Nos modelos apresentados, os agentes buscam utilizar as informações dos diferentes níveis hierárquicos para encontrar políticas que obtenham um melhor desempenho coletivo. A organização hierárquica proposta é apresentada de forma genérica na Seção 4.1 e sua aplicação para o controle semafórico e seus detalhes são apresentados na Seção 4.2.

## 4.1 Organização Hierárquica Proposta

A organização hierárquica proposta é inspirada nas ideias do aprendizado hierárquico (DIETTERICH, 1999), aprendizado feudal (DAYAN; HINTON, 1993) e aprendizado holônico (GERBER; SIEKMANN; VIERKE, 1999). Um agente região $a$ de nível hierárquico arbitrário $l$, $R_a^l$, possui $S^{-a} = \{A_1, A_2, \cdots, A_s\}$ subordinados de nível $l-1$ e é responsável por uma região como ilustrado pela Fig. 4.1. Na hierarquia proposta podemos ter um número arbitrário de níveis hierárquicos, logo os agentes região podem ser supervisores tanto dos agentes em nível zero (nível inferior) da hierarquia quanto de outros agentes região de diferentes níveis como ilustrado pela Fig. 4.2. Os agentes subordinados passam informações processadas ($P$) de seus estados para seus supervisores. Essa informação pode ser qualquer combinação de funções aplicadas sobre as observações parciais dos agentes subordinados, a fim de realizar uma abstração dessas observações. Assim, os estados dos agentes região são compostos pelas informações processadas $p \in P$ dos seus subordinados, conforme Eq. 4.1:

$$s = [p_{A_1}, ..., p_{A_s}] \tag{4.1}$$

Caso esses agentes região sejam subordinados a outros agentes região de níveis mais altos, uma recomendação do seu supervisor ($\mathcal{A}^+$) também compõe seu estado. Assim, esse se modifica, conforme Eq. 4.2:

$$s = [p_{A_1}, ..., p_{A_s}, \mathcal{A}^+] \tag{4.2}$$

Baseado no seu estado atual e no seu aprendizado, os agentes região escolhem

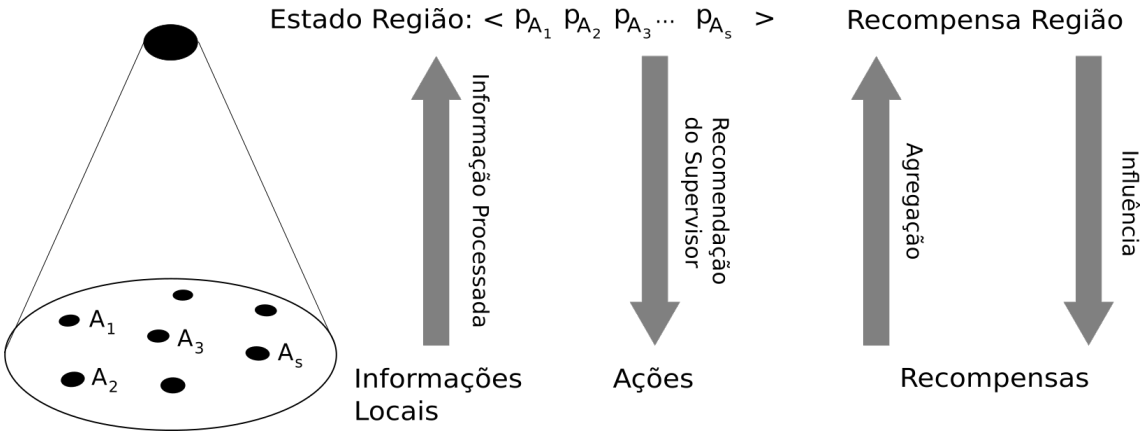



Figura 4.1 – Organização Hierárquica Proposta - Um agente hierárquico e seus subordinados

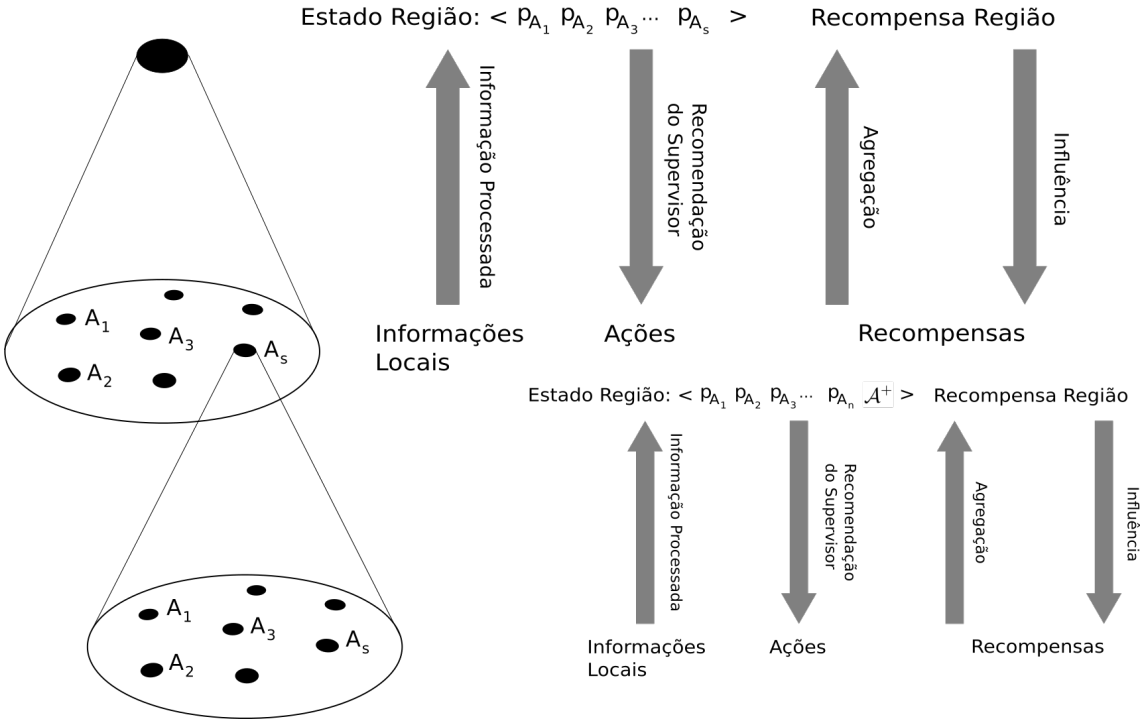



Figura 4.2 – Organização Hierárquica Proposta - Visão geral da hierarquia

suas ações, que servirão de recomendação para seus subordinados. A única restrição em relação a essas ações é que elas devem estar relacionadas às ações dos subordinados de maneira a concluir se um agente subordinado seguiu ou não sua recomendação. Assim, um agente região não restringe as ações dos agentes subordinados, mas passa uma recomendação abstrata a ser seguida.

Na organização hierárquica proposta, as recompensas dos agentes região são agregações das recompensas de seus subordinados, logo um agente região é dependente do desempenho de seus subordinados. Sendo $r_s$ a recompensa de um subordinado $s \in S^{-a}$ e $\mathfrak{F}$ uma agregação qualquer (somatório, média aritmética simples ou ponderada, valor máximo, entre outras), a recompensa do agente região supervisor $r_{R_a^l}$ é definida pela Eq. 4.3:

$$r_{R_a^l} = \underset{s \in S^{-a}}{\mathfrak{F}} (r_s) \tag{4.3}$$

Além disso, uma vez que é possível concluir se um agente subordinado seguiu a recomendação do seu supervisor, um supervisor influencia nas recompensas dos seus subordinados, podendo incentivar os que seguiram a sua recomendação, aumentado suas recompensas, e punir os que não seguiram, diminuindo-as. Assim, os agentes subordinados, por considerarem a recomendação dos seus supervisores no seu aprendizado e serem influenciados a segui-la, buscam uma melhora coletiva e não apenas individual.

Em relação aos conceitos apresentados na Seção 2.5, a organização hierárquica proposta se diferencia nos seguintes pontos:

- Diferente do aprendizado hierárquico (DIETTERICH, 1999), na organização proposta, os agentes subordinados podem ser, mas não são necessariamente, designados a subtarefas de uma tarefa maior. Os diferentes agentes da hierarquia proposta podem ter tarefas com o mesmo objetivo ou objetivos conflitantes. Outro ponto diferente do aprendizado hierárquico é que, na organização proposta, os agentes região dependem das informações processadas e influenciam nos aprendizados e nas recompensas de seus subordinados;

- Diferente do aprendizado feudal (DAYAN; HINTON, 1993), na organização proposta, os agentes região não possuem controle total sobre seus subordinados. Eles não atribuem objetivos a eles, apenas influenciam no aprendizado e em suas recompensas por meio de recomendações. Como no aprendizado feudal, os agentes supervisores se preocupam com informações pertinentes ao seu nível, mas, na or-

ganização proposta, eles dependem das informações processadas de seus subordinados;

- Diferente da abordagem holônica (GERBER; SIEKMANN; VIERKE, 1999), na organização proposta, os agentes região são dependentes dos seus subordinados para realizar seu aprendizado. Logo, esses agentes não são autônomos, não sendo considerados hólons. Além disso, não há comunicação intra-níveis, i.e., entre agentes de mesmo nível hierárquico.

Assim, a organização hierárquica proposta é mais flexível, pois possui apenas a restrição das ações dos agentes subordinados serem relacionadas às recomendações dos supervisores. Cabe ao projetista criar as relações entre os agentes de diferentes níveis: as funções que serão utilizadas para processar as informações dos subordinados; as relações entre as recomendações e as ações dos subordinados; e, como será realizada a influência nas recompensas.

Pelo restante do trabalho, os agentes de nível hierárquico zero, que controlarão as interseções, serão chamados de agentes interseção e serão representados por $I_a$. Os agentes de diferentes níveis hierárquicos, que controlaram diferentes regiões e possuem uma visão mais coletiva do problema, continuarão sendo chamados de agentes região e representados por $R_a^l$.

## 4.2 Organização Hierárquica baseada em Vetores para Controle Semafórico - VHO

Nesta dissertação, é apresentada uma organização hierárquica baseada em vetores para controle semafórico, o VHO (*Vector-based Hierarchical Organization*). Usando como base a estrutura genérica apresentada na seção anterior, utilizamos funções que trabalham com operações vetoriais nas informações processadas passadas aos agentes supervisores, na recomendação dos supervisores e nos cálculos dos incentivos nas recompensas dos subordinados. Utilizar vetores permite, de uma maneira simples, passar importantes informações do controle semafórico para os agentes de nível hierárquico mais alto. Os vetores permitem o uso das diferentes orientações das faixas da rede de transporte, não se limitando às orientações normalmente encontradas na literatura (N, S, E, W), o que poderia permitir um melhor desempenho em redes mais complexas. Além disso, porque os vetores são estruturas simples, os agentes região podem considerar as informações de todos seus subordinados e não apenas tratar a região como um caixa preta, i.e, considerar

apenas informações dos limites de suas regiões. Diante disso, operação utilizando vetores é um método eficiente para controle semafórico utilizando a organização hierárquica proposta.

A Fig. 4.3 ilustra um exemplo de região controlada na aplicação da organização proposta para o controle semafórico. Neste exemplo, temos quatro agentes interseção $\{I_0, I_1, I_2, I_3\}$, de nível hierárquico zero, subordinados a um agente região $R_0^1$ de índice zero e nível hierárquico um.

Os conceitos base para o VHO são apresentados nas seções a seguir.

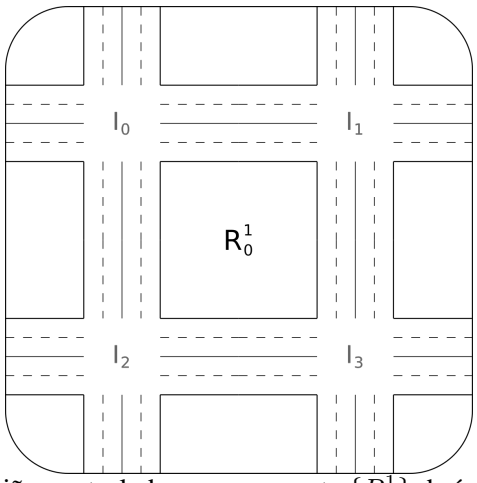



Figura 4.3 – Exemplo de região controlada por um agente $\{R_0^1\}$ de índice zero e nível hierárquico um com quatro agentes interseção subordinados $\{I_0, I_1, I_2, I_3\}$, de nível hierárquico zero.

### 4.2.1 Informação Processada

A informação processada passada entre agentes de diferentes níveis hierárquicos no VHO são vetores em coordenadas polares. Os ângulos dos vetores representam a orientação geográfica do fluxo de tráfego e as magnitudes representam a quantidade de veículos se movendo nesta direção em um instante $t$. No VHO, temos duas formas de informações processadas: os vetores resultantes de interseção e os vetores resultantes de região.

#### *4.2.1.1 Vetores resultantes de interseção*

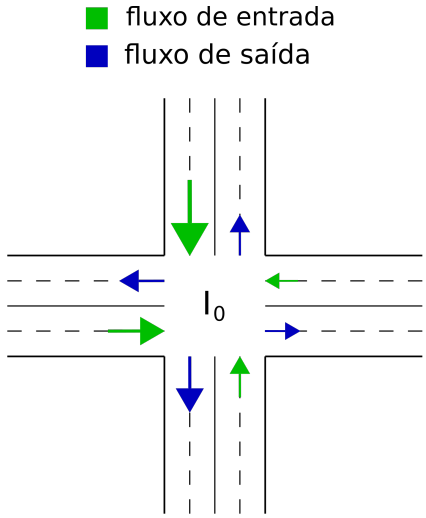
Os vetores resultantes de interseção de entrada $\vec{e}_{I,i}$ e de saída $\vec{s}_{I,i}$ de uma interseção $i \in I$ são as informações que um agente interseção passa ao seu agente supervisor. Essas resultantes são calculadas fazendo-se a soma dos vetores de faixas de entrada $\vec{e}_l$ e saída

$\vec{s}_l$, conforme equações (4.4) e (4.5), onde $L_e$ e $L_s$ são os conjuntos de faixas entrando e saindo de uma interseção.

$$\vec{e}_{I,i} = \sum_{l \in L_e} \vec{e}_l \tag{4.4}$$

$$\vec{s}_{I,i} = \sum_{l \in L_s} \vec{s}_l \tag{4.5}$$

Um vetor de faixa em coordenadas polares guarda, em sua magnitude, a quantidade de veículos passando por uma faixa em um instante $t$ e, em seu ângulo, a orientação geográfica dela. A Fig. 4.4 ilustra um exemplo de resultantes de uma interseção.




| Vetor | Magnitude (quantidade de veículos) | Ângulo (orientação da faixa) |
|---|---|---|
| ↓ | 10 | 270° |
| → | 8 | 0° |
| ↑ | 4 | 90° |
| ← | 2 | 180° |
| ↘ | 8.48 | 315° |

Soma dos vetores entrando em $I_0$ : ↘

Soma dos vetores saindo de $I_0$ : ↙

Figura 4.4 – Exemplo de vetores resultantes de uma interseção - vetores no formato (magnitude, ângulo)

#### *4.2.1.2 Vetores resultantes de região*

Os vetores resultantes de região de entrada $\vec{e}_R$ e saída $\vec{s}_R$ são as informações que um agente região passa ao seu supervisor. Essas resultantes são calculadas fazendo-se a soma dos vetores resultantes dos seus subordinados. Assim, um agente região $a$ de nível hierárquico $l$, $R_a^l$, passa ao seu supervisor os vetores $\vec{e}_{R,a}$ e $\vec{s}_{R,a}$. Caso os subordinados de $a$ sejam outras regiões ($l - 1 > 0$), as resultantes de $a$ são os somatórios das resultantes dessas regiões $\vec{e}_{R,b}$ e $\vec{s}_{R,b}$, onde $b \in R^{-a}$, conjunto dos agentes região de nível hierárquico $l - 1$ subordinados ao agente $a$. Caso os subordinados de $a$ sejam interseções ($l - 1 = 0$),

as resultantes de $a$ são os somatórios das resultantes dessas interseções $\vec{e}_{I,i}$ e $\vec{s}_{I,i}$, onde $i \in I^{-a}$, conjunto dos agentes interseção subordinados a $a$. Assim, temos as equações (4.6) e (4.7):

$$\vec{e}_{R,a} = \begin{cases} \sum_{b \in R^{-a}} \vec{e}_{R,b} & \text{se } l-1 > 0 \\ \sum_{i \in I^{-a}} \vec{e}_{I,i} & \text{se } l-1 = 0 \end{cases} \tag{4.6}$$

$$\vec{s}_{R,a} = \begin{cases} \sum_{b \in R^{-a}} \vec{s}_{R,b} & \text{se } l-1 > 0 \\ \sum_{i \in I^{-a}} \vec{s}_{I,i} & \text{se } l-1 = 0 \end{cases} \tag{4.7}$$

As Fig. 4.5 e 4.6 ilustram exemplos de vetores resultantes de região. Na Fig. 4.5 observamos as resultantes de uma região com quatro interseções subordinadas e, na Fig. 4.6, observamos as resultantes de uma região com outras quatro regiões subordinadas.

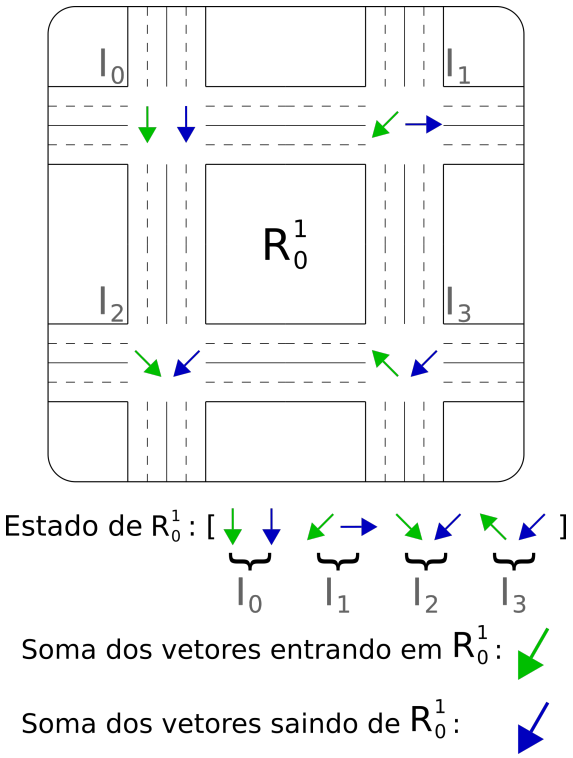



Figura 4.5 – Resultantes de uma região com quatro interseções subordinadas

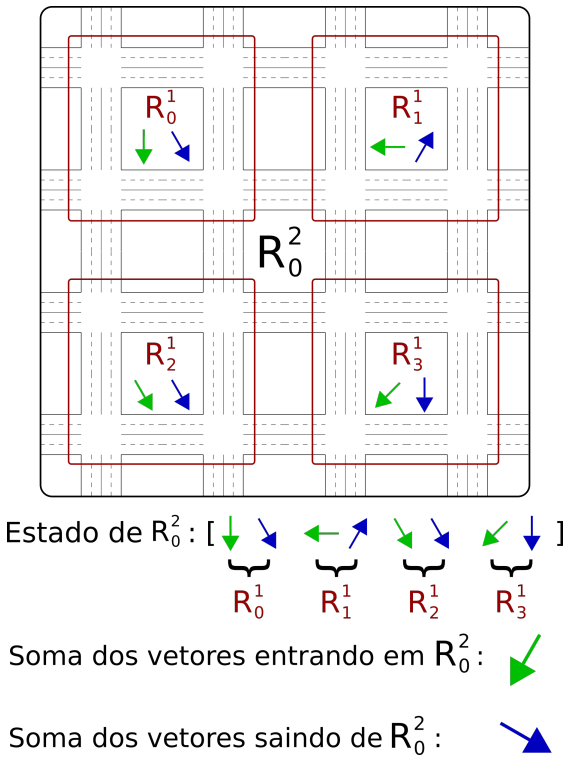



Figura 4.6 – Resultantes de uma região com quatro regiões subordinadas

### 4.2.2 Recomendação dos Supervisores

Os agentes região aprendem a indicar um fluxo de tráfego aos seus subordinados. No modelo proposto, existem oito vetores possíveis de recomendação, representando os oito pontos cardeais da rosa dos ventos (N, NE, L, SE, S, SO, O, NO). Assim, as recomendações são representadas pelos ângulos correspondentes aos pontos cardeais indicados. A Fig. 4.7 ilustra as possíveis recomendações dos supervisores.

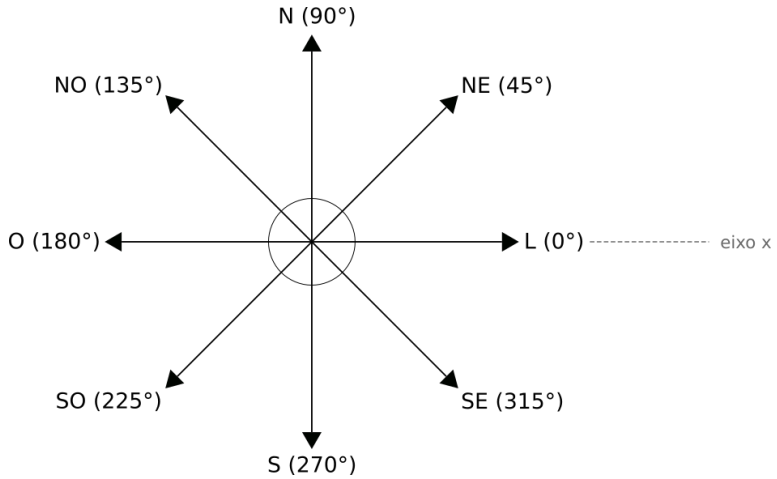



Figura 4.7 – Possíveis recomendações dos supervisores no VHO

### 4.2.3 Cálculos dos incentivos nas recompensas

Como visto na organização hierárquica genérica (Seção 4.1), os agentes subordinados podem receber um incentivo ou uma punição na suas recompensas, conforme tenham seguido ou não a recomendação de seu supervisor. No VHO, seja $b$ um agente subordinado a um agente região $R_i^l$; $r_b$ a recompensa que o agente $b$ recebeu do ambiente; $r'_b$ a nova recompensa do agente $b$ influenciada pelo agente supervisor; e $\sigma$ uma função que relaciona a ação do agente $b$ ($a_b$) e a recomendação do agente supervisor de $b$ ($\mathcal{A}^{+b}$) a um número real; o agente região $R_i^l$, supervisor de $b$, altera a recompensa do seu subordinado conforme a Eq. 4.8:

$$r'_b = r_b + r_b * \sigma(a_b, \mathcal{A}^{+b}) \tag{4.8}$$

No VHO, um agente subordinado segue a recomendação, indicação de tráfego, de seu supervisor ao liberar ou indicar um fluxo de tráfego semelhante. Assim, a função $\sigma$ é uma função cosseno entre os vetores da ação do subordinado e da recomendação do supervisor com seu domínio controlado entre $[0, 1]$. Controlamos o domínio da função, truncando seus valores negativos, para não termos punições para agentes que não seguiram a recomendação, apenas incentivos. A Fig. 4.8 ilustra um exemplo do cálculo do incentivo com a função cosseno. No exemplo, a diferença entre os ângulos da ação e da recomendação superior é de $45$ graus e seu cosseno é de $0,7$, representando um aumento de $70\%$ na recompensa do subordinado.

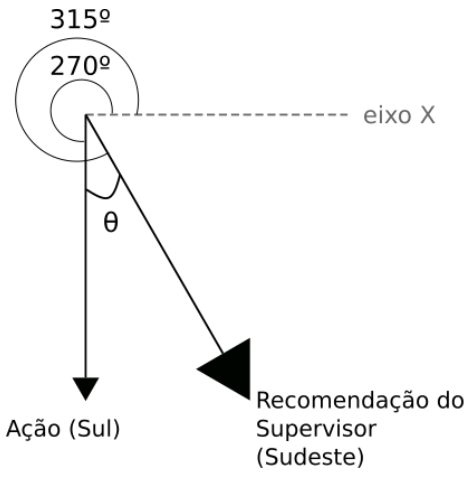



Figura 4.8 – Exemplo de cálculo de incentivo usando a função cosseno entre uma ação de um subordinado (Sul) e uma recomendação de seu supervisor (Sudeste)

Entre regiões o cálculo do incentivo acontece de maneira direta, visto que tanto

o agente região subordinado, quanto o agente região supervisor, indicam um fluxo de tráfego conforme Seção 4.2.2. Entre um agente interseção subordinado e um agente região supervisor é necessário transformar a ação do agente interseção, i.e, mapear a fase atual que libera fluxos de tráfego a um conjunto de vetores e, assim, realizar o cálculo do incentivo. A Fig. 4.9 ilustra esse processo, onde usamos as faixas de origem e de destino liberadas pela fase atual para criar vetores de fluxo de tráfego. Como uma fase de um agente interseção pode liberar o tráfego em mais de um sentido, realizamos o cálculo do cosseno com cada sentido liberado pela fase atual e utilizamos o maior valor encontrado. Logo, os agentes interseção recebem o maior incentivo possível de acordo com os fluxos de tráfego que liberam.

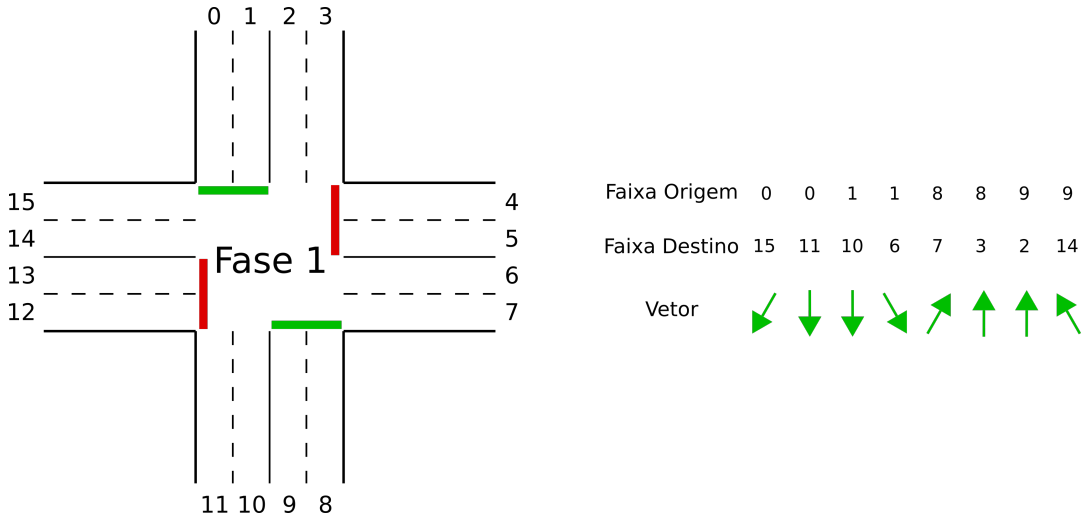



Figura 4.9 – Processo de transformação da ação da interseção (fase atual) em vetores de indicação de tráfego

## 4.3 Resumo

Este capítulo apresentou uma estrutura genérica com organização hierárquica para aprendizado por reforço e uma aplicação dessa estrutura para controle semafórico. Foram discutidos os detalhes e as interações entre os agentes da estrutura genérica: informações processadas; ação dos supervisores (assim como sua relação com as ações dos subordinados) e a influência nas recompensas dos agentes subordinados. Os pontos que diferenciam a estrutura genérica de outros conceitos da literatura que utilizam uma organização hierárquica foram apresentados. Em geral, o diferencial da organização hierárquica genérica proposta é a utilização de abstrações tanto para construir os estados dos agentes supervisores quanto na recomendação dos supervisores, sendo uma organização hierárquica

mais flexível por não haver controle dos agentes subordinados, mas sim, um influência no aprendizado deles. Respeitando-se as ideias principais da organização proposta, o projetista pode combinar diferente funções para realizar as abstrações de estado e relações entre as ações entre subordinados e supervisores, adaptando o modelo proposto para seu problema em específico.

Por fim, foi apresentada a aplicação para controle semafórico, explicando em maior detalhes como a estrutura genérica foi adaptada ao problema utilizando cálculos com vetores para realizar: as trocas de informações entre os agentes com os vetores resultantes; a recomendação do supervisor com o vetor de indicação de fluxo de tráfego; e o incentivo na recompensa dos subordinados com a função cosseno entre os vetores de ação do subordinado e de indicação do supervisor. A utilização de cálculos com vetores para realizar as abstrações e relações entre os agentes de diferentes níveis é o diferencial da aplicação proposta, permitindo aos agentes supervisores uma maneira simples de considerarem as informações de seus subordinados e de influenciarem nos aprendizados deles e, ao mesmo tempo, mantendo o potencial de escalabilidade do método.

# 5 EXPERIMENTOS

Nesse capítulo são apresentados os experimentos realizados a fim de avaliar a abordagem proposta apresentada no Capítulo 4. A Seção 5.1 apresentada o simulador utilizado. Na Seção 5.2, são apresentados os detalhes do cenário estudado. Na Seção 5.3, são detalhados os métodos que serão comparados nos experimentos, assim como os modelos dos agentes utilizados nesses métodos. Na Seção 5.4, é detalhado o algoritmo utilizado no método VHO. As Seções 5.5, 5.6 e 5.7 detalham, respectivamente, as configurações usadas nas simulações, as métricas usadas nas comparações dos métodos e os resultados encontrados nos diferentes experimentos realizados. Por fim, a Seção 5.8 resume o capítulo dos experimentos.

## 5.1 Simulador de Tráfego: SUMO

Como apresentado na Seção 2.7, nesta dissertação precisamos simular os veículos em alto nível de detalhes, logo utilizamos o simulador de tráfego microscópico SUMO - *Simulation of Urban Mobility* (LOPEZ et al., 2018).

O SUMO é um simulador *open-source* programado em C++. O simulador permite a modelagem de diferentes entidades de tráfego, entre elas: controladores semafóricos, veículos particulares, transporte público e pedestres. Ele pode receber diferentes configurações de simulação como arquivos de entrada e pode escrever arquivos de saída com diversas medidas, como tempo de viagem e de espera dos veículos, tempo de viagem das vias, emissão de poluentes, entre outras.

O SUMO pode utilizar a arquitetura TraCI - *Traffic control interface* (WEGENER et al., 2008) para se conectar, via *sockets*, a uma aplicação externa, permitindo a essa aplicação controle sobre a execução de uma simulação. Assim, é possível enviar comandos que alteram o estado da simulação e obter diferentes informações das entidades de tráfego simuladas. Nos experimentos, utilizamos a arquitetura TraCI para obter as informações necessárias para o aprendizado dos diferentes agentes simulados.

## 5.2 Cenário Estudado - Rede em grid 4x4

A abordagem proposta foi testada em uma rede em *grid* 4x4 com um total de 16 interseções. A rede em *grid* foi escolhida pelas seguintes razões: é um tipo de rede comumente encontrada na literatura de aprendizado por reforço em controle semafórico, o que facilita a reprodução dos experimentos e comparação de resultados para outros pesquisadores. Além disso, várias cidades são baseadas em um esquema de *grid* (Nova York, Washington, Miami) e, dessa maneira, os resultados apresentados podem ser relevantes em alguns cenários reais.

A rede em *grid* 4x4 utilizada possui vias de mão dupla internas de 200m e externas de 100m, conforme Fig. 5.1. Todas as faixas da rede possuem velocidade máxima de 50 km/h e todas as 16 interseções são controladas por semáforos. Cada interseção possui duas fases verdes: uma que libera o fluxo de veículos na direção Norte-Sul, permitindo conversões a esquerda; e outra que libera o fluxo na direção Leste-Oeste, também permitindo conversões a esquerda.

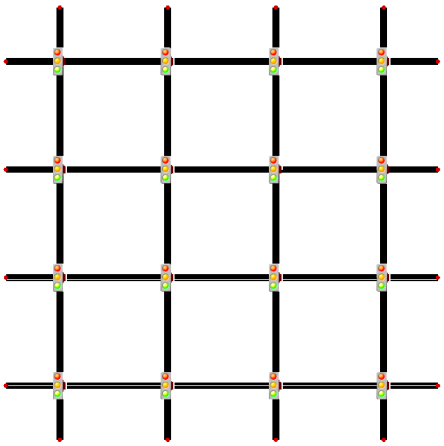


Figura 5.1 – Rede *grid* 4x4 utilizada nos experimentos

Nesse cenário, modelamos a demanda em 8 pares de origem-destino que representam as viagens realizadas pelos veículos na horizontal (A2F5, A3F4, A4F4, A5F2) e na vertical (B6E1, C6D1, D6C1, E6B1) conforme Fig. 5.2. Os pares ODs definem rotas cruzadas, pois o cenário se torna mais interessante do que apenas rotas diretas entre as extremidades da rede. Após testes com diferentes fluxos de tráfego utilizando esses pares ODs, definimos três diferentes experimentos que proporcionam cenários interessantes, i.e, com um nível significativo de congestionamento:

- O experimento padrão: no qual o volume de veículos nos sentidos Sul-Norte e

Oeste-Leste são iguais e um veículo é inserido a cada 7,2 segundos (500 veículos por hora) em cada um dos 8 pares ODs;

- O experimento NS: no qual o volume de veículos é maior no sentido Sul-Norte (direção Norte-Sul) do que no sentido Oeste-Leste. Nesse experimento, um veículo é inserido a cada 4,8 segundos (750 veículos por hora) em cada um dos 4 pares ODs verticais e um veículo é inserido a cada 14.4 (250 veículos por hora) segundos em cada um dos 4 pares ODs horizontais;

- O experimento LO: no qual o volume de veículos é maior no sentido Oeste-Leste (direção Leste-Oeste) do que no sentido Sul-Norte. Nesse experimento, um veículo é inserido a cada 4.8 segundos em cada um dos 4 pares ODs horizontais e um veículo é inserido a cada 14.4 segundos em cada um dos 4 pares ODs verticais.

Nas simulações, os veículos são inseridos na rede durante os 10000 segundos iniciais (aproximadamente 2,8 horas) e as simulações têm duração de 25000 segundos (aproximadamente 7 horas). Os arquivos de rotas foram gerados utilizando as ferramentas proporcionadas pelo simulador e seu processo é detalhado no Apêndice A.

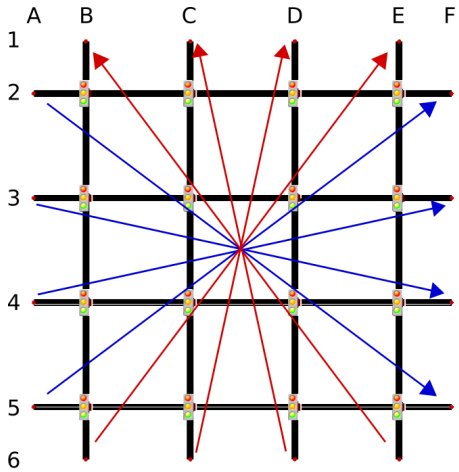



Figura 5.2 – Pares origem-destino da rede grid 4x4 utilizada nos experimentos

## 5.3 Métodos comparados nos experimentos

Os experimentos são realizados comparando três métodos para validar o melhor desempenho da organização hierárquica proposta para o controle semafórico. Comparamos um método com tempo fixo para os controladores semafóricos, um método onde os agentes dos controladores semafóricos utilizam aprendizado por reforço e um método que

utiliza o aprendizado por reforço com a organização hierárquica proposto na Seção 4.2, o VHO.

### 5.3.1 Método com tempo Fixo - Webster

Nos experimentos, o método de controle semafórico com tempo fixo utilizado é baseado na ferramenta proporcionada pelo SUMO, *tlsCycleAdaptation*. Essa ferramenta recebe como entrada um arquivo de definição de rede e um arquivo de definição de demanda, e calcula os tempos de ciclo e de fases para cada semáforo de acordo com o método Webster (WEBSTER, 1958) para melhor acomodar a determinada demanda.

Os tempos das fases verde dos controladores semafóricos para os diferentes experimentos podem ser encontrados na Tab. 5.1.

Tabela 5.1 – Tempos das fases dos semáforos nos experimentos

| **Fase** | **Experimento** | | |
|---|---|---|---|
| | **Padrão** | **NS** | **LO** |
| Fase Norte-Sul (s) | 8 | 12 | 4 |
| Fase Leste-Oeste (s) | 8 | 4 | 12 |

### 5.3.2 Método com aprendizado por reforço - Aprendizado

No método com aprendizado por reforço, os agentes interseção são modelados como MDPs e foram inspirados no modelo utilizado em um trabalho realizado dentro do MASLab (ALEGRE, 2019) - versão de Janeiro 2020. Assim, o modelo de agente interseção possui as seguintes características de estado, ação e recompensa para realizar seu aprendizado.

#### *5.3.2.1 Estado*

A definição do estado influencia diretamente no comportamento e na eficiência do agente. Cada agente interseção observa um vetor de informações que representam parcialmente o estado da interseção controlada. No presente trabalho, o estado de um agente interseção é definido como um vetor $s \in \mathbb{R}^{2|L|}$, onde $L$ é o conjunto de faixas da interseção. E, seja $o_l \in [0, 1]$ a porcentagem de ocupação de veículos da faixa $l$ controlada

pela interseção e $f_l \in [0, 1]$ o número de veículos parados (com velocidade menor que 0,1 m/s) na faixa $l$ controlada pela interseção divido pelo comprimento da faixa, o estado de um agente interseção é definido pela Eq. 5.1:

$$s = [o_l, f_l, ..., o_{|L|}, f_{|L|}] \tag{5.1}$$

A Fig. 5.3 ilustra um exemplo de um estado de um agente interseção em um determinado instante $t$. Na figura, a ocupação máxima de um faixa é de três veículos. Observando a faixa $0$, temos um veículo na faixa, representando uma ocupação de $33\%(o_0 = 0.33)$ da faixa e, como este veículo não está parado, uma fila $f_0 = 0$. Observando a faixa $4$, temos três veículos parados, caracterizando uma ocupação e um fila de de $100\%(o_4 = 1, f_4 = 1)$.

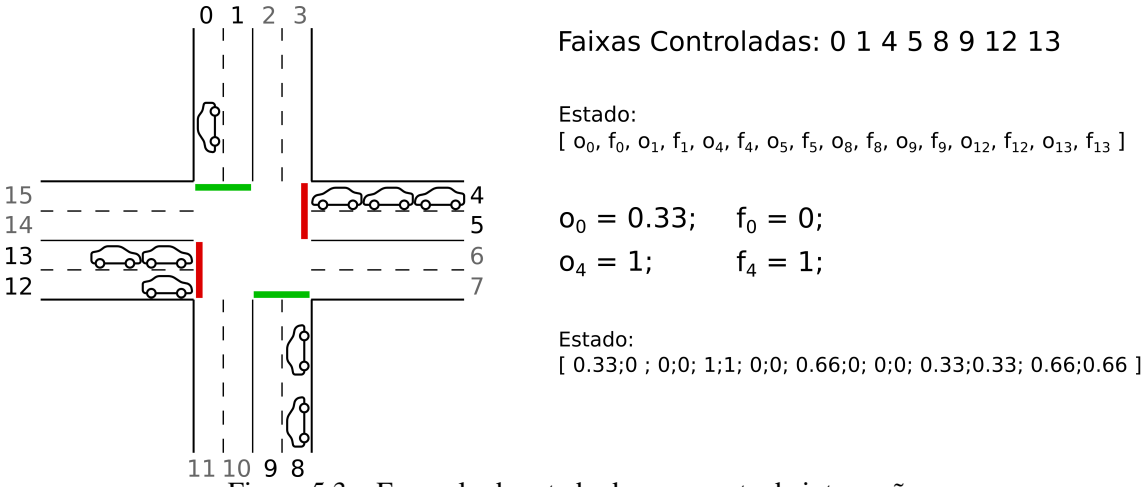



Figura 5.3 – Exemplo de estado de um agente de interseção

Como os componentes que descrevem os diferentes estados da intersecção listados acima são de característica contínua e o número de estados possíveis cresce exponencialmente com o número de faixas controladas pela interseção, os agentes interseção utilizam o algoritmo de aprendizado por reforço Sarsa (Seção 2.2) com aproximação de função (Seção 2.3). É importante notar que, para obtermos essas informações das interseções, múltiplos sensores devem ser instalados, sendo uma operação custosa e, talvez, não factível no mundo real.

#### *5.3.2.2 Ação*

Em um MDP, o agente escolhe uma ação $a \in A$, seu conjunto de ações, a cada $t$ segundos decorridos. Neste modelo, o número de ações possíveis é igual ao número de fases verdes possíveis na interseção (duas fases, conforme Seção 5.2) e os agentes

tomam sua ação a cada 5 segundos (tempo comumente encontrado na literatura de controle semafórico). Nessa perspectiva, o agente interseção, escolhe a próxima fase verde a ser executada. Caso escolha a fase atual, ela é estendida, recebendo mais tempo de verde. Caso escolha outra fase possível, ocorre o procedimento de troca de fases, que executa uma fase amarela de duração fixa de três segundos antes de trocar para a próxima fase verde escolhida. Há duas restrições na seleção da ação: a fase atual só pode ser trocada caso o tempo decorrido da fase atual seja maior que o tempo mínimo de verde (nos experimentos, 10 segundos); e, caso uma fase atinja o tempo máximo de vermelho (nos experimentos, o tempo do ciclo do semáforo), ela deve ser ativada, receber tempo de verde. As restrições e o procedimento de troca de fases são utilizados para melhor modelar o comportamento de um controlador semafórico na vida real.

#### *5.3.2.3 Recompensa*

As recompensas dos agentes interseção são definidas como a diferença na soma dos tempos de espera dos veículos entre ações sucessivas. Seja $V_s$ o conjunto de veículos nas faixas controladas pela interseção em um determinado estado $s$ e $w_{v,s}$ o tempo de espera do veículo $v$ em um determinado estado $s$, $W_s$ é a soma do tempo de espera dos veículos nas faixas controladas pela interseção dada pela Eq. 5.2:

$$W_s = \sum_{v \in V_s} w_{v,s} \tag{5.2}$$

Após executar uma ação $a$ em um estado $s$ e chegar em um estado $s'$, o agente recebe a recompensa $r$ definida pela Eq. 5.3:

$$r = W_s - W_{s'} \tag{5.3}$$

Assim, quanto maior a redução no tempo de espera dos veículos, maior a recompensa, incentivando os agentes a reduzirem esse tempo nas intersecções, melhorando o fluxo de tráfego. No SUMO, um veículo é considerado esperando quando se encontra parado, i.e, com velocidade menor que 0.1 m/s.

A Fig. 5.4 ilustra um exemplo de recompensa de um agente interseção. No estado $s$, temos quatro veículos esperando, com um tempo total de espera $W_s = 35$ segundos e, no estado $s'$, temos 3 veículos esperando, cum um tempo total de espera $W_{s'} = 10$ segundos. Assim a recompensa, a diferença entre os tempos de espera, no exemplo é igual a 25.

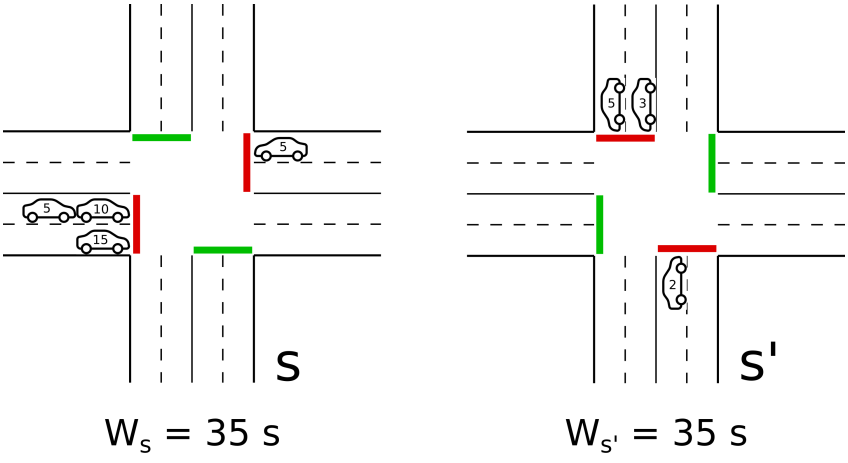



Figura 5.4 – Exemplo de recompensa de um agente de interseção

### 5.3.3 Método com a organização hierárquica proposta - VHO

No método que utiliza a organização hierárquica proposta, como explicado na Seção 4.2, temos dois tipos de agentes: os agentes interseção e os agentes região.

Os agentes interseção são modelados como apresentado na Seção 5.3.2, com o ajuste da recomendação do agente supervisor ser acoplada na sua definição de estado. Assim, sendo $\mathcal{A}^{+i}$ a recomendação do agente supervisor do agente $i$, o estado de um agente interseção $i$ é definido pela Eq. 5.4:

$$s_i = [o_l, f_l, ..., o_{|L|}, f_{|L|}, \mathcal{A}^{+i}] \tag{5.4}$$

Os agentes região são modelos utilizando os conceitos apresentados na Seção 4.2. Assim, sendo $a$ um agente região; $\vec{e}_{R,i}$ e $\vec{s}_{R,i}$ os vetores resultantes de entrada e saída dos agentes $i \in S^{-a}$ subordinados ao agente $a$, definidos pelas equações 4.6 e 4.7; e $\mathcal{A}^{+a}$ a recomendação do agente supervisor do agente $a$, caso ele seja subordinado; o estado do agente $a$ é definido pela Eq. 5.5:

$$s_a = \begin{cases} [\vec{e}_{R,0}; \vec{s}_{R,0}; \cdots; \vec{e}_{R,|S^{-a}|}; \vec{s}_{R,|S^{-a}|}] \\ [\vec{e}_{R,0}; \vec{s}_{R,0}; \cdots; \vec{e}_{R,|S^{-a}|}; \vec{s}_{R,|S^{-a}|}; \mathcal{A}^{+a}] \quad \text{quando } a \text{ é subordinado} \end{cases} \tag{5.5}$$

As ações dos agentes região são as recomendações dos supervisores descritas na Seção 4.2.2 e a suas recompensas são as médias aritméticas simples das recompensas dos

seus subordinados, i.e, no VHO, $\mathfrak{F}$ é a media aritmética simples. É utilizada uma média aritmética simples para construir a recompensa do agente região para que ele leve em consideração as recompensas de todos são subordinados quando for realizar seu aprendizado.

Novamente, como os componentes que descrevem os diferentes estados dos agentes de interseção e de região são de característica contínua e o número de estados possíveis cresce exponencialmente com o número de faixas controladas pela interseção e número de agentes subordinados da região, os agentes utilizam o algoritmo de aprendizado por reforço Sarsa (Seção 2.2) com aproximação de função (Seção 2.3).

No método com organização hierárquica, tantos os agentes interseção quanto os agentes região, quando são subordinados, recebem os incentivos de seus supervisores conforme descrito na Seção 4.2.3. As divisão de regiões do cenário estudado, a rede em grid 4x4, é ilustrada na Fig. 5.5.

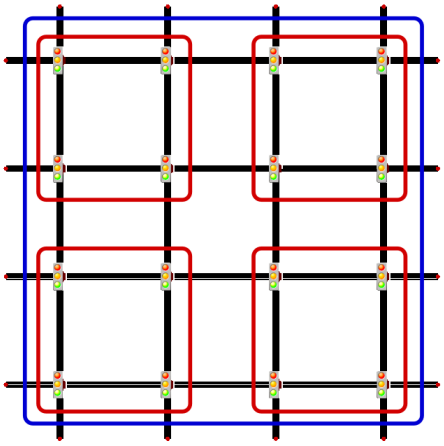


Figura 5.5 – Divisão das regiões do cenário em rede grid 4x4 - 4 regiões de nível hierárquico 1 (vermelho), com 4 interseções subordinadas cada; 1 região de nível hierárquico 2 (azul) cobrindo toda a rede, com as 4 regiões de nível hierárquico 1 como subordinados

## 5.4 Algoritmo do VHO

O Algoritmo 1 utiliza os conceitos apresentados na Seção 4.2 e os modelos apresentados nas Seções 5.3.2 e 5.3.3 para descrever o comportamento da organização hierárquica proposta em uma simulação. É necessário, para o método, uma descrição da ordem hierárquica dos agentes região e seus respectivos subordinados, conforme Apêndice B. Nas linhas 1 - 6 do algoritmo, temos a inicialização dos parâmetros dos agentes, os vetores de pesos são inicializados com zero e os valores de $\alpha$, $\gamma$ e $\varepsilon$ dos diferentes agentes são inicializados com os valores de entrada. Os agentes coletam as informações iniciais de

estado e ação conforme algoritmo 2, percorrendo a ordem hierárquica de forma crescente para coletar as informações processadas dos subordinados e percorrendo a ordem hierárquica de forma decrescente para acoplar a recomendação dos supervisores nos estados dos subordinados. Enquanto a simulação não termina, os passos de simulação são avançados de 5 em 5 segundos, linha 11. A cada passo de simulação, são observadas as recompensas dos agentes e são calculados os próximos estados e as próximas ações a serem tomadas para que os agentes possam aprender, linha 15 . As recompensas são calculadas seguindo a ordem hierárquica de forma crescente e os incentivos são calculados seguindo a ordem decrescente, conforme algoritmo 3. Ao final de cada passo de simulação, métricas são coletadas sobre as informações da rede de transporte e dos agentes, linha 18.

**Algoritmo 1:** Simulação utilizando o VHO

**Dados:** Ordem Hierárquica e subordinados, $\alpha_I$, $\gamma_I$, $\varepsilon_I$, $\alpha_R$, $\gamma_R$, $\varepsilon_R$
**Resultado:** Conjunto de métricas sobre a simulação realizada
// Inicialização
1 **para cada** *agente interseção* $i \in I$ **faça**
2  $i$.weights = [$0 \cdots 0$];
3  $i.\alpha = \alpha_I$ ; $i.\gamma = \gamma_I$ ; $i.\epsilon = \varepsilon_I$;
4 **para cada** *agente região* $z \in R$ **faça**
5  $z$.weights = [$0 \cdots 0$];
6  $z.\alpha = \alpha_R$ ; $z.\gamma = \gamma_R$ ; $z.\epsilon = \varepsilon_R$;
// Informações Iniciais
7 Coleta_Informacoes ("Estado", "Acao");
8 tempo_da_simulação = 0;
9 métricas[tempo_da_simulação] = valores iniciais das métricas;
// Loop
10 **enquanto** *não terminou a simulação* **faça**
11  Avança a simulação em 5 segundos, atualizando tempo_da_simulação;
 // Recompensas
12  Calcula recompensas();
 // Próximas informações
13  Coleta_Informacoes ("Proximo_Estado", "Proxima_Acao");
 // Aprendizado
14  **para todo** *agente* $a \in A$ **faça**
15  $a$.aprende(estado, ação, recompensa, proximo_estado, proxima_ação) usando Sarsa com aproximação de função;
16  $a$["Estado"] = $a$["Proximo_Estado"];
17  $a$["Acao"] = $a$["Proxima_Acao"];
 // Métricas
18  métrica[tempo_da_simulação] = valores das métricas observadas;

**Algoritmo 2:** Coleta_Informacoes()

**Dados:** chaveDeEstado, chaveDeAcao
**Resultado:** Estados e ações dos agentes

**para cada** *agente interseção* $i \in I$ **faça**
    $i$[chaveDeEstado] = estado a partir de suas observações;
**para cada** *agente região* $z \in R$ *seguindo a ordem hierárquica de forma crescente* **faça**
    state = [];
    **para cada** *agente subordinado* $s \in S^{-z}$ **faça**
        state.append(s.vetoresResultantes()) conforme Eq. 5.5;
    $z$[chaveDeEstado] = state;
**para cada** *agente região* $z \in R$ *seguindo a ordem hierárquica de forma decrescente* **faça**
    z[chaveDeAcao] = z.act();
    **para cada** *agente subordinado* $s \in S^{-z}$ **faça**
        $s$[chaveDeEstado].append(z.ação) conforme Eq. 5.4;
**para cada** *agente interseção* $i \in I$ **faça**
    $i$[chaveDeAcao] = $i$.act();

**Algoritmo 3:** Calcula_Recompensas()

**Resultado:** Recompensas dos agentes

**para cada** *agente interseção* $i \in I$ **faça**
    $i$.recompensa = recompensa segundo Eq. 5.3;
**para cada** *agente região* $z \in R$ *seguindo a ordem hierárquica de forma crescente* **faça**
    $z$.recompensa = média aritmética simples das recompensas dos seus subordinados $s \in S^{-z}$;
**para cada** *agente região* $z \in R$ *seguindo a ordem hierárquica de forma decrescente* **faça**
    **para cada** *agente subordinado* $s \in S^{-z}$ **faça**
        $s$.recompensa = recompensa com incentivo conforme Seção 4.2.3;

## 5.5 Configurações das simulações

Comparamos os três métodos descritos na Seção 5.3 no cenário e experimentos descritos na Seção 5.2. Nos modelos que utilizam aprendizado, os agentes de interseção e de região escolhem suas ações usando uma estratégia de exploração $\varepsilon$-*greedy*. A taxas de exploração e o fator de desconto são mantidas constantes e seus valores são $\varepsilon_I, \varepsilon_R = 0,05$ e $\gamma_I, \gamma_R = 0,95$, valores normalmente utilizados na literatura de RL. Por simplicidade, em ambos agentes interseção e agentes região, a aproximação de função é feita diretamente sobre as observações parciais dos agentes, i.e, diretamente sobre suas variáveis de estado, não havendo funções de construções de *features*.

Nos experimentos com o método com aprendizado por reforço (Aprendizado), a taxa de aprendizagem $\alpha_I = 0,0001$, para os agentes interseção, foi a que obteve melhores resultados após extensa experimentação. Nos experimentos com o método de aprendizado por reforço utilizando a organização hierárquica proposta (VHO), as taxas de aprendizado $\alpha_I = 1 \times 10^{-5}$ para os agentes interseção e $\alpha_R = 2 \times 10^{-6}$ para os agentes região foram as que obtiveram melhores resultados após extensa experimentação.

Nos experimentos com métodos que utilizam aprendizado por reforço (Aprendizado e VHO) foram realizadas 10 simulações para cada método usando os parâmetros acima. São necessárias múltiplas simulações para esses métodos pois ao utilizá-los temos os seguintes fatores não determinísticos que podem alterar os seus desempenhos: a exploração aleatória que os agentes realizarão durante o seu aprendizado; o fato do próprio aprendizado não ser determinístico por estarmos em um sistema multiagente; e o comportamento do tráfego ser diferente conforme o aprendizado realizado em cada simulação.

Após cada simulação, são gerados arquivos *.csv* que guardam as informações coletadas a cada passo de simulação, conforme exemplo no Apêndice C, e arquivos de *log* do simulador, os quais geram informações importantes da simulação como um todo. Ambos os arquivos *.csv* e *log* são usados para realizar as comparações feitas neste capítulo e testes estatísticos de ANOVA (KAUFMANN; SCHERING, 2014) e Tukey (HAYNES, 2013) são realizados sobre as métricas encontradas para comprovar suas diferenças estatísticas, conforme valores apresentados no Apêndice D.

## 5.6 Métricas

Para cada experimento realizado serão apresentadas tabelas onde os três métodos mencionados serão comparados usando as seguintes métricas:

- Recompensa: recompensa média recebida pelos agentes interseção nas simulações, calculada conforme Eq. 5.3;
- Fluxo de viagens concluídas: quantidade de veículos por segundo chegando ao seu destino e concluindo suas viagens na simulação.
- Tempo para suprir demanda: tempo no qual o último veículo completou sua viagem;
- Tempo médio de espera durante a viagem: tempo no qual os veículos estavam com velocidade igual ou menor a 0.1m/s;
- Tempo médio perdido durante a viagem: tempo perdido devido aos veículos se deslocarem abaixo da velocidade ideal (estimada pelo modelo de veículo do simulador);
- Atraso médio de partida: tempo que os veículos têm que esperar antes de começarem suas viagens devido ao congestionamento na rede;

## 5.7 Resultados encontrados

Os resultados apresentados a seguir foram coletados de simulações usando as configurações da Seção 5.5 e, além das métricas mencionadas na Seção 5.6, são apresentados gráficos para cada experimento realizado. Nos resultados a seguir, serão apresentadas as médias e os desvios padrão: nas tabelas de comparação, no formato *valor* $\pm$ *desvio padrão* e, nos gráficos, na forma de *linhas e sombras*. Como o método que utiliza tempo fixo (Webster) não possui variação, são apenas apresentadas suas médias.

### 5.7.1 Experimento padrão

Como descrito na Seção 5.2, no experimento padrão temos um volume de demanda igual em ambos os sentidos Sul-Norte e Oeste-Leste. A Tab. 5.2 e as Fig. 5.6 - 5.8, mostram os resultados encontrados neste experimento.

Tabela 5.2 – Desempenho dos métodos no experimento padrão com diferentes métricas de avaliação - Valor ± desvio padrão

| **Métrica** | **Método** | | |
|---|---|---|---|
| | **Webster** | **Aprendizado** | **VHO** |
| Recompensa ⇑ | -3644 | -1901 ± 141 | -1384 ± 212 |
| Fluxo de viagens concluídas (vei/s) ⇑ | 0,51 | 0,69 ± 0,03 | 0,74 ± 0,06 |
| Tempo para suprir demanda (s) ⇓ | 21792 | 16053 ± 714 | 14926 ± 1321 |
| Tempo médio de espera durante a viagem (s) ⇓ | 530 | 435 ± 41 | 393 ± 79 |
| Tempo médio perdido durante viagem (s) ⇓ | 660 | 543 ± 48 | 503 ± 87 |
| Atraso médio de partida (s) ⇓ | 2818 | 1351 ± 262 | 879 ± 304 |

Como podemos observar na Tab. 5.2, o método VHO tem melhor desempenho visto que leva um tempo menor para suprir a demanda de veículos, tem um menor tempo de espera médio durante e um menor tempo médio perdido durante as viagens, tem um menor tempo de menor de inserção de veículos na simulação e um maior fluxo de viagens concluídas quando comparado aos outros métodos.

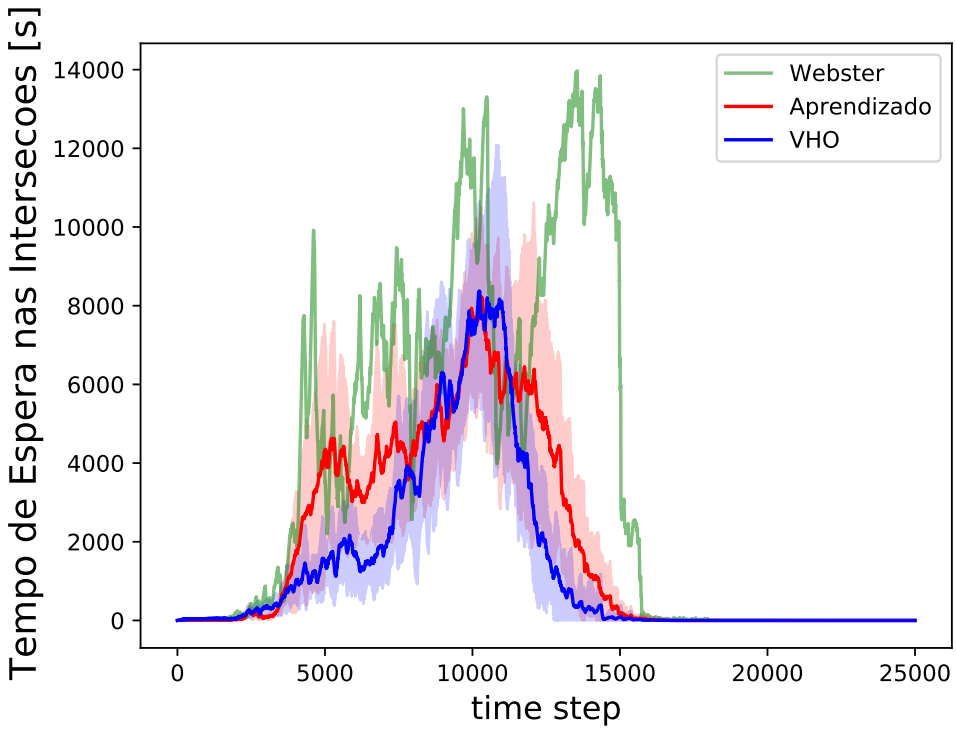



Figura 5.6 – Experimento padrão - Tempo de espera médio nas interseções

Como podemos observar nos gráficos, o VHO possui melhor desempenho. No gráfico do tempo de espera médio nas interseções (Fig. 5.6), observamos que durante e depois da inserção da demanda (10k step), os tempos de esperas usando o VHO são menores que os tempos dos outros métodos. Observamos, no gráfico dos veículos na simulação (Fig. 5.7), que o VHO apresenta um declínio antes do outros métodos, i.e, o

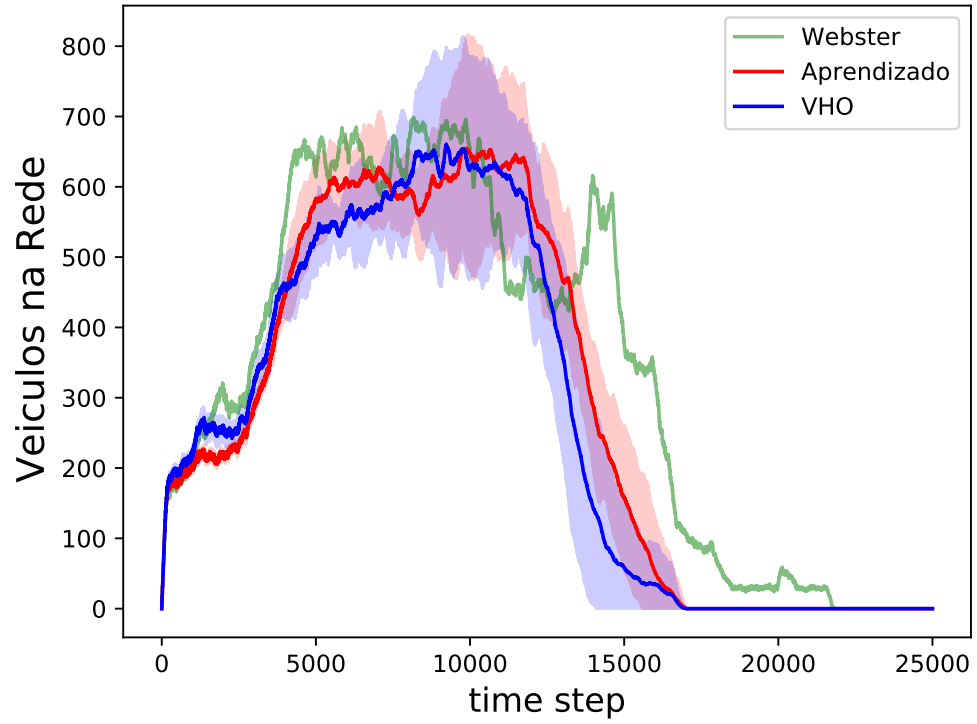



Figura 5.7 – Experimento padrão - Número de veículos na rede durante a simulação

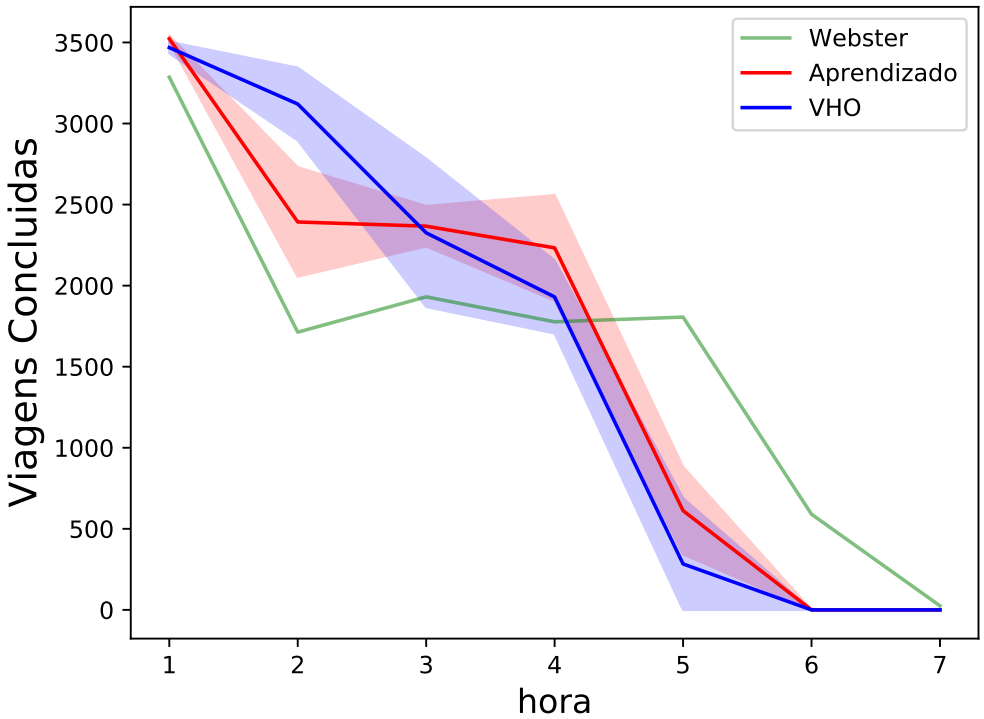



Figura 5.8 – Experimento padrão - Número de viagens concluídas por hora

número de veículos na rede diminui antes dos outros métodos, provando que o VHO supre a demanda mais rápido. No gráfico das viagens concluídas por hora (Fig. 5.8), podemos notar que o VHO começa completando mais viagens que os outros métodos nas primeiras horas e, em algumas simulações, supre a demanda na quinta hora.

### 5.7.2 Experimentos NS e LO

Como descrito na Seção 5.2, os experimentos NS e LO possuem um volume de maior de veículos na direção que identifica o experimento. Como os veículos entram em tempos defasados, um a cada 4.8 segundos nos pares ODs verticais/horizontais e um a cada 14.4 nos pares ODs horizontais/verticais, há menos congestionamentos nestes experimentos, resultando em valores de métricas menores do que os valores encontrados no experimento padrão. A Tab. 5.3 e as Fig. 5.9 - 5.11 mostram os resultados encontrados no experimento NS e a Tab. 5.4 e as Fig. 5.12 - 5.14 mostram os resultados encontrados no experimento LO.

Tabela 5.3 – Desempenho dos métodos no experimento NS com diferentes métricas de avaliação - Valor ± desvio padrão

| **Métrica** | **Método** | | |
|---|---|---|---|
| | **Webster** | **Aprendizado** | **VHO** |
| Recompensa ⇑ | -411 | -183 ± 97 | -134 ± 42 |
| Fluxo de viagens concluídas (vei/s) ⇑ | 0,77 | 0,9 ± 0,05 | 0,94 ± 0,03 |
| Tempo para suprir demanda (s) ⇓ | 14313 | 12374 ± 733 | 11842 ± 397 |
| Tempo médio de espera durante a viagem (s) ⇓ | 210 | 131 ± 16 | 129 ± 21 |
| Tempo médio perdido durante viagem (s) ⇓ | 326 | 203 ± 16 | 205 ± 27 |
| Atraso médio de partida (s) ⇓ | 1031 | 320 ± 150 | 239 ± 83 |

Tabela 5.4 – Desempenho dos métodos no experimento LO com diferentes métricas de avaliação - Valor ± desvio padrão

| **Métrica** | **Método** | | |
|---|---|---|---|
| | **Webster** | **Aprendizado** | **VHO** |
| Recompensa ⇑ | -979 | -651 ± 73 | -546 ± 31 |
| Fluxo de viagens concluídas (vei/s) ⇑ | 0,61 | 0,72 ± 0,03 | 0,75 ± 0,01 |
| Tempo para suprir demanda (s) ⇓ | 17975 | 15379 ± 692 | 14709 ± 378 |
| Tempo médio de espera durante a viagem (s) ⇓ | 414 | 301 ± 27 | 276 ± 18 |
| Tempo médio perdido durante viagem (s) ⇓ | 584 | 415 ± 32 | 388 ± 23 |
| Atraso médio de partida (s) ⇓ | 2556 | 1262 ± 243 | 1115 ± 93 |

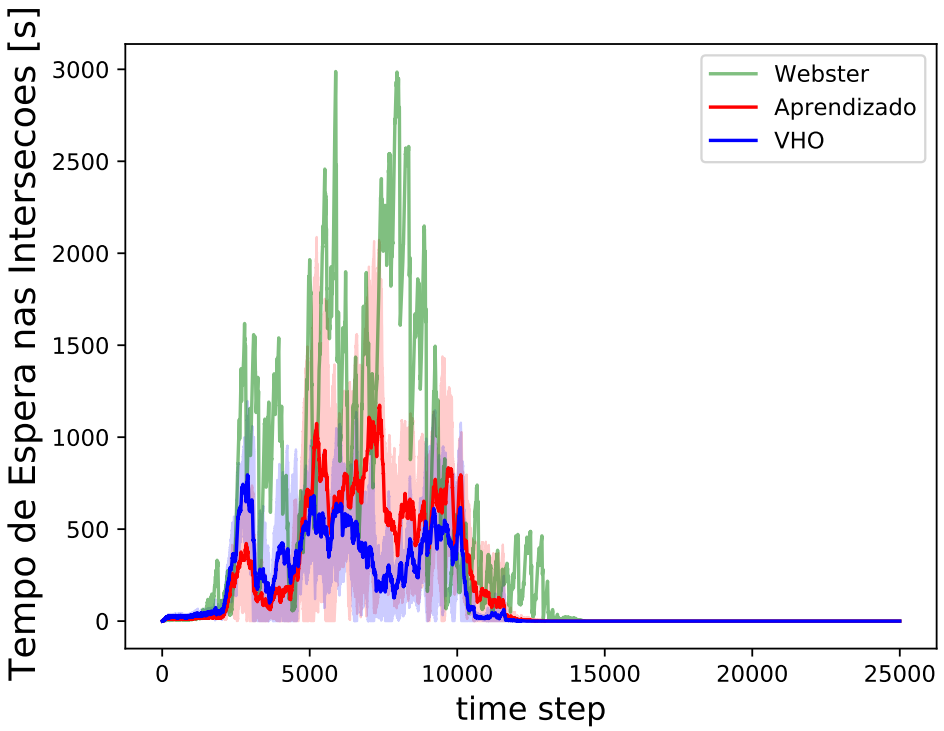



Figura 5.9 – Experimento NS - Tempo de espera médio nas interseções

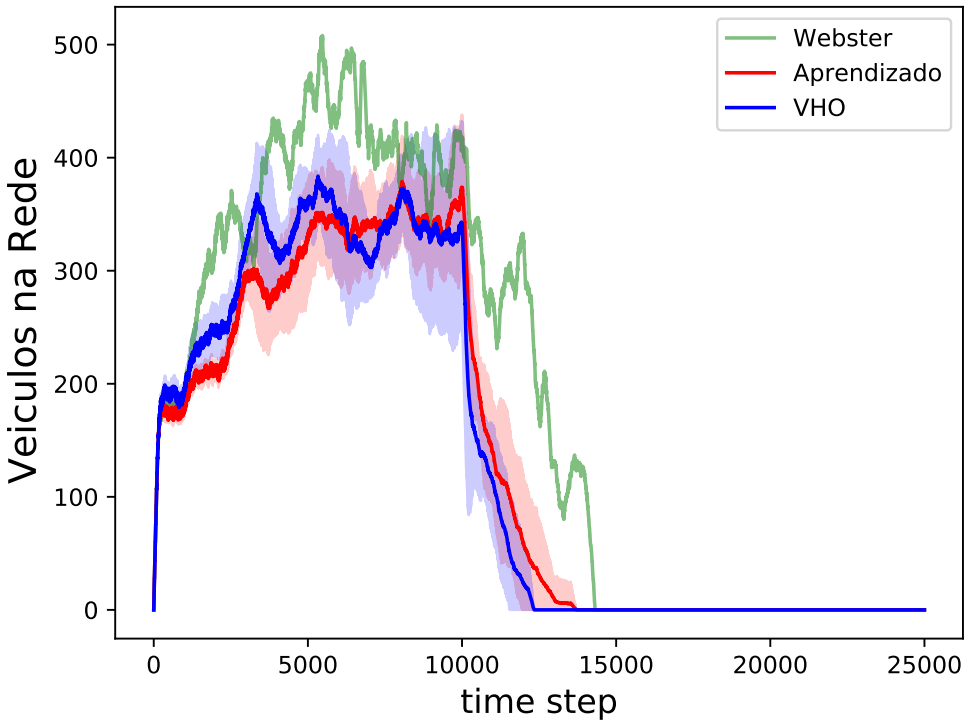



Figura 5.10 – Experimento NS - Número de veículos na rede durante a simulação

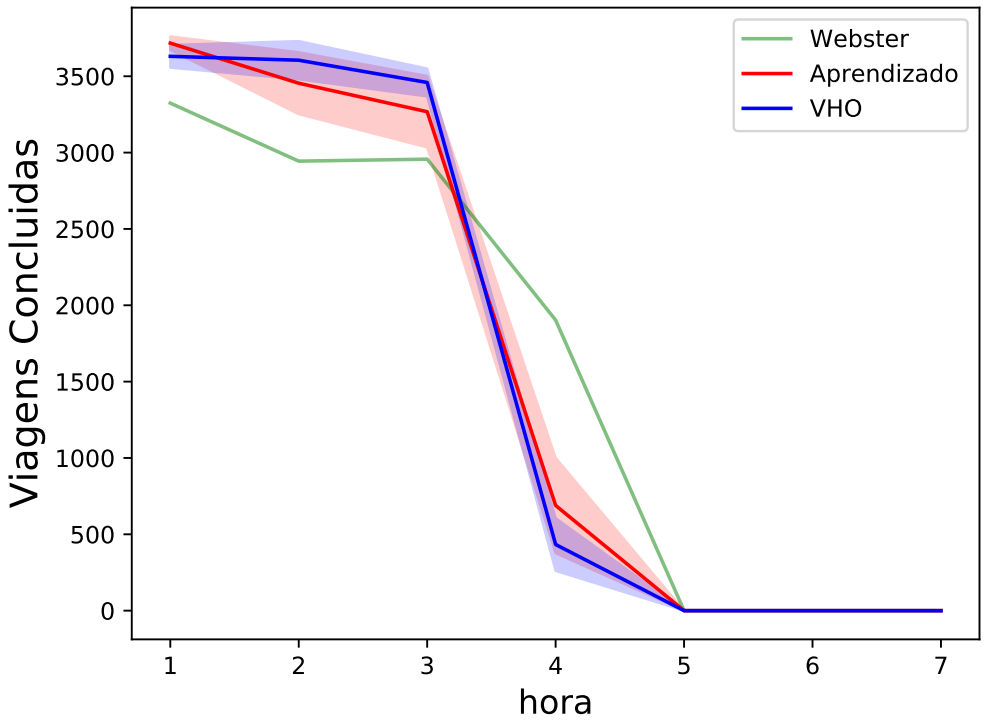



Figura 5.11 – Experimento NS - Número de viagens concluídas por hora

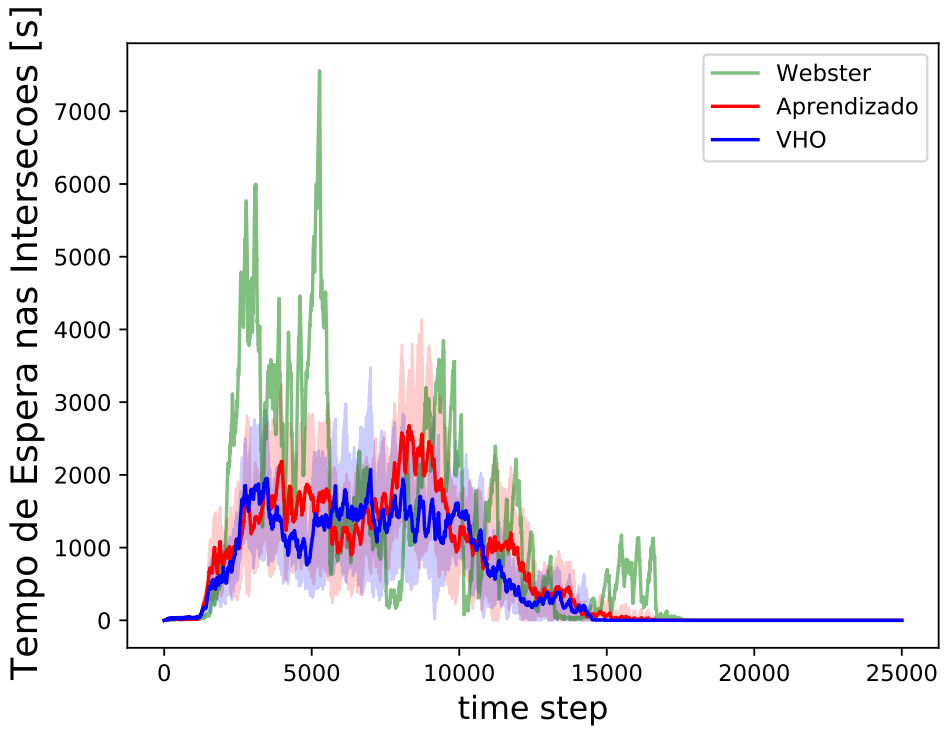



Figura 5.12 – Experimento LO - Tempo de espera médio nas interseções

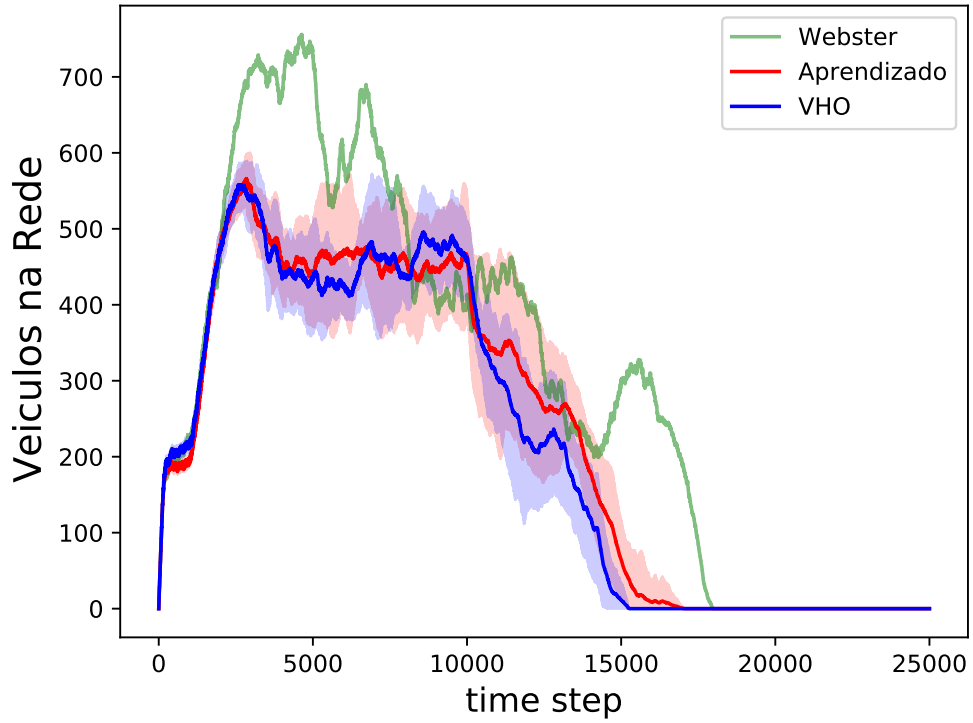



Figura 5.13 – Experimento LO - Número de veículos na rede durante a simulação

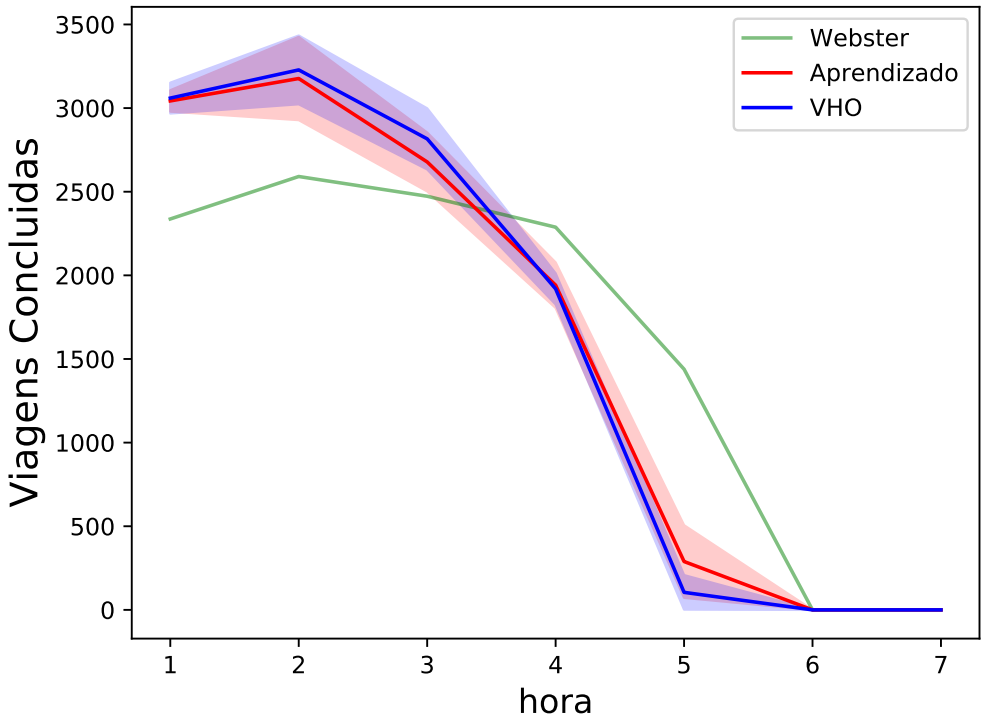



Figura 5.14 – Experimento LO - Número de viagens concluídas por hora

Observando as tabelas e os gráficos, podemos notar que utilizar a organização proposta possui um melhor desempenho que os outros métodos comparados nos experimentos NS e LO. Contudo, como esses experimentos apresentam um cenário mais simples, com menos congestionamento na rede, a diferença entre os desempenhos dos métodos diminuiu quando comparada com a diferença encontrada no experimento padrão.

### 5.7.3 Experimento de qualidade de aprendizado

O experimento de qualidade de aprendizado tem como objetivo mostrar que o aprendizado nos diferentes níveis hierárquicos, i.e, nos agentes região, é significativo para a metodologia proposta. Chamamos de *recomendação trivial*, a possível recomendação do agente região quando não realiza aprendizado e indica apenas a soma dos vetores que compõem seus estado, i.e, a soma dos vetores dos seus subordinados. Os vetores dos subordinados são somados e depois o vetor resultante é aproximado a uma das recomendações dos supervisores possíveis como ilustrado pelo exemplo da Fig. 5.15. Utilizando essa aproximação, constatamos que no experimento padrão (Seção 5.7.1), os agentes região, realizando aprendizado, indicavam a *recomendação trivial* 52% das vezes, logo surgiu o questionamento se o aprendizado nos níveis hierárquicos seria significativo para a metodologia. Para resolver essa questão, o experimento padrão foi executado com os agentes região não realizando aprendizado, apenas indicando a *recomendação trivial*, chamado de *VHO-*. A Tab. 5.5 e os gráficos 5.16 - 5.18 apresentam os resultados encontrados neste experimento e os compara aos resultados do experimento padrão.

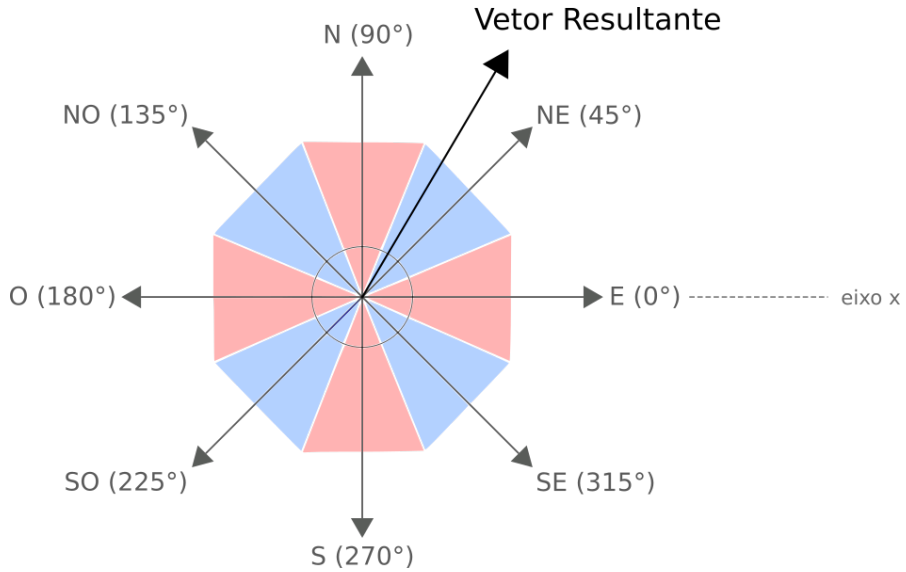



Figura 5.15 – Exemplo de aproximação de um vetor resultante para uma indicação NE

Tabela 5.5 – Desempenho do método sem aprendizado nos agentes região (VHO-) comparado ao método VHO - Valor ± desvio padrão

| **Métrica** | **Método** | |
|---|---|---|
| | **VHO-** | **VHO** |
| Recompensa ⇑ | -2377 ± 247 | -1348 ± 212 |
| Fluxo de viagens concluídas (vei/s) ⇑ | 0,68 ± 0,04 | 0,74 ± 0,06 |
| Tempo para suprir demanda (s) ⇓ | 16418 ± 1021 | 14926 ± 1321 |
| Tempo médio de espera durante a viagem (s) ⇓ | 492 ± 45 | 393 ± 79 |
| Tempo médio perdido durante viagem (s) ⇓ | 605 ± 49 | 503 ± 87 |
| Atraso médio de partida (s) ⇓ | 1555 ± 313 | 879 ± 304 |

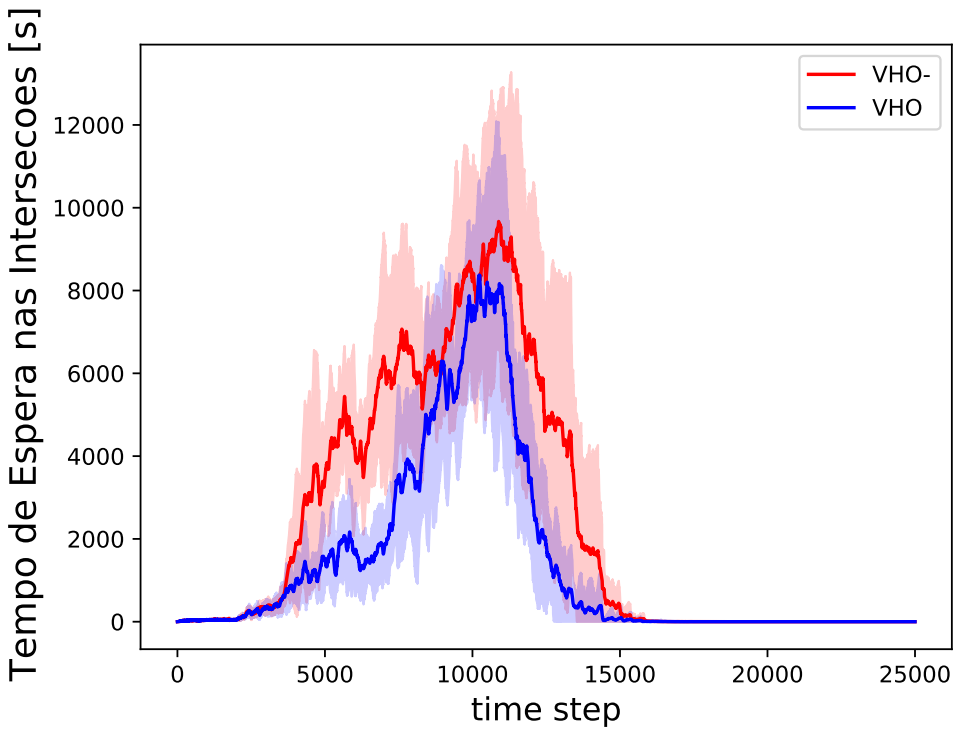



Figura 5.16 – Experimento de Qualidade Aprendizado - Tempo de espera médio nas interseções

Observando os resultados, comprovamos que o aprendizado nos agentes região é importante para metodologia proposta visto que o desempenho diminui quando esses agentes não estão aprendendo. Quando comparado aos outros métodos testados, utilizar a organização proposta com os agentes hierárquicos não realizando o aprendizado apresenta resultados melhores que o método de tempo fixo e resultados levemente piores que o método com aprendizado sem organização hierárquica.

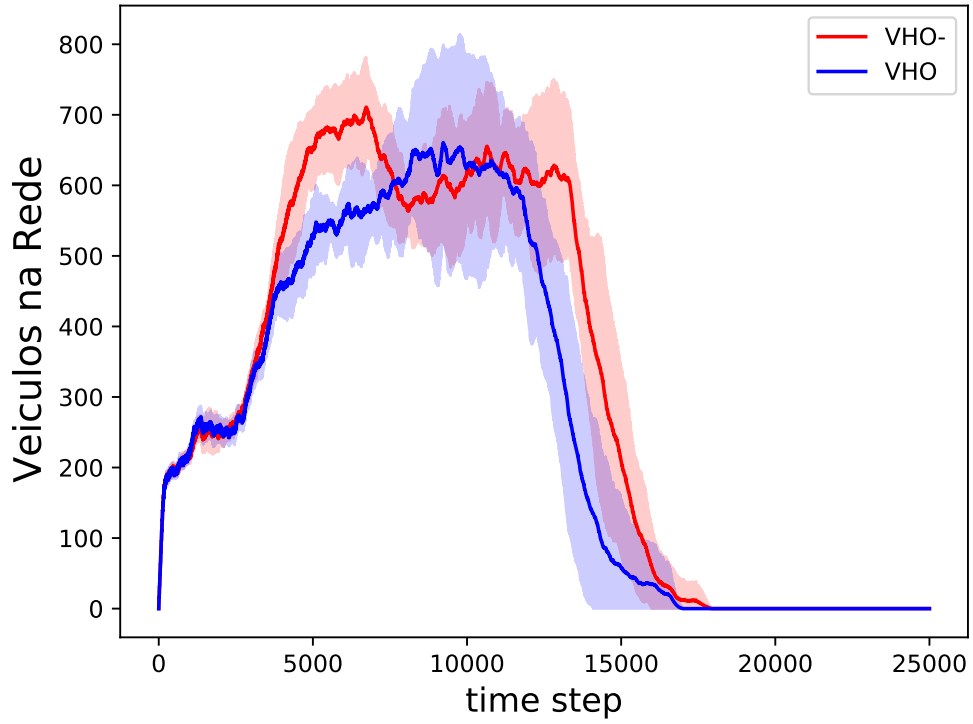



Figura 5.17 – Experimento de Qualidade Aprendizado - Número de veículos na rede durante a simulação

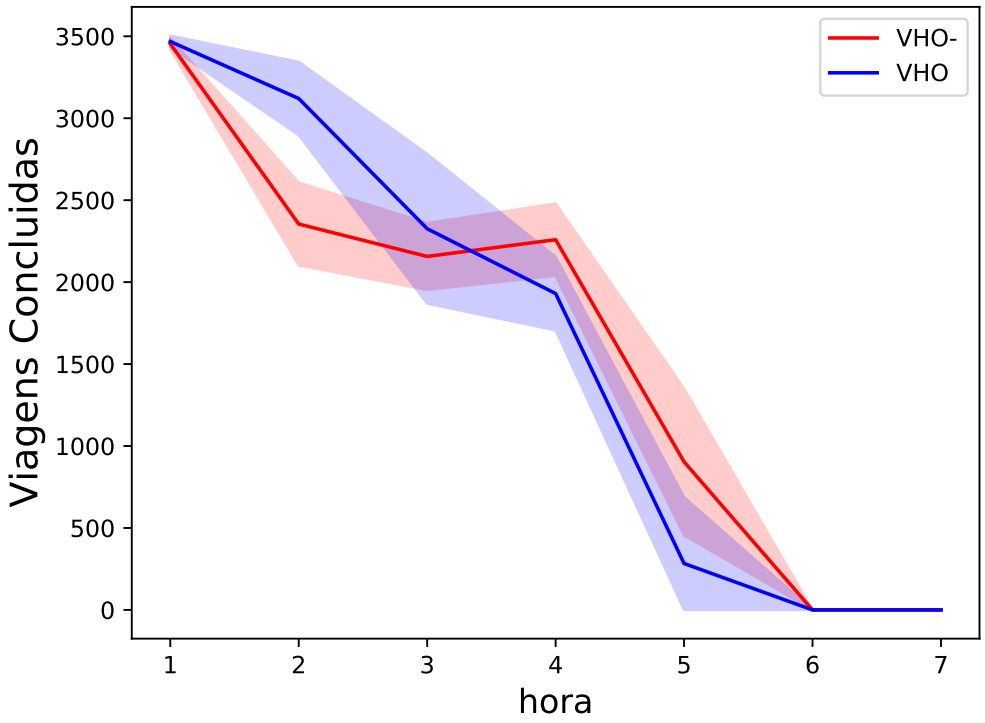



Figura 5.18 – Experimento de Qualidade Aprendizado - Número de viagens concluídas por hora

## 5.8 Resumo dos experimentos

Neste capítulo foram apresentados os detalhes dos experimentos realizados para avaliar a metodologia proposta no Capítulo 4. Inicialmente, o simulador SUMO é apresentado, o qual nos permite simular o tráfego de maneira microscópica, permitindo o aprendizado dos diferentes agentes. Apresentamos o cenário estudado, constituído de uma rede em *grid* 4x4 e três diferentes definições de demanda: uma com volumes iguais em ambas direções Norte-Sul e Leste-Oeste, uma com maior volume na direção Norte-Sul e outra com maior volume na direção Leste-Oeste.

Foram detalhados os métodos comparados nos diferentes experimentos: um método com controladores semafóricos com tempo fixo, um método com aprendizado por reforço sem organização hierárquica e o método que reflete a organização hierárquica proposta. Os três métodos foram comparados em três tipos de experimentos, um para cada definição de demanda, e o método com a metodologia proposta mostrou um melhor desempenho que os outros métodos em todos os experimentos. Por fim, foi realizado um experimento comprovando que a utilização de aprendizado por reforço nos níveis hierárquicos é necessária e significativa para o desempenho da metodologia desenvolvida nesta dissertação.

Em geral, o método para controle semafórico proposto, o VHO, apresentou resultados positivos nos experimentos realizados, comprovando que a utilização de uma organização hierárquica em conjunto com aprendizado com reforço pode melhorar o desempenho dos agentes em determinados problemas.

# 6 CONSIDERAÇÕES FINAIS

Este capítulo apresenta uma revisão das questões discutidas nos demais capítulos desta dissertação a fim de oferecer uma visão geral dos tópicos abordados. Além disso, as contribuições e possíveis trabalhos futuros são listados e discutidos.

## 6.1 Visão Geral

Nessa dissertação, apresentamos uma técnica de aprendizado por reforço com organização hierárquica para controle semafórico. Contextualizamos, no Capítulo 1, que o problema de congestionamentos nas áreas urbanas vem aumentando conforme o crescimento dessas áreas, causando problemas aos cidadãos como o aumento da poluição e possíveis acidentes. O controle semafórico se apresenta como uma solução prática para atenuar o congestionamento, uma vez que a expansão da infraestrutura das redes de transporte é cada vez mais complicada por questões econômicas, sociais e ambientais, e o uso eficiente da infraestrutura existente se torna necessário. Técnicas de aprendizado por reforço têm apresentado resultados positivos na otimização de controladores semafóricos. Contudo, poucos trabalhos na área utilizam uma organização hierárquica para auxiliar no aprendizado dos controladores. Nessa dissertação, apresentamos uma organização hierárquica para aprendizado por reforço, inspirada em ideias da área para obter um melhor desempenho no controle semafórico.

No Capítulo 4, foi apresentado o modelo genérico proposto para organizar o aprendizado de agentes de forma hierárquica. Nessa organização, os agentes região, de níveis hierárquicos diversos, levam em consideração as informações de seus subordinados, agentes de níveis hierárquicos mais baixos, para ter uma visão coletiva do problema. Com essa visão, os agentes região fazem uma recomendação abstrata a qual afetará o aprendizado e as recompensas dos seus subordinados. Os subordinados podem seguir ou não essa recomendação, podendo ou não serem incentivados ou punidos por suas ações. As recompensas dos agentes subordinados são agregadas para formar a recompensa do seu agente região supervisor, logo o desempenho dos agentes região depende do desempenho dos seus subordinados. Desta maneira, os agentes nos diferentes níveis hierárquicos guiam o aprendizado dos seus agentes subordinados para um melhor desempenho coletivo.

No VHO, a aplicação proposta apresentada no Capítulo 4, adaptamos o modelo proposto para o problema de controle semafórico utilizando operações com vetores. Os

pontos principais do modelo, a informação processada dos agentes subordinados, a recomendação dos supervisores e a influência nas recompensas dos subordinados, ganham formas específicas para solucionar o problema: os vetores resultantes, o vetor de indicação de fluxo de tráfego e o incentivo - conforme o fluxo de tráfego liberado ou indicação de fluxo - utilizando a função cosseno.

Os experimentos dessa dissertação (Capítulo 5) foram realizados em um simulador de tráfego microscópico, o SUMO (Seção 5.1), e em um cenário sintético de rede em *grid* 4x4. Foram realizados quatro experimentos: três comparando diferentes métodos de controle semafórico, em três definições de demanda diferentes, e um comprovando a importância do aprendizado por reforço nos agentes dos diferentes níveis hierárquicos da aplicação proposta. Nos três primeiros experimentos, são comparados: um método de tempo fixo, um método com aprendizado por reforço sem organização hierárquica e um método que utiliza a aplicação proposta. Os resultados desses três experimentos comprovam o melhor desempenho do modelo proposto, que utiliza aprendizado por reforço com organização hierárquica para controle semafórico. O quarto experimento comprova que é necessário realizar aprendizado nos diferentes níveis hierárquicos do modelo proposto para se obter um melhor desempenho.

## 6.2 Contribuições

Do ponto de vista de técnicas de aprendizado por reforço, o modelo genérico apresentado nessa dissertação representa uma contribuição para técnicas que utilizam organização hierárquica em conjunto com aprendizado por reforço. As ideias principais do modelo proposto são: a recomendação abstrata que deve estar relacionada às ações realizadas dos agentes de diferentes níveis; e, a agregação das informações e recompensas dos agentes de níveis mais baixos para seus supervisores. A estrutura genérica apresentada se diferencia de outras técnicas com organização hierárquica para aprendizado por reforço pois: pode dividir uma tarefa em sub tarefas a serem alocadas a agentes de níveis mais baixos, mas não necessariamente o faz; os agentes de níveis mais altos não possuem controle total sobre seus subordinados, apenas influenciam em seus aprendizados; e, há uma dependência entre os agentes de diferentes níveis hierárquicos, visto que os agentes de níveis mais altos dependem das informações e das recompensas dos agentes de níveis mais baixos.

Em geral, o diferencial do método genérico é a utilização de abstrações nas in-

terações entre agentes de diferente níveis (estados e recomendações) que o torna mais flexível. Contudo, o método limita-se à possibilidade de adaptação dessas abstrações para um problema em específico e ao fato de ser vantajoso a utilização de uma organização hierárquica para a resolução de um problema. Não sendo recomendado, por exemplo, em problemas com apenas um agente por nível hierárquico ou onde abstrações de informações não são possíveis ou não recomendadas.

Do ponto de vista de técnicas realizadas em controle semafórico, a aplicação proposta nessa dissertação é uma contribuição às técnicas de aprendizado por reforço com organização hierárquica para controle semafórico. A aplicação utiliza operações com vetores para realizar as diferentes interações entre os níveis hierárquicos. As características que diferenciam a aplicação proposta de outros trabalhos da área são: a aplicação é facilmente configurável para diferentes números de níveis hierárquicos e definições de regiões; os agentes região consideram as informações de todos os subordinados dentro de sua região, não tratando-a como uma caixa-preta; e, o fato de realizar operações com vetores permite a utilização das diferentes orientações geográficas da rede transporte sem necessidade de alterar o modelo.

Em geral, o diferencial da aplicação do método genérico para o controle semafórico é a utilização dos vetores para abstrair e relacionar as informações e ações dos agentes interseção e agentes região. Contudo, a aplicação é limitada ao fato dos cálculos com vetores propostos serem relevantes em problemas que trabalhem com apenas duas dimensões e, no próprio contexto de controle semafórico, ao fato das informações serem de difícil obtenção no mundo real.

## 6.3 Perspectivas de Continuidade

Com relação ao modelo genérico proposto, investigações futuras em trabalhos fora do contexto de controle semafórico poderiam ser realizadas para analisar o desempenho do modelo em outras áreas de aplicação. Mantendo-se as ideias principais de agregação de informações, relação entre ações e influência entre os diferentes níveis hierárquicos, o modelo pode ser adaptado para resolução de qualquer problema de aprendizado por reforço no qual seja vantajoso utilizar uma organização hierárquica, dependendo apenas da criatividade do projetista. Na aplicação para controle semafórico, temos relações idênticas entre os diferentes níveis hierárquicos: sempre é feita uma soma vetorial das informações dos subordinados; a relação entre as ações tomadas sempre é uma função cosseno; e a

recompensa dos supervisores sempre é uma média aritmética simples das recompensas dos subordinados. Investigações em trabalhos onde essas relações não sejam constantes, por exemplo, no qual um nível hierárquico seja uma média aritmética simples e em outro seja uma média ponderada, podem ser interessantes e comprovariam o carácter versátil do modelo genérico proposto.

Com relação à aplicação para controle semafórico proposta, são listadas as seguintes possibilidades futuras:

- Experimentação com diferentes algoritmos de aprendizado por reforço. Entre eles, algoritmo de aprendizado profundo e, principalmente, algoritmos que utilizem funções com construção de *features*, que levam em consideração as relações das diferentes variáveis de estado e que, assim, podem melhorar o desempenho da aplicação.

- Experimentação com diferentes funções de incentivo, por exemplo, aplicando punições;

- Experimentação com diferentes definições de regiões e diferentes definições de demandas que possam apresentar resultados significativos quando realizadas em conjunto;

- Experimentação em cenários com demandas mais realísticas;

- Experimentação em cenários que possuam interseções mais relevantes em determinadas regiões, podendo adaptar o modelo para dar mais importância às informações dessas interseções em específico.

# REFERÊNCIAS


ABDOOS, M.; BAZZAN, A. L. Hierarchical traffic signal optimization using reinforcement learning and traffic prediction with long-short term memory. **Expert Systems with Applications**, p. 114580, 2021. ISSN 0957-4174.

ABDOOS, M.; MOZAYANI, N.; BAZZAN, A. L. Holonic multi-agent system for traffic signals control. **Engineering Applications of Artificial Intelligence**, v. 26, n. 5–6, p. 1575–1587, 2013. ISSN 0952-1976.

ABDOOS, M.; MOZAYANI, N.; BAZZAN, A. L. Hierarchical control of traffic signals using Q-learning with tile coding. **Appl. Intell.**, Springer US, v. 40, n. 2, p. 201–213, 2014.

ALEGRE, L. N. **SUMO-RL**. [S.l.]: GitHub, 2019. <https://github.com/LucasAlegre/sumo-rl>.

ASLANI, M.; MESGARI, M. S.; WIERING, M. Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. **Transportation Research Part C: Emerging Technologies**, v. 85, p. 732–752, 2017. ISSN 0968-090X.

ASLANI, M. et al. Traffic signal optimization through discrete and continuous reinforcement learning with robustness analysis in downtown tehran. **Advanced Engineering Informatics**, v. 38, p. 639–655, 2018. ISSN 1474-0346.

BAZZAN, A. L.; OLIVEIRA, D. d.; SILVA, B. C. d. Learning in groups of traffic signals. **Engineering Applications of Artificial Intelligence**, v. 23, n. 4, p. 560 – 568, 2010. ISSN 0952-1976.

BAZZAN, A. L. C. Opportunities for multiagent systems and multiagent reinforcement learning in traffic control. **Autonomous Agents and Multiagent Systems**, v. 18, n. 3, p. 342–375, June 2009.

BUŞONIU, L.; BABUSKA, R.; SCHUTTER, B. D. A comprehensive survey of multiagent reinforcement learning. **Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on**, IEEE, v. 38, n. 2, p. 156–172, 2008.

CEYLAN, H.; BELL, M. Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. **Transportation Research Part B: Methodological**, v. 38, p. 329–342, 2004.

CHOY, M. C.; SRINIVASAN, D.; CHEU, R. L. Cooperative, hybrid agent architecture for real-time traffic signal control. **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, v. 33, n. 5, p. 597–607, 2003.

CHOY, M. C.; SRINIVASAN, D.; CHEU, R. L. Neural networks for continuous online learning and control. **IEEE Transactions on Neural Networks**, v. 17, n. 6, p. 1511–1531, 2006.

CHU, T. et al. Multi-agent deep reinforcement learning for large-scale traffic signal control. **IEEE Transactions on Intelligent Transportation Systems**, v. 21, n. 3, p. 1086–1095, 2019.

CLAUS, C.; BOUTILIER, C. The dynamics of reinforcement learning in cooperative multiagent systems. In: **Proceedings of the Fifteenth National Conference on Artificial Intelligence**. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1998. (AAAI '98), p. 746–752.

DAYAN, P.; HINTON, G. E. Feudal reinforcement learning. In: HANSON, S. J.; COWAN, J. D.; GILES, C. L. (Ed.). **Advances in Neural Information Processing Systems 5**. [S.l.]: Morgan-Kaufmann, 1993. p. 271–278.

DIETTERICH, T. G. **Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition**. 1999.

FRANCE, J.; GHORBANI, A. A. A multiagent system for optimizing urban traffic. In: **Proceedings of the IEEE/WIC International Conference on Intelligent Agent Technology**. Washington, DC, USA: IEEE Computer Society, 2003. p. 411–414. ISBN 0-7695-1931-8.

FRANKLIN, S.; GRAESSER, A. It is an agent, or just a program? a taxonomy for autonomous agents. In: SPRINGER (Ed.). **Intelligent Agents**. [S.l.]: Jennings, N. and Wooldridge, M., 1997. III.

GAUD, N. **Holonic Multi-Agent Systems: From the analysis to the implementation. Metamodel, Methodology and Multilevel simulation**. Thesis (PhD) — Université de Technologie de Belfort-Montbéliard, Belfort, France, 12 2007.

GERBER, C.; SIEKMANN, J.; VIERKE, G. **Holonic Multi-Agent Systems**. [S.l.]: DFKI, 1999. 42 p. (Deutsches Forschungszentrum für Künstliche Intelligenz).

HAYNES, W. Tukey's test. In: ____. **Encyclopedia of Systems Biology**. New York, NY: Springer New York, 2013. p. 2303–2304. ISBN 978-1-4419-9863-7.

HUNT, P. B. et al. **SCOOT - A Traffic Responsive Method of Coordinating Signals**. Berkshire, 1981.

KAUFMANN, J.; SCHERING, A. Analysis of variance anova. In: ____. **Wiley StatsRef: Statistics Reference Online**. [S.l.]: American Cancer Society, 2014. ISBN 9781118445112.

KOESTLER, A. **The ghost in the machine**. The danube ed., 2. impr. [S.l.]: Hutchinson, 1979. 381 p. ISBN 0091271304.

LOPEZ, P. A. et al. Microscopic traffic simulation using sumo. In: **The 21st IEEE International Conference on Intelligent Transportation Systems**. [S.l.: s.n.], 2018.

LOWRIE, P. The Sydney coordinate adaptive traffic system - principles, methodology, algorithms. In: **Proceedings of the International Conference on Road Traffic Signalling**. Sydney, Australia: [s.n.], 1982.

MA, J.; WU, F. Feudal multi-agent deep reinforcement learning for traffic signal control. In: **Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)**. Auckland, New Zealand: [s.n.], 2020. p. 816–824.

MANNION, P.; DUGGAN, J.; HOWLEY, E. An experimental review of reinforcement learning algorithms for adaptive traffic signal control. In: MCCLUSKEY, T. L. et al. (Ed.). **Autonomic Road Transport Support Systems**. Cham: Springer International Publishing, 2016. p. 47–66.

NISHI, T. et al. Traffic signal control based on reinforcement learning with graph convolutional neural nets. In: **21st International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2018. p. 877–883.

PAPPIS, C. P.; MAMDANI, E. H. A fuzzy logic controller for a trafc junction. **IEEE Transactions on Systems, Man, and Cybernetics**, v. 7, n. 10, p. 707–717, 1977.

ROESS, R.; PRASSAS, E.; MCSHANE, W. **Traffic engineering**. 4th. ed. [S.l.]: Prentice Hall, 2011. 714 p.

ROOZEMOND, D. A. Using intelligent agents for pro-active, real-time urban intersection control. **European Journal of Operational Research**, v. 131, n. 2, p. 293 – 301, 2001. ISSN 0377-2217. Artificial Intelligence on Transportation Systems and Science.

RUMMERY, G.; NIRANJAN, M. **On-Line Q-Learning Using Connectionist Systems**. [S.l.], 1994.

SUTTON, R. S.; BARTO, A. G. **Reinforcement learning: An introduction**. Second. [S.l.]: The MIT Press, 2018.

TCHAPPI, I. H. et al. A critical review of the use of holonic paradigm in traffic and transportation systems. **Engineering Applications of Artificial Intelligence**, v. 90, p. 1–54, 2020. ISSN 0952-1976.

WEBSTER, F. V. **Traffic Signal Setting**. London, 1958.

WEGENER, A. et al. TraCI: an interface for coupling road traffic and network simulators. In: ACM. **11th communications and networking simulation symposium**. [S.l.], 2008. p. 155–163.

WEI, H. et al. Colight: Learning network-level cooperation for traffic signal control. In: **Proceedings of the 28th ACM International Conference on Information and Knowledge Management**. [S.l.]: Association for Computing Machinery, 2019. p. 1913–1922. ISBN 9781450369763.

WEI, H. et al. **A Survey on Traffic Signal Control Methods**. 2019.

WEI, H. et al. Intellilight: A reinforcement learning approach for intelligent traffic light control. In: **Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: Association for Computing Machinery, 2018. (KDD '18), p. 2496–2505. ISBN 9781450355520.

WOOLDRIDGE, M. J. **An Introduction to MultiAgent Systems**. Chichester: John Wiley & Sons, 2009. 461 p. Second edition.

YAU, K.-L. A. et al. A survey on reinforcement learning models and algorithms for traffic signal control. **ACM Comput. Surv.**, ACM, v. 50, n. 3, 2017. ISSN 0360-0300.

ZHAO, D.; DAI, Y.; ZHANG, Z. Computational intelligence in urban traffic signal control: A survey. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, v. 42, n. 4, p. 485–494, 2012.

ZHENG, G. et al. Learning phase competition for traffic signal control. In: **Proceedings of the 28th ACM International Conference on Information and Knowledge Management**. New York, NY, USA: Association for Computing Machinery, 2019. p. 1963–1972. ISBN 9781450369763.

ZHENG, G. et al. **Diagnosing Reinforcement Learning for Traffic Signal Control**. 2019.

# APÊNDICE A — CRIAÇÃO DOS ARQUIVOS DE ROTAS

Para criar os arquivo de rotas, inicialmente são criados os arquivos de fluxos de veículos por par OD. No exemplo da Fig. A.1, temos 500 veículos por hora por par OD. Tendo esses arquivos criados, eles servem de entrada para a ferramenta do SUMO, *duaiterate.py* que, após um número definido de iterações realizando a alocação dinâmica de usuários (*dynamic user assignement*), criam os arquivos de definição de rotas. Os arquivos de rotas definem, para cada veículos, o tempo de partida e a rota pela qual ele irá viajar. Nos experimentos, os arquivos de rotas foram criados executando o *duaiterate.py* com 5 iterações.

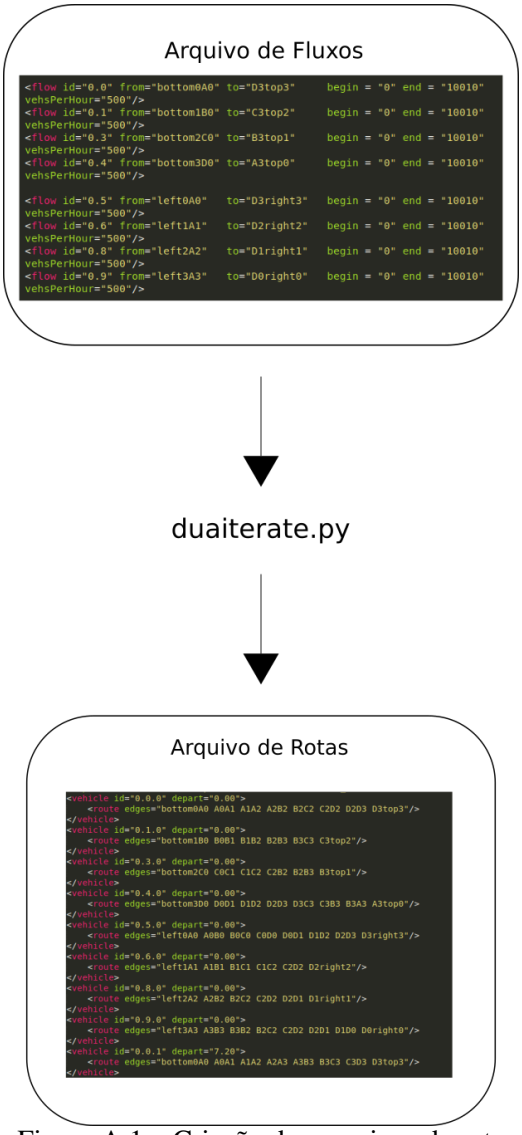



Figura A.1 – Criação dos arquivos de rota

# APÊNDICE B — ARQUIVO DE DEFINIÇÃO DE ORDEM HIERÁRQUICA E SUBORDINADOS

Exemplo de um arquivo de definição de ordem hierárquica e subordinados onde temos dois níveis hierárquicos de região: quatro regiões no primeiro nível R11, R12, R13, R14 e uma região no segundo nível R21. O primeiro nível de região controla 16 interseções A0, A1, ... , D3 e o segundo nível de região controla as regiões do primeiro nível.

```
{
  "Order": [ ["R11", "R12", "R13", "R14"], [ "R21" ] ],
  "RegionsDict":
  {
    "R11": { "subordinates": [ "A3", "B3", "A2", "B2" ] },
    "R12": { "subordinates": [ "C3", "D3", "C2", "D2" ] },
    "R13": { "subordinates": [ "A1", "B1", "A0", "B0" ] },
    "R14": { "subordinates": [ "C1", "D1", "C0", "D0" ] },
    "R21": { "subordinates": [ "R11", "R12", "R13", "R14" ] }
  }
}
```

# APÊNDICE C — EXEMPLO ARQUIVO DE MÉTRICAS DE UMA SIMULAÇÃO

| step_time | recompensa | veiculos_na_rede | vei_partindo | vei_chegando | ocupacao | tempo_de_espera | vei_parados | velocidade_media |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 1 | 0 | 8 | 8 | 0 | 0,0035 | 0 | 0,5 | 13,0219 |
| 2 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,1368 |
| 3 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,2553 |
| 4 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,3842 |
| 5 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,5066 |
| 6 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,6189 |
| 7 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,7482 |
| 8 | 0 | 8 | 0 | 0 | 0,0035 | 0 | 0 | 13,8163 |
| 9 | 0 | 16 | 8 | 0 | 0,007 | 0 | 0,5 | 13,4237 |
| 10 | 0 | 16 | 0 | 0 | 0,0068 | 0 | 0 | 13,4055 |
| 11 | 0 | 16 | 0 | 0 | 0,0054 | 0 | 0 | 13,3073 |
| 12 | 0 | 16 | 0 | 0 | 0,0057 | 0 | 0 | 13,337 |
| 13 | 0 | 16 | 0 | 0 | 0,0061 | 0,0625 | 0,0625 | 13,3788 |
| 14 | 0 | 16 | 0 | 0 | 0,0061 | 0,25 | 0,1875 | 13,4679 |
| 15 | -0,5 | 16 | 0 | 0 | 0,0061 | 0,5 | 0,25 | 13,4412 |
| ⋮ | | | | | | | | |
| 13433 | -1700,875 | 442 | 1 | 0 | 0,1046 | 1663,4375 | 21,5625 | 10,7347 |
| 13434 | -1700,875 | 443 | 1 | 0 | 0,1046 | 1656,9375 | 21,25 | 10,8122 |
| 13435 | -1639,375 | 442 | 0 | 1 | 0,1045 | 1639,375 | 21 | 10,836 |
| 13436 | -1639,375 | 442 | 0 | 0 | 0,1044 | 1635,625 | 20,8125 | 10,8637 |
| 13437 | -1639,375 | 442 | 0 | 0 | 0,1047 | 1617,6875 | 20,75 | 10,8421 |
| 13438 | -1639,375 | 442 | 0 | 0 | 0,1048 | 1604,5625 | 20,625 | 10,8599 |
| 13439 | -1639,375 | 441 | 0 | 1 | 0,1047 | 1605,625 | 20,875 | 10,9345 |
| 13440 | -1598,25 | 440 | 0 | 1 | 0,1046 | 1598,25 | 20,6875 | 10,902 |
| 13441 | -1598,25 | 439 | 0 | 1 | 0,1048 | 1590,3125 | 20,9375 | 10,8466 |
| 13442 | -1598,25 | 439 | 0 | 0 | 0,1049 | 1589,1875 | 20,875 | 10,8539 |
| 13443 | -1598,25 | 439 | 0 | 0 | 0,1048 | 1592,4375 | 21,0625 | 10,8108 |
| 13444 | -1598,25 | 439 | 0 | 0 | 0,1049 | 1597,75 | 21,125 | 10,7563 |
| 13445 | -1615,375 | 437 | 0 | 2 | 0,105 | 1615,375 | 21,3125 | 10,7628 |
| 13446 | -1615,375 | 437 | 0 | 0 | 0,1048 | 1600,6875 | 21 | 10,6221 |
| 13447 | -1615,375 | 437 | 0 | 0 | 0,1048 | 1590,4375 | 21,0625 | 10,5231 |
| 13448 | -1615,375 | 437 | 0 | 0 | 0,1047 | 1592,9375 | 21,1875 | 10,503 |
| 13449 | -1615,375 | 436 | 0 | 1 | 0,1046 | 1598,9375 | 21,5625 | 10,57 |
| ⋮ | | | | | | | | |
| 24983 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24984 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24985 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24986 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24987 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24988 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24989 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24990 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24991 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24992 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24993 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24994 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24995 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24996 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24997 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24998 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |
| 24999 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13,89 |

# APÊNDICE D — RESULTADOS DOS TESTES ESTATÍSTICOS ANOVA E TUKEY

Os testes estatísticos de ANOVA (KAUFMANN; SCHERING, 2014) e Tukey (HAYNES, 2013) recebem como entrada os valores encontrados em cada simulação para cada métrica. Para testes que comparam 3 ou mais grupos, o teste ANOVA aponta que há diferença estatística entre os métodos, mas não especifica onde essa diferença ocorre. Logo, para testes com 3 ou mais grupos, realizamos o teste Tukey par a par para mostrar a diferença entre os grupos. Para teste com 2 grupos, o teste ANOVA é considerado suficiente. Normalmente, os resultados são considerados estatisticamente diferente quando o valor P do teste estatístico é menor que 0,05. As tabelas abaixo apresentam os valores dos testes estatísticos para os diferentes experimentos realizados nessa dissertação.

Tabela D.1 – Experimento Padrão - Resultados ANOVA e Tukey

| **Métrica** | **ANOVA** | | **Tukey** | | |
|---|---|---|---|---|---|
| | | | Aprend. | Aprend. | VHO |
| | F value | P value | VHO | Webster | Webster |
| Recompensa | 646,25 | 1,57E-23 | 0,001 | 0,001 | 0,001 |
| Fluxo de viagens concluídas (vei/s) | 93,95 | 6,88E-13 | 0,007 | 0,001 | 0,001 |
| Tempo para suprir demanda (s) | 180,08 | 2,40E-16 | 0,011 | 0,001 | 0,001 |
| Tempo médio de espera durante a viagem (s) | 18,19 | 9,90E-06 | 0,16 | 0,001 | 0,001 |
| Tempo médio perdido durante viagem (s) | 20,15 | 4,40E-06 | 0,27 | 0,001 | 0,001 |
| Delay médio de partida (s) | 189,63 | 1,20E-16 | 0,001 | 0,001 | 0,001 |

Tabela D.2 – Experimento NS - Resultados ANOVA e Tukey

| **Métrica** | **ANOVA** | | **Tukey** | | |
|---|---|---|---|---|---|
| | | | Aprend. | Aprend. | VHO |
| | F value | P value | VHO | Webster | Webster |
| Recompensa | 147,58 | 3,75E-15 | 0,013 | 0,001 | 0,001 |
| Fluxo de viagens concluídas (vei/s) | 57,7 | 1,77E-10 | 0,04 | 0,001 | 0,001 |
| Tempo para suprir demanda (s) | 72,81 | 1,32E-11 | 0,03 | 0,001 | 0,001 |
| Tempo médio de espera durante a viagem (s) | 90,03 | 1,13E-12 | 0,9 | 0,001 | 0,001 |
| Tempo médio perdido durante viagem (s) | 144,58 | 3,75E-15 | 0,9 | 0,001 | 0,001 |
| Delay médio de partida (s) | 191,78 | 1,10E-16 | 0,16 | 0,001 | 0,001 |

Tabela D.3 – Experimento LO - Resultados ANOVA e Tukey

| **Métrica** | **ANOVA** | | **Tukey** | | |
|---|---|---|---|---|---|
| | | | Aprend. | Aprend. | VHO |
| | F value | P value | VHO | Webster | Webster |
| Recompensa | 238,82 | 6,80E-18 | 0,001 | 0,001 | 0,001 |
| Fluxo de viagens concluídas (vei/s) | 115,28 | 5,97E-14 | 0,002 | 0,001 | 0,001 |
| Tempo para suprir demanda (s) | 143,4 | 4,15E-15 | 0,003 | 0,001 | 0,001 |
| Tempo médio de espera durante a viagem (s) | 148,65 | 2,66E-15 | 0,009 | 0,001 | 0,001 |
| Tempo médio perdido durante viagem (s) | 212,96 | 2,93E-17 | 0,028 | 0,001 | 0,001 |
| Delay médio de partida (s) | 276,29 | 1,05E-18 | 0,079 | 0,001 | 0,001 |

Tabela D.4 – Experimento de Qualidade de Aprendizado - Resultados ANOVA

| **Métrica** | **ANOVA** | |
|---|---|---|
| | VHO- vs VHO | |
| | F value | P value |
| Recompensa | 92,79 | 1,58E-08 |
| Fluxo de viagens concluídas (vei/s) | 8,32 | 0,009 |
| Tempo para suprir demanda (s) | 7,97 | 0,011 |
| Tempo médio de espera durante a viagem (s) | 11,68 | 0,003 |
| Tempo médio perdido durante viagem (s) | 10,31 | 0,004 |
| Delay médio de partida (s) | 23,92 | 0,0001 |