

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**ALEXANDRE LIZIERI LEITE MELLADO**

**IMPLEMENTAÇÃO DE UM MECANISMO DE CONTROLE DE PREÇOS EM UM  
SISTEMA MULTIAGENTE PARA GERENCIAMENTO DE VAGAS EM UM SMART  
PARKING**

**PONTA GROSSA**

**2021**

**ALEXANDRE LIZIERI LEITE MELLADO**

**IMPLEMENTAÇÃO DE UM MECANISMO DE CONTROLE DE PREÇOS EM UM SISTEMA MULTIAGENTE PARA GERENCIAMENTO DE VAGAS EM UM SMART PARKING**

**Implementation of a price control mechanism in a multi-agent system to manager spaces in a smart parking**

Trabalho de conclusão de curso de graduação apresentada como requisito para obtenção do título de Bacharel da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador(a): André Pinz Borges

Coorientador(a): Gleifer Vaz Alves

**PONTA GROSSA**

**2021**



[4.0 Internacional](https://creativecommons.org/licenses/by-sa/4.0/)

Esta licença permite remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es) e que licenciem as novas criações sob termos idênticos. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

**ALEXANDRE LIZIERI LEITE MELLADO**

**IMPLEMENTAÇÃO DE UM MECANISMO DE CONTROLE DE PREÇOS EM UM SISTEMA  
MULTIAGENTE PARA GERENCIAMENTO DE VAGAS EM UM SMART PARKING**

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Ciência da Computação da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 19 de novembro de 2021

---

André Pinz Borges  
Doutorado  
Universidade Tecnológica Federal do Paraná

---

Gleifer Vaz Alves  
Doutorado  
Universidade Tecnológica Federal do Paraná

---

Paulo Leitão  
Doutorado  
Instituto Politécnico de Bragança

---

André Koscianski  
Doutorado  
Universidade Tecnológica Federal do Paraná

**PONTA GROSSA**

**2021**

## RESUMO

Longos intervalos de tempo a procura por uma vaga de estacionamento causam o desperdício de combustível e maior congestionamento. Em grandes centros urbanos, onde o crescimento da população foi mais rápido que os investimentos infraestruturais, ocorre um problema de mobilidade que necessita de soluções inovadoras. O paradigma de *Smart City* (Cidade Inteligente) consiste no desenvolvimento e integração de novas tecnologias de modo a solucionar os problemas de trânsito de uma cidade. A falta de vagas disponíveis e o problema de mobilidade são abrangidos na área de *Smart Parking* (Estacionamento Inteligente) e o projeto MAPS (Multi-Agent Parking System) almeja criar soluções para estacionamentos com a utilização de Sistemas Multiagentes. Este trabalho é uma extensão deste projeto e tem o objetivo de desenvolver um módulo de precificação dinâmico que altera o preço das vagas de um estacionamento considerando as características de seu ambiente. Este mecanismo de precificação foi implantado em um Sistema Multiagente e um agente gerente decide o momento de seu uso. O módulo tem a função de observar a demanda por vagas, o horário do dia e a condição climática e decide a partir de regras contidas em um arquivo de configuração como o preço deve ser alterado. Foram realizadas simulações de estacionamentos com diferentes aspectos usando seis distintos arquivos de configuração e cenários de controle nos quais o módulo não foi utilizado. Os resultados encontrados mostraram que o módulo de precificação possui influência significativa no funcionamento do estacionamento, porém apresentam valores oscilantes devido características pseudorrandômicas. Em geral, o uso de qualquer configuração proporcionou melhores resultados em pelo menos um quesito, seja este maior total de créditos ao fim do mês ou maior ocupação média das vagas.

Palavras-chave: Sistema Multiagente; Estacionamento Inteligente; Agentes Inteligentes; Precificação Dinâmica; Framework JaCaMo.

## ABSTRACT

Long periods of time searching for a parking space causes wasted fuel and more congestion. In large urban centers, where population growth has been faster than infrastructure investments, a mobility problem occurs that needs innovative solutions. The Smart City paradigm consists of the development and integration of new technologies in order to solve the traffic problems of a city. The lack of available parking spaces and the mobility problem are covered in the Smart Parking area and the MAPS (Multi-Agent Parking System) project aims to create solutions for parking lots with the use of Multi-agent Systems. This work is an extension of this project and has the objective to develop a dynamic pricing module that alters the price of parking spaces considering the characteristics of its environment. This pricing mechanism was implemented in a Multiagent System and a managing agent decides the moment of its use. The module has the function of observing the demand for parking spaces, the time of day and the weather condition and decides from rules contained in a conFIGuration file how the price should be changed. Simulations of parking lots with different aspects were run using six different conFIGuration files and control scenarios in which the module was not used. The results found showed that the pricing module has significant influence on the operation of the parking facility, but has oscillating values due to pseudorandom characteristics. In general, the use of any conFIGuration provided better results in at least one aspect, be it higher total credits at the end of the month or higher average occupancy of the spaces.

Keywords: Multiagent System; Smart Parking; Intelligent Agents; Dynamic Pricing; Framework JaCaMo.

## LISTA DE ALGORITMOS

Algoritmo 1 - Criação de crenças em Jason.....	23
Algoritmo 2 - Anotações em crenças em Jason.....	25
Algoritmo 3 - Plano interno em jason.....	26
Algoritmo 4 - Inicialização de um artefato.....	28
Algoritmo 5 - Definição das crenças do agente <i>driver</i> .....	54
Algoritmo 6 - Plano de espera até chegar ao estacionamento.....	55
Algoritmo 7 - Pedir preço atual das vagas.....	55
Algoritmo 8 - Confirmar preço recebido.....	55
Algoritmo 9 - Requisitar por uma vaga.....	56
Algoritmo 10 - Estacionar na vaga recebida.....	56
Algoritmo 11 - Sair da vaga.....	56
Algoritmo 12 - Inicialização dos artefatos e organização das vagas.....	57
Algoritmo 13 - Observação do ambiente e atualização do preço.....	58
Algoritmo 14 - Consulta do preço e notificação do <i>driver</i> .....	58
Algoritmo 15 - Processamento de uma requisição por vaga.....	58
Algoritmo 16 - Salvando requisições por cada tipo de vaga.....	59
Algoritmo 17 - Retirando <i>driver</i> da vaga.....	59
Algoritmo 18 - Alocação de uma vaga, se disponível, a um <i>driver</i> .....	60
Algoritmo 19 - Operação de atualização do preço.....	63
Algoritmo 20 - Inserção de driver na lista de espera.....	64
Algoritmo 21 - Remoção de driver na lista de espera.....	65
Algoritmo 22 - Exemplo de crenças das vagas.....	66

## LISTA DE ILUSTRAÇÕES

Figura 1 - Representação de um agente inteligente.....	17
Figura 2 - Representação de um sistema multiagente.....	18
Figura 3 - Representação de um modelo BDI.....	20
Figura 4 - Divisão dos níveis do JaCaMo.....	21
Figura 5 - Tipos de termos do AgentSpeak no Jason.....	23
Figura 6 - Tipos de gatilhos de evento.....	26
Figura 7 - Camadas básicas do modelo PARKAGENT.....	33
Figura 8 - Segmentação da área do estacionamento.....	34
Figura 9 - Agrupamento dos trabalhos relacionados.....	41
Figura 10 - Conexão entre agente <i>driver</i> e <i>manager</i> .....	51
Figura 11 - Modelagem do SMA de um Estacionamento Inteligente.....	52
Figura 12 - Criação das matrizes usando o arquivo de configuração 6.....	61
Figura 13 - Exemplo de verificação do ambiente usando o arquivo de configuração 6 .....	62
Figura 14 - Log de eventos criado durante execução do sistema usando o arquivo de configuração 6.....	64
Figura 15 - Exemplo de Crenças dos Setores e Tipos de Vagas.....	66
Figura 16 - Utilização do módulo de precificação pelo agente <i>manager</i> .....	70
Gráfico 1 - Ocupações e preços em 100 vagas, 50 <i>drivers</i> e conjunto Baixo.....	83
Gráfico 2 - Ocupações e preços em 100 vagas, 100 <i>drivers</i> e conjunto Baixo.....	86
Gráfico 3 - Ocupações e preços em 100 vagas, 100 <i>drivers</i> e conjunto Médio.....	89
Gráfico 4 - Ocupações e preços em 100 vagas, 200 <i>drivers</i> e conjunto Baixo.....	92
Gráfico 5 - Ocupações e preços em 100 vagas, 200 <i>drivers</i> e conjunto Alto.....	94
Gráfico 6 - Ocupações e preços em 200 vagas, 50 <i>drivers</i> e conjunto Baixo.....	98
Gráfico 7 - Ocupações e preços em 200 vagas, 100 <i>drivers</i> e conjunto Baixo.....	101
Gráfico 8 - Ocupações e preços em 200 vagas, 100 <i>drivers</i> e conjunto Médio.....	104
Gráfico 9 - Ocupações e preços em 200 vagas, 200 <i>drivers</i> e conjunto Baixo.....	107
Gráfico 10 - Ocupações e preços em 200 vagas, 200 <i>drivers</i> e conjunto Alto.....	110
Gráfico 11 - Comparação dos resultados dos cenário sem uso do módulo.....	121
Gráfico 12 - Comparação dos resultados da configuração 1.....	122
Gráfico 13 - Comparação dos resultados da configuração 2.....	122
Gráfico 14 - Comparação dos resultados da configuração 3.....	123
Gráfico 15 - Comparação dos resultados da configuração 4.....	124
Gráfico 16 - Comparação dos resultados da configuração 5.....	125
Gráfico 17 - Comparação dos resultados da configuração 6.....	126
Quadro 1 - Comparação entre estratégias de estacionamento.....	40
Quadro 2 - Comparação entre os trabalhos relacionados.....	50

## LISTA DE TABELAS

Tabela 1 - Configuração com duas categorias.....	68
Tabela 2 - Configuração com três categorias.....	68
Tabela 3 - Variáveis e valores dos cenários de teste.....	74
Tabela 4 - Configuração 1.....	75
Tabela 5 - Configuração 2.....	75
Tabela 6 - Configuração 3.....	76
Tabela 7 - Configuração 4.....	76
Tabela 8 - Configuração 5.....	77
Tabela 9 - Configuração 6.....	77
Tabela 10 - Tabela de conjuntos.....	78
Tabela 11 - Resultados dos cenários simulados.....	79
Tabela 12 - Comparação dos maiores e menores resultados de cada cenário.....	114



## LISTA DE ABREVIATURAS E SIGLAS

BDI	<i>Belief, Desire and Intention</i>
CArtAgO	<i>Common ARTifact infrastructure for AGents Open environments</i>
EI.	<i>Estacionamento Inteligente</i>
JaCaMo	<i>JASON + CArtAgo + Moise</i>
JASON	<i>A Java-based interpreter for an extended version of AgentSpeak</i>
MAPS	<i>Multi-Agent Parking System</i>
SMA	<i>Sistema Multiagente</i>
UTFPR	<i>Universidade Tecnológica Federal do Paraná</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
<b>1.1</b>	<b>Objetivo</b>	<b>13</b>
1.1.1	Objetivo Geral	13
1.1.2	Objetivos Específicos	14
<b>1.2</b>	<b>Justificativa</b>	<b>14</b>
<b>1.3</b>	<b>Trabalhos Publicados</b>	<b>14</b>
<b>1.4</b>	<b>Organização do Trabalho</b>	<b>15</b>
<b>2</b>	<b>SISTEMAS MULTIAGENTES</b>	<b>16</b>
<b>2.1</b>	<b>Agentes Inteligentes</b>	<b>16</b>
2.1.1	Autonomia	17
2.1.2	Proatividade	17
2.1.3	Reatividade	18
2.1.4	Habilidade Social	18
<b>2.2</b>	<b>Modelo BDI</b>	<b>19</b>
<b>2.3</b>	<b>Framework JaCaMo</b>	<b>21</b>
2.3.1	Jason	22
<u>2.3.1.1</u>	<u>Crenças</u>	<u>23</u>
<u>2.3.1.2</u>	<u>Objetivos</u>	<u>25</u>
<u>2.3.1.3</u>	<u>Planos</u>	<u>25</u>
2.3.2	Cartago	27
<u>2.3.2.1</u>	<u>Workspaces</u>	<u>27</u>
<u>2.3.2.2</u>	<u>Repertório de Ações e Artefatos</u>	<u>27</u>
<u>2.3.2.3</u>	<u>Integração Jason-Cartago</u>	<u>27</u>
2.3.3	Moise	28
2.3.4	Considerações Finais	28
<b>3</b>	<b>ESTACIONAMENTOS INTELIGENTES</b>	<b>29</b>
<b>3.1</b>	<b>Benefícios de Estacionamentos Inteligentes</b>	<b>29</b>
<b>3.2</b>	<b>Funções de estacionamentos inteligentes</b>	<b>30</b>
<b>3.3</b>	<b>Gerenciamento de Estacionamento</b>	<b>30</b>
3.3.1	Tipos de Estacionamentos	31
3.3.2	Estratégias de estacionamentos	32

3.3.2.1	Grau de Confiança	32
3.3.2.2	Distância até o Destino	33
3.3.2.3	Preço da Vaga	35
3.3.2.4	Preço e Distância	35
<b>3.4</b>	<b>Precificação Dinâmica</b>	<b>35</b>
3.4.1	Função de Demanda Dinâmica	36
3.4.2	Função de Demanda Estática	37
3.4.2	Demanda Dinâmica e Aprendizado	37
3.4.2.1	Inventário Infinito	38
3.4.2.2	Inventário Finito	38
3.4.2.3	Aprendizado de Máquina	38
<b>3.5</b>	<b>Considerações Finais</b>	<b>39</b>
<b>4</b>	<b>TRABALHOS RELACIONADOS</b>	<b>41</b>
4.1	Smart Parking	42
4.2	Agentes	43
4.3	Alocação de Recurso	45
4.4	Precificação Dinâmica	46
4.5	Projeto MAPS	48
<b>5</b>	<b>DESENVOLVIMENTO DO TRABALHO</b>	<b>51</b>
5.1	SMA Desenvolvido	51
5.2	Variáveis do Sistema	53
5.2.1	Planos do Agente Driver	54
5.2.2	Planos do Agente Manager	57
5.3	Artefatos do Sistema	60
5.3.1	Leitor do Arquivo de Configuração	60
5.3.2	Checagem do Ambiente	61
5.3.3	Alteração do Preço	62
5.3.4	Criação do Log de Eventos	63
5.3.5	Controle da Fila de Espera	64
5.3.6	Simulador de Progressão do Tempo	65
5.4	Processo de Reorganização de Vagas	65
5.5	Regras de Precificação	67
5.5.1	Arquivo de Configuração	67
5.6	Processo de Precificação	69
5.6.1	Mudança Interna do Preço	71

5.6.2	Mudança Externa do Preço	71
<b>5.7</b>	<b>Exemplo do Processo Completo</b>	<b>72</b>
<b>5.8</b>	<b>Considerações Finais</b>	<b>73</b>
<b>6</b>	<b>RESULTADOS</b>	<b>74</b>
<b>6.1</b>	<b>Cenários de Testes</b>	<b>74</b>
6.1.1	Arquivos de Configuração 1 e 2	75
6.1.2	Arquivos de Configuração 3 e 4	75
6.1.3	Arquivos de Configuração 5 e 6	76
<b>6.2</b>	<b>Resultados Obtidos</b>	<b>78</b>
6.2.1	Cenários com 100 vagas e 50 <i>drivers</i>	81
6.2.2	Cenários com 100 vagas e 100 <i>drivers</i>	85
6.2.3	Cenários com 100 vagas e 200 <i>drivers</i>	90
6.2.4	Cenários com 200 vagas e 50 <i>drivers</i>	97
6.2.5	Cenários com 200 vagas e 100 <i>drivers</i>	100
6.2.6	Cenários com 200 vagas e 200 <i>drivers</i>	106
<b>6.3</b>	<b>Discussão dos Resultados</b>	<b>112</b>
6.3.1	Vantagens e Desvantagens	120
6.3.2	Melhores Casos de Uso	125
<b>7</b>	<b>CONCLUSÃO</b>	<b>128</b>
<b>7.1</b>	<b>Trabalhos Futuros</b>	<b>129</b>
<b>7.2</b>	<b>Limitações do Trabalho</b>	<b>130</b>
	<b>REFERÊNCIAS</b>	<b>132</b>

## 1 INTRODUÇÃO

Grandes centros urbanos estão diante de um processo crescente de aglomeração e desenvolvimento que, como consequência, causa o crescimento da população dessas áreas urbanas. No Brasil, Devido a impedimentos legislativos e culturais, não foram realizados os investimentos para preparar as cidades com a infraestrutura urbana e serviços públicos necessários às necessidades da população. A falta de recursos públicos e de estruturas organizacionais reduzem a capacidade de continuar o processo de planejamento estratégico e gerenciamento de projetos. É preciso, portanto, buscar implementações inovadoras para solucionar este problema (MONZONI MARIO. NICOLLETTI, 2014).

Neste sentido, a abordagem de *Smart City* (do inglês, Cidade Inteligente) representa um paradigma que propõe o desenvolvimento de projetos caracterizados pelo uso e integração de novas tecnologias para solucionar dificuldades no ambiente urbano (DI NAPOLI; DI NOCERA; ROSSI, 2014). Dentre essas está o problema de trânsito e mobilidade em cidades de grande ou médio porte. Os habitantes nestas cidades enfrentam diariamente estes problemas de tráfego que são muitas vezes causados por procura de estacionamento vago (NOBRE, 2014). Isso pode vir prejudicar a interação social, causar a poluição do meio ambiente e até mesmo afetar a economia local. No entanto, encontrar um espaço de estacionamento vazio em áreas urbanas é um problema que envolve mais do que apenas o motorista e a sua possível vaga (BARILE *et al.*, 2015).

Motoristas desejam uma vaga de estacionamento que atenda às suas necessidades. Proprietários de estacionamentos querem maximizar suas rendas econômicas com a venda do maior número possível de vagas de seus estacionamentos. Gestores urbanos, por sua vez, esperam reduzir a circulação de veículos em partes da cidade e ter uma distribuição justa de vagas entre os motoristas solicitantes, limitando o congestionamento de tráfego nas proximidades dos estacionamentos (BARILE *et al.*, 2015).

O desenvolvimento de Estacionamentos Inteligentes pode ser realizado a partir da utilização de Sistemas Multiagentes SMA. Nestes sistemas, agentes computacionais autônomos simulam as interações e ações de participantes em um estacionamento. Os agentes são implementados em um ambiente com o intuito de alcançar um objetivo

indicado a partir de execuções de ações sobre este ambiente (WOOLDRIDGE, 2009). Para o sistema multiagente desenvolvido ser capaz de empregar conceitos do raciocínio humano, é utilizado o modelo BDI (*Belief, Desire e Intention*).

Este modelo é usado como base para a implementação dos agentes que compõem o SMA (BORDINI; HÜBNER; WOOLDRIDGE, 2007). A suposição essencial do modelo BDI é que as ações destes agentes são derivadas de um processo de raciocínio prático, que é composto de duas etapas (WOOLDRIDGE, 2009). Na primeira etapa, a definição de objetivos, um conjunto de desejos é selecionado à ser alcançado, de acordo com a situação atual dos conhecimentos do agente. A segunda etapa é responsável pela determinação de como estes objetivos podem ser alcançadas por meio das opções disponíveis para o agente.

Estes objetivos são um conjunto de desejos concretos consistentes que são alcançáveis, evitando a necessidade de uma fase complexa de deliberação de metas. Um agente deve tomar decisões que maximizem a utilidade de cada ação. A utilidade de cada ação é definida a partir dos objetivos do agente. Uma ação que levará o agente a alcançar um objetivo em tempo desejado, terá uma utilidade maior que ações que não convêm a estes objetivos (WOOLDRIDGE, 2009). É importante utilizar uma linguagem capaz de desenvolver os agentes com base no modelo BDI, bem como ferramentas que sejam capazes de modelar e implementar as partes que compõem o ambiente onde o agente está inserido e também as normas que regem este ambiente, definindo-se assim o que um agente pode ou não fazer.

O *framework* JaCaMo (JACAMO, 2011) é uma ferramenta utilizada para a implementação de SMAs com base no modelo BDI. Este *framework* é composto por três módulos: uma linguagem para o desenvolvimento e execução dos agentes; um *framework* capaz de implementar o ambiente onde os agentes estão inseridos; uma ferramenta capaz de estabelecer normas no ambiente ou organização onde os agentes atuam. A linguagem de implementação dos agentes é o . Para desenvolvimento do ambiente, é utilizado o *framework* Cartago, em qual são implementados dos artefatos que os agentes interagem. E finalmente, na ferramenta Moise é estabelecido das normas que permitem ou requerem ações dos agentes na organização.

A proposta deste trabalho é realizar uma implementação (utilizando o *framework* JaCaMo) de um mecanismo de atribuição e modificação de preços de vagas em SMAs. Este trabalho é uma extensão do projeto MAPS (UTFPR) para gerenciamento de

Els. Outros participantes neste projeto já: implementaram um SMA para alocação de vagas baseado no grau de confiança dos agentes motoristas (CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, 2017); apresentaram um SMA onde os agentes negociam por vagas seguindo um modelo aberto e descentralizado (DUCHEIKO; BORGES; ALVES, G. V., 2018); ou propõem a implantação de um Smart Parking empregando agentes embarcados (BOTELHO; BORGES; ALVES, G. V., 2019). Este sistema por sua vez tem a função de gerenciar preços e a organização de vagas considerando variáveis de ambiente em um sistema de *Smart Parking*.

A verificação será realizada explorando situações em um estacionamento onde o SMA deve verificar as variáveis de ambiente — horário, evento, demanda e reserva — e alterar dinamicamente os preços e tipos das vagas durante o gerenciamento do estacionamento. A variável de horário afeta o estacionamento dependendo do período do dia. A existência de um evento ocorrendo no estacionamento (como uma estreia de filme) altera os preços das vagas. A porcentagem de ocupação do estacionamento em um período anterior altera a variável de demanda e afeta os valores e organização do próximo período do estacionamento. A reserva de uma vaga realizada previamente reduz o preço dessa vaga.

## **1.1 Objetivo**

A seguir são descritos o objetivo geral e os objetivos específicos para a realização do trabalho. Neste trabalho ambos o modelo de precificação e ambiente do estacionamento foram tratados de forma simples. O preço é alterado usando apenas porcentagens e o ambiente é observado apenas no passado próximo sem nenhum tipo de previsão.

### **1.1.1 Objetivo Geral**

Desenvolvido um mecanismo de controle dinâmico de preços de vagas a partir de mudanças no ambiente e identificar sua influência na performance de um Smart Parking em seu gerenciamento dessas vagas.

### 1.1.2 Objetivos Específicos

A seguir são detalhadas os objetivos específicos para o desenvolvimento do mecanismo proposto no objetivo geral deste trabalho.

- Desenvolvido um método de reorganização das vagas no SMA base;
- Implementado mecanismos de decisão de preços relacionado a variáveis do ambiente;
- Simulado sistema em cenários elaborados para tomada de decisão;
- Comparado o SMA modificado sem mecanismos de alteração do preço com o mesmo sistema utilizando do mecanismo;

## 1.2 Justificativa

Custos derivados da concentração de veículos sem planejamento e sem estrutura para acolher uma grande quantidade de pessoas recaem nos orçamentos públicos e privados e crescem à medida que os ecossistemas urbanos chegam ao esgotamento. A Comissão Europeia de Mobilidade e Transporte estima que os congestionamentos custam cerca de U\$S 149 bilhões anualmente (MONZONI MARIO. NICOLLETTI, 2014). Isso é importante porque segundo a União Internacional de Transportes Públicos (UITP), automóveis possuem baixa eficiência de transporte, na medida da sua baixa relação de passageiros transportados por metro quadrado ocupado. Isso causa os congestionamentos nas vias para estacionamento (NOBRE, 2014).

Considerando que o Brasil possui um problema de congestionamento em grandes cidades, avanços em tecnologias que amenizam este problema serão boas para o ambiente e para a economia. Assim, é válido mostrar que uma implementação de um SMA capaz de gerenciar e alocar de vagas consegue aprimorar o funcionamento de um estacionamento. Desta forma será possível proporcionar uma maior organização e menores gastos nesta área.

## 1.3 Trabalhos Publicados

Os artigos publicados a seguir foram desenvolvidos no mesmo contexto deste trabalho. Em (MELLADO; ALVES, G.; BORGES, 2019), foi apresentado uma comparação entre diferentes soluções de *Smart Parkings* que utilizam de agentes inteligentes.



As soluções comparadas identificam a existência de diversas maneiras de desenvolver um EI.. Em (MELLADO; ALVES, G. V. *et al.*, 2020), é apresentado o desenvolvimento de uma versão inicial deste módulo de precificação dinâmica. Essa implementação não possui a definição de diferentes tipos de vagas ou seus métodos de reorganização. As simulações realizadas neste artigo mostram indicações da influência do módulo porém os resultados apresentados não são suficientes para alcançar essa conclusão. No trabalho publicado, (MELLADO *et al.*, 2021), é apresentado o módulo de precificação finalizado e, apesar de não detalhado no texto, o SMA contendo métodos de reorganização. Os resultados neste artigo mostram melhor como a mudança do preço das vagas é realizada, afeta o comportamento do estacionamento.

#### **1.4 Organização do Trabalho**

O restante deste trabalho está organizado da seguinte maneira. No capítulo 2 são apresentadas as definições e funcionamentos de um SMA, o *framework* JaCaMo e sua plataformas. No Capítulo 3 são descritos estacionamentos inteligentes, tipos de estacionamentos e as estratégias em seus gerenciamentos incluindo métodos de precificação dinâmica. No Capítulo 4 são detalhados os trabalhos relacionados. No Capítulo 5 é mostrado todo o desenvolvimento e funcionamento do módulo de precificação dinâmica. No Capítulo 6 são apresentados e discutidos os resultados encontrados nas execuções de diversos cenários simulados. No Capítulo 7, uma conclusão sobre o resultado e proposto trabalhos é apresentada. O trabalho é finalizado com as referências utilizadas em seu desenvolvimento.

## 2 SISTEMAS MULTIAGENTES

Neste capítulo serão descritos agentes, sistemas multiagentes. Brevemente, um SMA é composto por múltiplos agentes inteligentes, que possuem comportamento autônomo e interagem com os outros agentes presentes no ambiente e com o próprio ambiente afim de alcançar seus objetivos. Na seção 2.1 serão classificados os diferentes tipos de agentes. A seção 2.2 detalha o modelo BDI.

### 2.1 Agentes Inteligentes

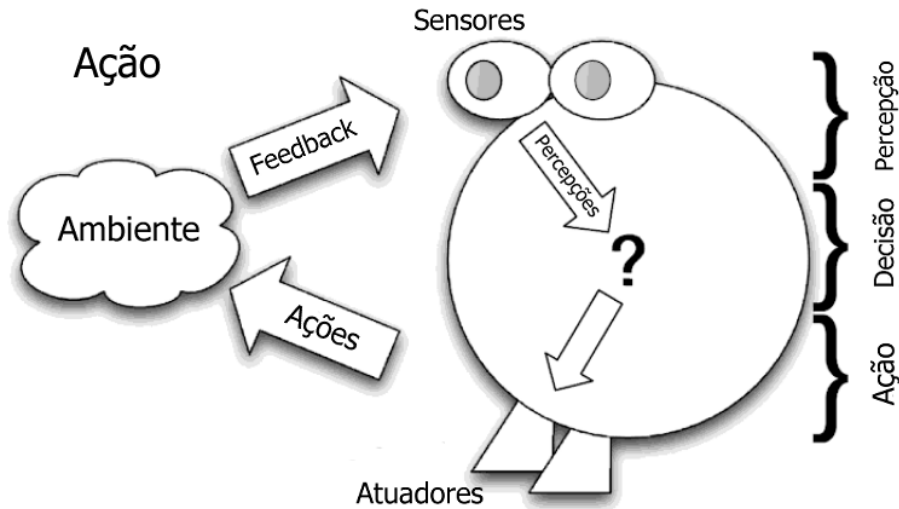
Não existe uma definição universalmente adotada do termo agente (WOOLDRIDGE, 2009). Embora haja um consenso geral de que a autonomia é central para a noção de agência, há pouca concordância além disso. Parte da dificuldade é que vários atributos associados ao agente possuem importância diferente em domínios diferentes. Para algumas aplicações a capacidade dos agentes de aprender com suas experiências são de maior importância; para outras, a aprendizagem não é apenas desimportante, é indesejável. Três definições são comumente usadas:

(WOOLDRIDGE; JENNINGS, 1995; WOOLDRIDGE, 2009) define que: "Um agente é um sistema de computador que está situado em algum ambiente, e que é capaz de ação autônoma neste ambiente para atingir seus objetivos delegados"; (FERBER, 1999) define agente como: "Uma entidade física ou virtual que pode atuar, perceber seu ambiente e se comunicar com outros, é autônomo e possui habilidades para atingir seus objetivos e tendências"; e (RUSSELL; NORVIG, 1995) define como: "Um agente é qualquer coisa que pode ser vista percebendo seu ambiente por meio de sensores e agindo sobre esse ambiente por meio de atuadores."

A figure 1 mostra os elementos que compõe um agente inteligente. Um agente possui sensores capazes de observar e receber *feedback* de um ambiente. este agente utiliza essa percepção do ambiente e toma uma decisão a fim de cumprir um objetivo previamente estabelecido. A partir dessa decisão tomada, o agente realiza uma ação sobre o ambiente utilizando seus atuadores.

Um agente possui quatro características principais: autonomia, proatividade, reatividade e interatividade social. Juntas elas trabalham para alcançar um objetivo previamente estabelecido. Devido à essas características e à capacidade de comunicação

**Figura 1 – Representação de um agente inteligente**



**Fonte: Adaptado de (WOOLDRIDGE, 2009).**

entre os agentes, estes são utilizados para resolverem problemas dinâmicos onde há interação humana no ambiente de atuação dos mesmos (WOOLDRIDGE, 2009).

### 2.1.1 Autonomia

Autonomia é a principal característica que define um agente. Outros sistemas computacionais são desenvolvidos utilizando de uma linguagem orientada a objetos destinados a seguir uma funcionalidade específica. Esses objetos são criados com um objetivo e são definidas regras que atingem este objetivo sempre da mesma maneira. Autonomia permite que um agente decida qual é a melhor abordagem para resolver a situação atual. Este agente possui a capacidade de observar o ambiente tomar uma decisão e formar as suas decisões independentemente de como irá atingir os objetivos delegados a ele. As decisões estão sob seu controle e não controlada por outros, tal como por um usuário (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

### 2.1.2 Proatividade

Proatividade é a capacidade de tomar a iniciativa em relação ao objetivo a ser atingido. O agente não reage simplesmente em resposta às ações do ambiente, mas sim focando em sua tarefa final. Por exemplo, se tenho o desejo de ficar ganhar mais dinheiro e surge uma oportunidade de fazer fortuna, então é racional tentar explorar essa oportunidade, talvez gerando um novo objetivo. (WOOLDRIDGE, 2009).

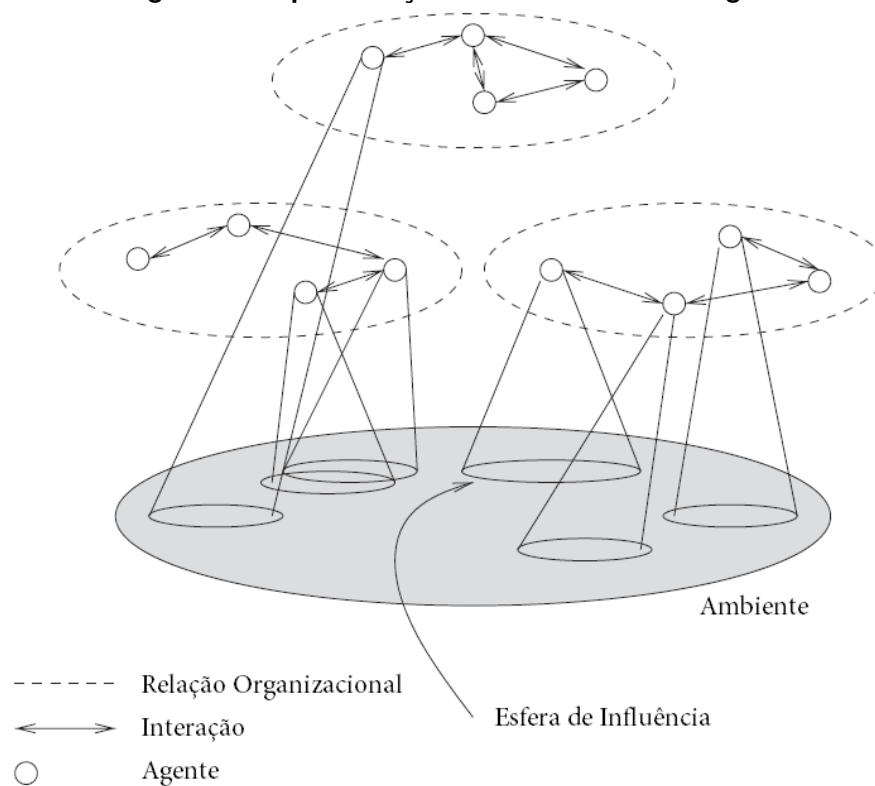
### 2.1.3 Reatividade

A característica de reatividade de um agente é a capacidade dele de interpretar o ambiente e reagir prontamente de acordo com as mudanças que ocorrem no mesmo. Essas reações podem ser reflexivas ou racionais. Reação reflexiva é simplesmente uma ação gerada por um estímulo vindo do ambiente, enquanto racionais são ações baseadas em conhecimentos adquiridos previamente (WOOLDRIDGE, 2009).

### 2.1.4 Habilidade Social

A capacidade de interação social de um agente é determinada pela comunicação com outros agentes no mesmo ambiente. Cada agente nestes sistemas possuem uma esfera de influência que é controlada total ou parcialmente por ele, conforme Figura 2. Além disso, a comunicação entre os agentes permitem que um agente tenha conhecimento de certas informações de outro agente (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

**Figura 2 – Representação de um sistema multiagente**



**Fonte: Adaptado de (BORDINI; HÜBNER; WOOLDRIDGE, 2007)**

Este processo de comunicação entre os agentes é um ponto fundamental em um SMA, visto que, diferentes agentes precisam de informações a todo momento

sobre o ambiente e de outros agentes. Essas informações, na maioria das vezes, é proveniente da troca de mensagens entre os agentes. Mensagens mais complexas que podem coordenar atividades a fim de tornar o SMA mais dinâmico e flexível a mudança (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

## 2.2 Modelo BDI

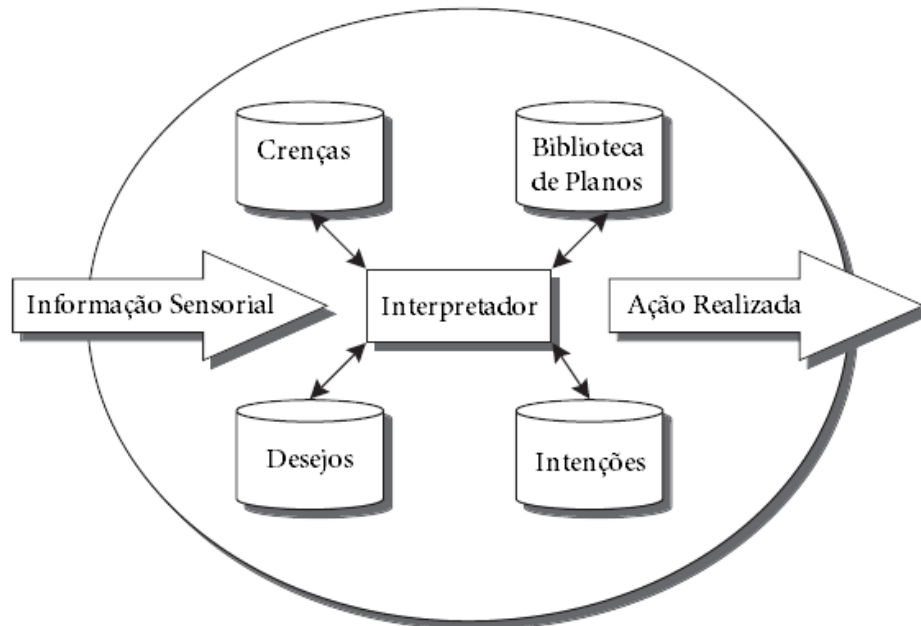
O modelo *Belief, Desire and Intention* (BDI), (Crença, Desejo e Intenção) é inspirado num comportamento humano estudado pelo campo da Filosofia. É uma arquitetura comum para agentes racionais, em que o agente possui crenças, desejos e intenções, e utiliza noções mentais a fim de representar raciocínio prático (BORDINI; HÜBNER; WOOLDRIDGE, 2007). As crenças, desejos e intenções são respectivamente entendidos como:

- Crença: São as informações que o agente possui sobre o mundo ou ambiente em que está situado. Essa informação poderá estar desatualizada ou imprecisa. Exemplo: agente motorista possui 5 créditos disponíveis.
- Desejos: São todos os possíveis estados que o agente pode querer vir realizar. Apesar disto, possuir um desejo não obriga o agente a satisfazê-lo imediatamente ou até lo realiza. Desejo são possíveis ações do agente. Um desejo pode até mesmo ser incompatível com outro desejo. De modo geral, desejos são todas as opções que o agente possui a fim de atingir um objetivo. Exemplo: agente driver deseja uma vaga do tipo normal.
- Intenções: São os desejos ou estados que o agente decidiu realizar. De modo geral, um agente analisa todos os seus desejos para atingir um objetivo. Ao escolher este desejo, este mesmo torna-se uma intenção. Um agente com um objetivo a ser cumprido, analisa os seus desejos, escolhe uma ou mais intenções a fim de atingir e completar este objetivo. Exemplo: agente driver vai requerer uma vaga normal no estacionamento.

A Figura 3 mostra o processo de raciocínio de um agente no modelo BDI. O programa de um agente executa de acordo com os mecanismos do interpretador que recebe as informações do ambiente por meio de sensores, decide a próxima ação a ser realizada e a executa usando atuadores:

1. O interpretador primeiro prepara sua base de crenças para receber os dados

**Figura 3 – Representação de um modelo BDI**



Fonte: Adaptado de (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

do ambiente. As informações sensoriais são usadas na atualização de crenças, que por sua vez influênciam na escolha do próximo evento.

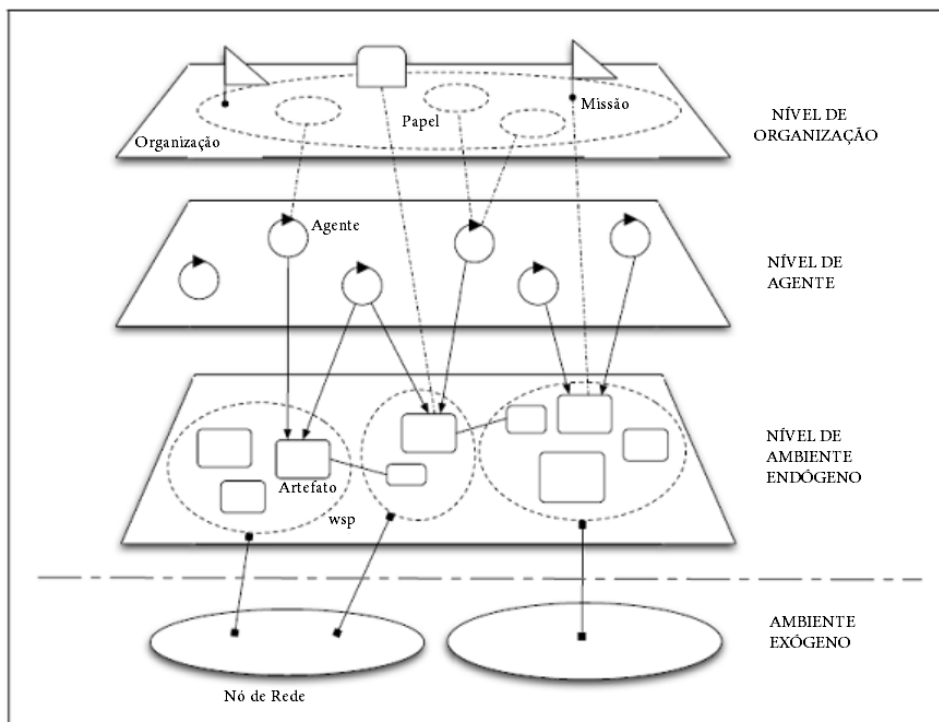
2. Em sequência são observadas a comunicação com outros agentes que pertencem ao mesmo ambiente que este. A troca de mensagens entre agentes pode resultar em novas metas ou crenças porém nem todas as mensagens são importantes para todos os agentes e o interpretador realiza um processo de filtro dessas usando uma função interna que define quais delas são "Socialmente Aceitas".
3. Podem existir vários eventos pendentes, que são caracterizados por mudanças de crenças ou de metas, porém apenas um é escolhido em cada iteração do interpretador. Um novo evento é inserido na lista quando uma crença é criada ou atualizada e seu conteúdo indica que uma nova meta é adicionada. Os eventos executam conforme uma lógica de lista *FIFO*, porém o programador possui a capacidade de indicar prioridade de cada evento.
4. Após a seleção do próximo evento, a biblioteca de planos é utilizada para encontrar a sequência de ações para alcançar suas intenções. O interpretador escolhe o primeiro plano disponível que cumpre o objetivo do evento. O agente realiza essas ações sobre o ambiente e o interpretador recomeça a próxima interação reavaliando suas crenças.

### 2.3 Framework JaCaMo

Este trabalho usa como base a primeira versão implementada do projeto MAPS (CASTRO, L. F. S. d., 2015) que foi desenvolvida utilizando o *framework* JaCaMo (JACAMO, 2011). O *framework* JaCaMo combina três plataformas para auxiliar no desenvolvimento de um Sistema Multiagente (SMA) e esses níveis podem ser vistos na Figura 4:

- JASON, linguagem usada pra a programação dos agentes;
- CArtaGO, *framework* usado pra a programação dos artefatos do ambiente;
- MOISE, modelo organizacional de normas dos agentes.

**Figura 4 – Divisão dos níveis do JaCaMo**



**Fonte: Adaptado de (JACAMO, 2011).**

Essas três plataformas que formam este *framework* possuem o seu próprio conjunto de abstrações para a programação, assim como um modelo e meta-modelo de programação. O JaCaMo em si, tem como objetivo definir as dependências, conexões, mapeamentos conceituais das sinergias entre as diferentes abstrações disponíveis nas três plataformas que compõem o *framework*.

Os agentes, os quais são programados na linguagem Jason, são inspirados no modelo BDI. Um agente é composto por um conjunto de crenças, desejos e intenções.

Em particular no Jason, desejos entende-se como *goals* ou objetivos e intenções entende-se como *plans* ou planos.

Cada instância de ambiente do CArtAgO é composta por uma ou mais entidades de *Workspace*. Cada *Workspace* é formado por um conjunto de artefatos que fornecem um conjunto de operações e propriedades observáveis que definem a interface de uso de um artefato. A entidade de *Operation* pode gerar atualizações para as propriedades observáveis e eventos observáveis específicos. A última entidade do nível de ambiente é o *Manual*, usada para representar a descrição das funcionalidades fornecidas por um artefato.

A organização criada no Moise, pode-se ser separada de três formas. Ela pode ser estrutural aonde ela é descrita pelo grupo e pelas regras das entidades. Ela pode ser funcional aonde é definida pelo esquema social, missão e objetivo. E ela pode ser normativa aonde ela é definida pela entidade *norm* que vincula as regras para as missões, restringindo o comportamento do agente.

### 2.3.1 Jason

*Java-based interpreter for an extended version of AgentSpeak*, Jason, é uma extensão da linguagem AgentSpeak(L). Foi desenvolvido utilizando a linguagem Java e desenvolvido por Jomi F. Hübner e Rafael H. Bordini e outros colaboradores (JASON, 2005). Esta linguagem implementa uma semântica operacional e disponibiliza uma plataforma de desenvolvimento para SMAs customizáveis, além de ser uma ferramenta de código aberto distribuído sob a licença GNU LGPL (GPL).

A programação do agente efetivamente determina o ciclo de raciocínio do agente. O agente está a todo momento recebendo *feedback* do ambiente por meio de suas percepções e realizando ações de forma a reagir a esses estímulos. Inicialmente, o agente possui planos pré-programados em sua base de dados, de como reagir a esses estímulos, porém, a escolha de que maneira o agente irá reagir é realizada de maneira autônoma (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

A seguir são descritos os componentes que pertencem à linguagem Jason: crenças, objetivos, planos e suas características.



### 2.3.1.1 Crenças

Assim como no *AgentSpeak*, a linguagem Jason possui uma base de crenças inicial, a qual é uma simples coleção de literais, de mesmo modo a uma programação lógica tradicional, sendo assim, a informação é representada usando predicados, como, por exemplo:

#### Algoritmo 1 – Criação de crenças em Jason

---

```

1 alto ( pedro ).
2 gosta ( pedro, musica ).

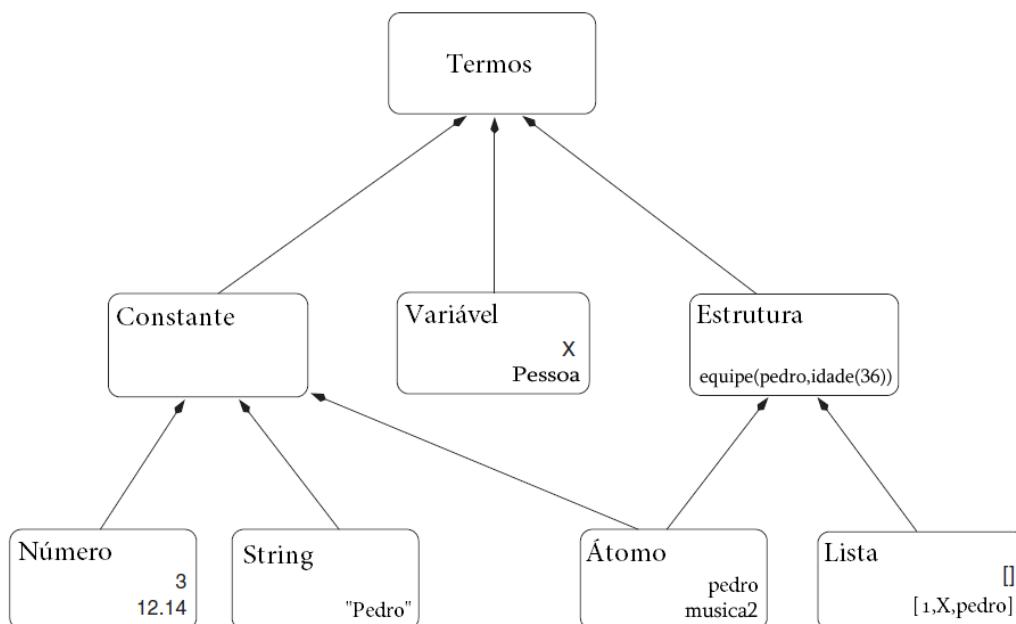
```

---

Fonte: Autoria Própria (2021).

No algoritmo 1, a linha 1 expressa uma propriedade — ser alto — de um objeto ou entidade, neste caso pedro. Na linha 2 é representado o relacionamento entre duas entidades. Neste caso: pedro gosta de música. De modo geral, estes itens representam o que o agente sabe sobre o mundo, ou ambiente que está inserido. As crenças não podem ser tomadas como verdades absolutas, tendo em vista que o agente muitas vezes recebe informações imprecisas ou inválidas provindas do ambiente. Na Figura 5 é mostrado a lógica de termos de *AgentSpeak* utilizado em Jason para representar os predicados e usados para se referir a uma constante, uma variável ou uma estrutura.

Figura 5 – Tipos de termos do AgentSpeak no Jason



Fonte: Adaptado de (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

Átomos são quaisquer símbolos ou sequência de caracteres começando com

uma letra minúscula e são usados para representar indivíduos ou objetos específicos. Por exemplo, *pedro* e *musica2* são átomos e podem representar respectivamente uma pessoa real chamada Pedro e a segunda música em alguma coleção. Além de átomos, números e strings também são classificados como constantes.

Variáveis são um símbolo sequência de caracteres começando com uma letra maiúscula e vinculadas a valores por unificação. Uma vez instanciadas a um determinado valor, elas mantêm este dado em todo o seu escopo - no caso do *AgentSpeak*, o escopo é o plano em que aparecem. Enquanto *pedro* é um átomo, Pessoa é uma variável e usando os exemplos acima, uma variável Pessoa pode ser ligada ao átomo pedro. Variáveis podem ser associadas a uma constante ou a uma estrutura.

Uma estrutura é usada para representar dados complexos, *equipe(pedro,idade(36))* por exemplo pode ser usado para representar as informações sobre um membro da equipe de uma universidade. As estruturas começam com um átomo (chamado de functor) e são seguidas por vários termos (chamados de argumentos) separados por vírgulas e colocados entre parênteses. Uma lista é um tipo especial de estrutura. Existem operações específicas para listas como, '|' que pode ser usado para separar o primeiro item de uma lista de todos os itens restantes nela. Então, ao realizar a união de uma lista [1,2,3] com [R|C], a variável R é instanciada com 1 e C com a lista [2, 3].

Devido às informações que o agente recebe vindas de um agente no ambiente ou do próprio ambiente, a linguagem Jason possui um recurso de anotações, onde é possível identificar a origem de cada informação recebida. As anotações são identificadas e delimitadas utilizando os colchetes.

Na linha 1 do algoritmo 2, a anotação é apenas uma nota mental ao agente que indica uma crença interna do próprio agente. Nas linhas 2 e 3, há a utilização da anotação "source". Esses casos indicam que o agente recebeu essa informação de um fonte externa. Na linha 2 a fonte é lucas, na linha 3 a fonte é a percepção do agente. A anotação de origem é importante porque em muitos casos dependendo da fonte de informação o agente pode desacreditar na crença ou tomar medidas de precaução ao acreditar na informação. Na linha 4, "expires" indica que essa crença deixara de ser verdade no inverno.

### Algoritmo 2 – Anotações em crenças em Jason

---

```

1 alto ( pedro )
2 alto ( pedro )[ source(lucas) ]
3 alto ( pedro )[ source(percept) ]
4 alto ( pedro )[ expires(inverno) ]

```

---

Fonte: Autoria Própria (2021).

#### 2.3.1.2 Objetivos

Objetivos são fundamentais na construção de um agente que segue o modelo BDI. Eles determinam o que o agente sabe sobre o ambiente e o que ele deve realizar sobre o mesmo.

Os objetivos na linguagem Jason definem um estado o qual o agente deve alcançar para cumprir seu objetivo final. De mesmo modo ao *AgentSpeak*, a linguagem Jason possui duas categorias: os objetivos de testes, onde se usa o operador “?”; e os objetivos de realização, onde se usa o operador “!”. Objetivos do teste são usados para recuperar informações que estão disponíveis na base de crença do agente. Um objetivo de realização é usado para indicar que o agente tem o objetivo de atingir uma situação desejada. O fato de um agente poder adotar um novo objetivo leva à execução de planos, que são ações que o agente tomará para cumprir essa meta. Assim como existe a base de crenças iniciais que o programador define, há também a criação de objetivos iniciais de um agente.

1. ?verificarCarrinho(usuario): Objetivo deste teste é verificar o conteúdo do carrinho do usuário;
2. imprimirRecibo(compra): Objetivo deste agente é imprimir o recibo da compra;

#### 2.3.1.3 Planos

Um plano em Jason é composto por três elementos: evento gatilho, contexto e corpo. O evento gatilho e o contexto são os elementos que compõem a cabeça do plano. Sendo assim, as três partes são sintaticamente separadas por  $1 : j$  e  $1 \leftarrow j$ .

#### **EventoGatilho : Contexto $\leftarrow$ Corpo**

- **EventoGatilho:** Mudanças no ambiente podem significar que existem novas oportunidades para o agente realizar ações, considerar a adoção de novos objetivos não possuíam anteriormente ou abandonar objetivos existentes.

Existem dois tipos de mudanças na atitude mental de um agente: atualização ou adição de crenças e novas metas. Mudanças nos dois tipos de atitudes criam, dentro da arquitetura do agente, os eventos sobre os quais os agentes agem. Essas alterações podem ser de dois tipos: adição e exclusão como é visto na Figura 6.

- Contexto: A escolha do plano para cumprir o objetivo do agente é realizada somente quando está prestes a começar a agir sobre este objetivo. Um agente pode ter vários planos diferentes para atingir o mesmo objetivo. O contexto de um plano é usado para verificar a situação atual, a fim de determinar se um plano específico é provável que tenha sucesso em lidar com o evento, dadas as informações mais recentes que o agente tem sobre seu ambiente.
- Corpo: O corpo de um plano é nada mais que a sequência de fórmulas que determinam um curso da ação.

**Figura 6 – Tipos de gatilhos de evento**

Notação	Nome
+ <i>l</i>	Adição de Crença
- <i>l</i>	Remoção de Crença
+! <i>l</i>	Adição de Objetivo a ser Alcançado
-! <i>l</i>	Remoção de Objetivo a ser Alcançado
+? <i>l</i>	Adição de Objetivo de Teste
-? <i>l</i>	Remoção de Objetivo de Teste

Fonte: Adaptado de (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

A algoritmo 3 apresenta um exemplo de plano de uma agente usando a ação interna *send* para iniciar outro plano em um agente diferente.

**Algoritmo 3 – Plano interno em jason**

```

1 +!askSpot : myTrust(MT) and myType(MTy) <-
2   .send(manager,achieve,askSpot(MT,MTy)).

```

Fonte: Autoria Própria (2021).

Quando o evento gatilho *askSpot* é acionado, este agente inicia este plano de requisição de vaga que utiliza de duas de suas crenças: *myTrust*, que contém o seu grau de confiança e indica a rapidez que será atendido; e *myType*, que contém qual tipo de vaga este agente está requisitando. O corpo deste plano possui apenas a ação interna *.send* que envia uma mensagem para o agente gerente (*manager*) contendo os valores de suas duas crenças e com o objetivo de adicionar uma nova meta no gerente.

### 2.3.2 Cartago

*Common ARTifact infrastructure for AGents Open environments*, CArtAgO , é um framework que possibilita desenvolver e executar ambientes virtuais para SMAs (CARTAGO, 2006). CArtAgO é baseado no meta-modelo de agentes e artefatos para modelar e implementar sistemas multiagentes. Esses artefatos são usados como recursos e ferramentas, e são construídos dinamicamente para serem usados ou manipulados por agentes a fim de realizar suas atividades individuais e coletivas.

CArtAgO possibilita o desenvolvimento e a execução de ambientes baseados em artefatos, estruturados em espaços de trabalho abertos que agentes de diferentes plataformas podem unir para trabalhar juntos nesses ambientes.

#### 2.3.2.1 Workspaces

Um ambiente é composto por um ou mais *workspaces*, possivelmente distribuídos em uma rede. Agentes implementados em Jason podem participar de um ou mais *workspaces* simultaneamente, sendo inicialmente designado a um *workspace* padrão. O agente deve obrigatoriamente pertencer a no mínimo um *workspace* para usufruir de um ambiente. Logo um *workspace* está sempre inserido em um ambiente.

#### 2.3.2.2 Repertório de Ações e Artefatos

É o conjunto de ações definidas pelos artefatos do ambiente, existindo um mapeamento um-para-um entre ações e operações. Isso implica que se uma operação tem sucesso, sua ação também possuirá um sucesso. O mesmo ocorre em uma falha. Um artefato é definido como sendo uma entidade não-autônoma dentro de um ambiente pois essa entidade é invocada apenas por agentes.

#### 2.3.2.3 Integração Jason-Cartago

A linguagens Jason, CArtAgO e Moise podem ser utilizadas independentemente ou realizar a troca de dados entre si de forma manual. O uso, portanto, do *framework* JaCaMo está em sua integração destas em um único ambiente diminuindo a complexidade da troca de informações entre elas. A conexão Jason-Cartago, especialmente, facilita a comunicação entre agentes e artefatos sem a necessidade de importar e exportar parâmetros entre elas.

Em JaCaMO, é possível que um agente inicialize um artefato e uso de suas operações dentro de seu código. O agente executa a função e o JaCaMO é encarregado de iniciar o algoritmo no código do artefato. Para um agente utilizar de um artefato são necessárias a criação de sua instancia e foco do agente no artefato. O algoritmo 4 apresenta o uso da ação interna *makeArtifact*, provida do *framework* JaCaMo e que o agente utiliza para ter acesso as operações do artefato.

---

**Algoritmo 4 – Inicialização de um artefato**

---

```
1 makeArtifact("Price","mapsEngine.PriceEngine", [PB], ArtId);
2 focus(ArtId);
```

---

**Fonte: Aatoria Própria (2021).**

- **Nome do artefato:** "Price";
- **Localização do artefato:** "mapsEngine.PriceEngine": Nome do sistema.Nome do pacote;
- **Parâmetro de inicialização:** PB : Preço Base;
- **Identificador do Artefato:** ArtId : Usado na referência do artefato.

### 2.3.3 Moise

Moise é um modelo organizacional para SMAs baseado em noções como papéis, grupos e missões. Ele permite que um SMA tenha um a especificação explícita de sua organização. Esta especificação deve ser usada tanto pelos agentes para argumentar sobre sua organização e por uma plataforma de organização que impõe que os agentes sigam a especificação (MOISE, 2006).

### 2.3.4 Considerações Finais

Nesta seção foi descrito JaCaMo, um *framework* que integra três linguagens e facilita o desenvolvimento do SMA para este projeto. JASON, a linguagem da camada de agentes, proporciona a criação dos usuários e gerentes do sistema. Suas crenças, objetivos e planos perpetuam as mudanças de estado e funcionamento do SMA. CArTAgO, a linguagem da camada do ambiente, simplifica a conversa entre o agentes e componentes do cenário em que encontram-se. Moise, a linguagem da camada de organização, permite a especificação das tarefas dos agentes do SMA. A comunicação direta entre os agentes do sistema em JASON e operações do ambiente em CArTAgO propicia a implantação do módulo de precificação dinâmica.

### 3 ESTACIONAMENTOS INTELIGENTES

Um *Smart Parking* é um sistema que ajuda motoristas a encontrar vagas de estacionamento satisfatórias, de maneira eficiente, por meio de sensores que detectam os espaços vazios de um estacionamento e direcionam os motoristas para estes (CHINRUNGRUENG; SUNANTACHAIKUL; TRIAMLUMLERD, 2007).

Os primeiros sistemas foram utilizados como meio de reunir informações sobre o estacionamento, como o status de disponibilidade e número de espaços disponíveis. Este tipo de sistema ainda é amplamente utilizado e é uma parte necessária de novos estacionamentos.

Um sistema mais complexo incorpora tecnologias mais avançadas para atender clientes de diferentes necessidades. Estacionamento em grandes complexos comerciais podem direcionar os clientes para espaços disponíveis perto de onde querem fazer compras. Ao mesmo tempo, algumas garagens podem empregar medidas de segurança, como a coleta de licenças e a verificação no *check-out*.

Na seção 3.1 são apresentados os benefícios do uso de estacionamentos inteligentes. A seguir na seção 3.2, são descritas as funções e utilidades destes estacionamentos. Na seção 3.3 é descrito o processo de gerenciamento de um estacionamento, incluindo os tipos e estratégias destes. Depois, na seção 3.4, é apresentado o conceito de precificação dinâmica e suas características. Este capítulo é finalizado com uma seção de considerações finais.

#### 3.1 Benefícios de Estacionamentos Inteligentes

A tecnologia de Smart Parking beneficia o cliente e o operador de estacionamento das seguintes maneiras: (CHINRUNGRUENG; SUNANTACHAIKUL; TRIAMLUMLERD, 2007)

- O cliente possui prontamente a disponibilidade de vagas antes ou depois de entrar no estacionamento;
- O cliente pode planejar seu percurso para um transporte público que utiliza sistemas de EI;
- O operador de estacionamento pode usar os dados do sistema para: desenvolver ou melhorar as estratégias de preços, prever futuros padrões e

tendências de estacionamento, e evitar roubos de veículo;

- O operador de estacionamento pode reduzir as necessidades de pessoal para controle de tráfego dentro da instalação;
- O sistema reduz significativamente o tráfego e emissões do veículo, diminuindo o tempo necessário para que os clientes localizem vagas disponíveis;

### **3.2 Funções de estacionamentos inteligentes**

As funções de Smart Parking são divididas em dois tipos de fluxos: tráfego e informação (SHIN; JUN, 2014).

O fluxo de tráfego acontece no caminho para encontrar estacionamento. Motoristas de veículos recebem informações de disponibilidade de vagas, dirigem para essas áreas desejadas e depois estacionam. Quando há muitos motoristas procurando por uma vaga, ocorre uma competição e resulta em conflitos cumulativos. Os comportamentos de estacionamento também variam considerando as fontes de informação e quanto tempo eles estão cruzando na rua. Uma vez que um veículo chega ou sai de uma vaga, as informações de disponibilidade mudam e são anunciadas para outros motoristas que procuram espaço para estacionar.

O fluxo de informações se dá pelos dados do estacionamento desde o momento em que veículos são detectados pelos sensores até o momento em que informações do estacionamento aparecem para o motorista. Para obter o status de disponibilidade do estacionamento, sensores são instalados para detectar entrada e saída de veículos. Esses sensores formam uma rede e enviam as informações mais recentes para os dispositivos de armazenamento de dados. Os motoristas podem obter as informações mais recentes a partir de mensagens ou de seus dispositivos inteligentes que trocam mensagens com o sistema.

### **3.3 Gerenciamento de Estacionamento**

A gestão do estacionamento é regida pelas políticas e programas que resultam em um uso mais eficiente dos recursos de estacionamento. O gerenciamento de estacionamento inclui diversas estratégias específicas. Quando aplicado adequadamente pode reduzir significativamente o número de vagas de estacionamento necessárias em uma situação particular, proporcionando uma variedade de benefícios econômicos, sociais e



ambientais. Quando todos os impactos são considerados, o gerenciamento aprimorado geralmente é a melhor solução para problemas de estacionamento (LITMAN, 2019).

Um automóvel típico fica estacionado 23 horas por dia e usa vários lugares de estacionamento por semana (LITMAN, 2019). O estacionamento é um componente essencial de um sistema de transporte. As soluções de gestão atendem aos objetivos de planejamento estratégico:

- Custos de desenvolvimento reduzidos e maior acessibilidade.
- Planejamento comunitário mais compacto e multimodal (crescimento inteligente).
- Incentivar o uso de modos alternativos e reduzir o uso de veículos motorizados (reduzindo assim o congestionamento do tráfego, acidentes e poluição).
- Melhoria nas opções do usuário e na qualidade do serviço, principalmente para não-motoristas.
- Melhoria na flexibilidade de design, criando comunidades mais funcionais e atraentes. Capacidade de acomodar novos usos e responder a novas demandas.
- Reduzir os benefícios ambientais e estéticos relacionados à superfície e à água.

O gerenciamento de estacionamento pode melhorar as opções do usuário. Isso inclui: opções de viagem (caminhada, ciclismo, compartilhamento de carro, transporte público, compartilhamento de carro); opções de estacionamento (opções que permitem que os motoristas escolham entre espaços mais convenientes, mas mais caros e horários mais baratos); taxas diárias ou mensais; e opções de pagamento (moedas, cartões de crédito, pagamentos online, etc...).

### 3.3.1 Tipos de Estacionamentos

Um estacionamento pode conter diferentes características dependendo do ambiente em que ele se encontra. A partir dessas características são definidos (LITMAN, 2019) diversos tipos de estacionamentos:

- Estacionamento na rua: Lugares de estacionamento designados localizados geralmente ao lado da calçada. Custos de construção moderados e alta densidade (relativamente pouca terra usada por espaço), porque não necessitam

de entrada de automóveis. Conveniente de usar e pode servir vários destinos. Os estacionamentos na rua devem ser gerenciados para máxima eficiência.

- Estacionamento de Superfície: Um lote de estacionamento diretamente no chão (pavimentado ou não pavimentado). Baixos custos de construção. Ineficientes se servem apenas para um único destino. Devem ser minimizados e gerenciados por eficiência.
- Estacionamento estruturado ou subterrâneo: Qualquer estrutura de estacionamento de vários andares, incluindo estacionamentos dentro ou sob um prédio. Custos elevados de construção, mas custos de terra relativamente baixos e altas densidades. Suporta o desenvolvimento de compactos, mas deve ser administrado de forma eficiente para justificar seus altos custos de construção.
- Estacionamento com preço: Qualquer estacionamento onde os motoristas são cobrados diretamente para uso, incluindo estacionamento com taxímetro na rua e lotes fora da rua onde os motoristas pagam por hora, dia, semana, mês ou ano. Pode ser aplicado a qualquer tipo de estrutura de estacionamento. Os preços tendem a incentivar o uso eficiente dos estacionamentos.
- Estacionamento comercial: Um estacionamento com fins lucrativos está disponível para qualquer motorista e serve vários destinos. Pode ser aplicado a qualquer tipo de estrutura de estacionamento. Tende a ser eficiente porque tem preço e geralmente serve vários destinos.

### 3.3.2 Estratégias de estacionamentos

O processo de tomada de decisão para encontrar uma vaga de estacionamento para um cliente não é muito variado. Dentre os projetos pesquisados, poucos consideram algo diferente de distância até o destino e preço do vaga. Uma exceção ocorre em (CASTRO, L. F. S. d., 2015) onde é realizado a alocação de vaga considerando um atributo de confiança em cada motorista.

#### 3.3.2.1 Grau de Confiança

No trabalho de (CASTRO, L. F. S. d., 2015) a estratégia está em cada agente de motorista possuir um valor de grau de confiança. Agentes com maior grau de confiança

são atribuídos vagas mais rapidamente do que aqueles com um grau de confiança inferior. Essa confiança pode ser explicada como um valor que mostra o quão confiável um agente é para cumprir uma meta específica ou executar uma tarefa. Este valor de confiança representa o compromisso do agente com o *Smart Parking*, de forma ele será recompensado desde que siga as regras do estabelecimento.

### 3.3.2.2 Distância até o Destino

O foco da estratégia no modelo PARKAGENT (BENENSON; MARTENS; BIRFIR, 2008) está em encontrar um lugar de estacionamento o mais próximo do destino o mais rápido possível. Para isso, a decisão do agente motorista é baseada na comparação da distância até o destino a partir do cruzamento atual e de todos os cruzamentos seguintes.

Em PARKAGENT é desenvolvido um modelo baseado em agente espacialmente explícito, como visto na Figura 7. É utilizado um GIS (*Geographic Information System*) urbano com camadas que representam cada elemento da infraestrutura de tráfego. Nelas os agentes representam os motorista que dirige até o destino, procuram por vaga, estacionam e deixam a vaga.

**Figura 7 – Camadas básicas do modelo PARKAGENT**



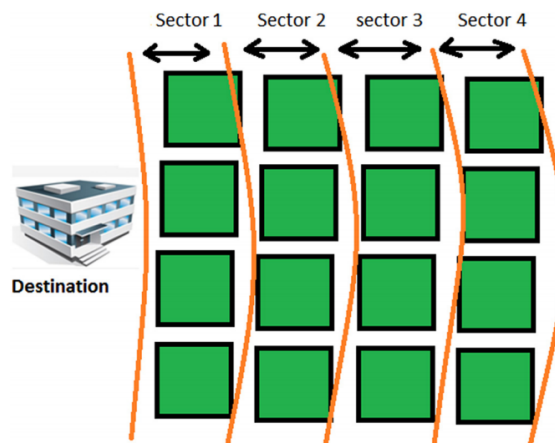
Fonte: (BENENSON; MARTENS; BIRFIR, 2008).

O comportamento e decisões dos agentes são controlados por regras que consideram à estimativa do motorista da distância até o seu destino final. Graficamente o modelo consiste em camadas com os seguintes componentes: ruas, contendo sinalização e estacionamento em cada segmento; E edifícios, definido por tipo de uso, capacidade vagas e utilização como destino.

A estratégia desenvolvida em (ALIEDANI; LOKE, 2018) tem o foco em utilizar uma organização de agentes descentralizada como meio de encontrar a vagas com menor distância de caminhada entre ela e o prédio de destino. Cada agente utiliza de comunicação sem fio com outros agentes próximos a fim de compartilhar informações e ajudar uns aos outros a encontrar uma vaga, o mais próximo possível do destino no menor tempo possível.

Ao entrar no estacionamento, o agente motorista recebe uma lista de vagas do estacionamento. Essas vagas são separadas em partições de acordo com suas distâncias até a entrada do edifício. A Figura 8 exemplifica a segmentação das vagas do estacionamento.

**Figura 8 – Segmentação da área do estacionamento**



Fonte: (ALIEDANI; LOKE, 2018).

O agente escolhe uma vaga na partição mais próxima do edifício e anuncia para os veículos próximos seu ID, posição e vaga escolhida. Enquanto o agente movimentar-se em direção a sua vaga desejada, mensagens de outros agentes são recebidas e a situação do estacionamento é atualizada. Se uma dessas mensagens corresponde a sua vaga desejada, o agente verifica qual dos dois agentes está mais próximo da vaga. O agente mais próximo continua em direção a vaga enquanto o mais longe verifica se existe outra vagas na mesma área. Caso não existam mais espaços livres na partição escolhida, o agente ordena novamente as áreas e escolhe uma vaga naquela com a seguinte menor distância até o edifício destino.

### 3.3.2.3 Preço da Vaga

Em (PIERCE; SHOUP, 2013) foi desenvolvido um programa de precificação dinâmica com a estratégia de influenciar a ocupação com a alteração do preço. A ideia é aumentar o preço das vagas de acordo com a demanda de um bloco superlotado e subvalorizado. Neste trabalho foi visto que essa estratégia gerou a liberação de uma ou duas vagas nestes blocos. Com mais vagas abertas os motoristas perdem menos tempo em trânsito procurando vaga. Com menor circulação de carros, ocorre menos congestionamento. Os preços mais altos das vagas resultam em maior rotatividade dessas causando assim mais carros utilizarem essas vagas no dia.

### 3.3.2.4 Preço e Distância

O estudo realizado por (SHIN; JUN, 2014) propõem um algoritmo que guia motoristas a um estacionamento adequado. A estratégia desenvolvida neste estudo considera o status em tempo real dos estacionamentos de uma cidade. São incluídos fatores como o custo esperado da vaga, a distância do veículo até o estacionamento, o congestionamento do tráfego neste percurso e a distância a pé da vaga até o destino do motorista. O objetivo deste algoritmo é atribuir cada carro a um estacionamento adequado de forma eficiente utilizando uma regra de despacho. Essa regra procurar encontrar o melhor estacionamento para o carro solicitante após considerar características dos estacionamentos usando funções de utilidade propostas.

## 3.4 **Precificação Dinâmica**

A precificação dinâmica é processo de determinação de preços de venda de produtos ou serviços, onde esses valores podem ser ajustados com facilidade e frequência de acordo com mudanças do ambiente. Com tecnologias atuais tornou-se possível o ajuste contínuo do preço de produtos sem um aumento do custo e esforço desta operação. este processo se aplica, por exemplo, a venda de produtos em lojas digitais na Internet ou em lojas físicas com preços digitais (BOER, 2015).

A utilização de um processo de precificação dinâmica permite a alteração de preço considerando uma grande variedade de dados sobre a venda do produto no ambiente. Esses dados permitem o aprimoramento das estratégias de alteração de preço, ao fornecer informações cruciais sobre o comportamento do consumidor em

relação aos preços dos produtos. Técnicas de precificação dinâmica são utilizadas atualmente devido suas capacidades de aprendizado em um ambiente incerto a partir do acúmulo de dados de vendas.

Precificação dinâmica pode ser considerada como a aplicação combinada de dois campos de pesquisa: Estatística, especificamente em estimar funções de demanda, e otimização de preços. A literatura classifica modelos de preços dinâmicos das formas seguintes (BOER, 2015).

#### 3.4.1 Função de Demanda Dinâmica

A função de demanda muda dinamicamente ao longo do tempo. Nesta classificação, a função de demanda pode depender da derivada temporal do preço, da quantidade de vendas cumulativas ou do histórico de preços da empresa, onde:

- Dependência temporal do preço: Modelada usando a informação que compradores não consideram apenas o preço de venda atual em sua decisão de comprar um produto, mas também as mudanças de preço previstas. O objetivo é calcular uma função de preço, em um intervalo de tempo contínuo, que maximize o lucro.
- Dependência no Histórico do Preço: Neste modelo são consideradas as percepções do consumidores sobre um preço cobrado no passado como um preço de referência. Diferenças entre o preço de venda e este preço de referência influenciam a demanda e, como consequência disso, a demanda atual e demandas futuras são afetadas pelos preços de vendas publicados.

Nestes modelos são considerados a quantidade de vendas acumuladas juntamente com o preço de venda a fim de obter a demanda por produtos. Dessa forma é possível modelar vários fenômenos relacionados à saturação do mercado, marketing, propaganda boca a boca e difusão do produto.

#### 3.4.2 Função de Demanda Estática

Devido a uma função de demanda estática, o dinamismo da precificação é causado pelo nível de estoque do modelo, mais concretamente por mudanças no valor marginal do estoque.

- Determinação de preço em estoque finito: Neste modelo é suposto a existência de um certo número de produtos a disposição. Esses serão vendidos em um período de tempo finito e sem reposição. O estoque restante ao fim do tempo é perdido e não poderá ser transferido para outro período de vendas. Para essa situação, a função de demanda não é a responsável pela mudança dinâmica dos preços. Neste caso é o fato de que o valor do estoque estará mudando ao decorrer do tempo que causa essa variação. Isso implica que o preço de venda ideal depende da quantidade de tempo de vendas restante e do estoque remanescente.
- Determinação de preço junto com aquisição: este modelo considera que o gerenciamento de preços e o estoque dos produtos pode ser realizado simultaneamente. Isso ocorre em ambientes em que a capacidade de produtos inicial é variável e é possível considerar a quantidade de itens que serão produzidos ou comprados.

### 3.4.3 Demanda Dinâmica e Aprendizado

Nesta extensão de precificação em demanda dinâmica, é considerado que uma empresa que vende um número finito de produtos em um único período de tempo deve definir de forma otimizada seu nível de produção e preço de venda. As decisões sobre este produto são influenciadas pela incerteza da demanda e em relação ao risco (neutro ao risco, avesso ao risco ou preferencial ao risco). O objetivo de precificação e aprendizagem dinâmica é desenvolver políticas de precificação que levem em consideração essa incerteza intrínseca sobre a relação entre preço e demanda esperada (BOER, 2015).

#### 3.4.3.1 Inventário Infinito

O processo de precificação é realizado em situações com estoque infinito e função de demanda desconhecida. A dinâmica dos preços é causada somente pelo fato de que a relação preço-demanda é aprendida por meio do acúmulo de dados de vendas.

Como a função de demanda depende de parâmetros desconhecidos, é utilizado inferência Bayesiana como forma de aprendizado. Inferência Bayesiana consiste na

avaliação de hipóteses a fim de alcançar informações verossímeis sobre dados desconhecidos no cenário (BOER, 2015). Neste caso o objetivo é minimizar o excesso de demanda. Em teoria, o ótimo Bayesiano pode ser calculado por meio de programação dinâmica, mas na maioria dos casos não existe uma expressão analítica de forma fechada da solução. Dessa forma são propostos dois métodos de aproximação. No primeiro, em cada momento de decisão é escolhido o preço que seria ótimo se as estimativas atuais estivessem corretas. No segundo, em cada momento de decisão, o preço é escolhido assumindo que ele será mantido durante o restante do período de venda.

#### 3.4.3.2 Inventário Finito

A precificação ocorre em produtos com quantidade fixa e estoque finito, com função de demanda desconhecida. A informação sobre o estoque já é dinâmica por si só, e o aspecto de aprendizado fornece uma fonte adicional da dinâmica de preços.

Em casos onde a taxa de chegada é desconhecida, porém sabe-se a disposição de pagar dos compradores, usa-se um modelo de venda de itens durante dois períodos. No primeiro período, vários clientes visitam a loja; se nenhum deles comprar o item, a sua crença é alterada para um valor anterior do produto, atualiza o preço de venda e tenta vender o item no segundo período. Nesta ocasião, é antecipado o comportamento estratégico dos clientes que podem adiar sua decisão de compra se anteciparem uma redução de preço. A taxa de chegada desconhecida é aprendida por meio de atualizações Bayesianas de uma distribuição gama. A política ótima de preços em tempo contínuo é caracterizada por meio de uma equação diferencial. Nos casos onde ambas a taxa de chegada e disposição de pagamento são desconhecidas e assumindo que a distribuição da demanda é um elemento de um conjunto finito conhecido, pode-se considerar uma configuração de tempo discreto com aprendizado Bayesiano e uma taxa de chegada em distribuição gama.

#### 3.4.3.3 Aprendizado de Máquina

Esta subseção não tem foco em realizar uma análise matemática do desempenho das políticas de preços, e sim em projetar um modelo realista para mercados eletrônicos e, subsequentemente, aplicar técnicas de aprendizado de máquina.



Uma vantagem dessa abordagem é a possibilidade de modelar fenômenos que influenciam a demanda, como concorrência, demanda flutuante e comportamento estratégico do comprador. Uma desvantagem é a complexidade destes modelos que dificulta sua análise analítica sobre seus comportamentos e estratégias de mudança de preços. O processo de aprendizado de máquina necessita de informações para o treinamento dos modelos desenvolvidos. Dados como transações de venda, descrição ou detalhes de produtos, promoções anteriores, preços da concorrência, dados de estoque e informações sobre localização física podem ser reunidos para o processo de aprendizado.

### 3.5 Considerações Finais

Este capítulo apresenta a tecnologia de estacionamentos inteligentes que abre a oportunidade de um melhor gerenciamento das funções de um estacionamento e permite o processo de determinação e alteração do preço conforme características dinâmicas. Esta tecnologia permite que futuros clientes possuam facilmente a disponibilidade de vagas e comunicação com o estacionamento. Para os proprietários, os dados de uso propiciam a criação ou aperfeiçoamento de estratégias de administração das vagas. O SMA desenvolvido neste trabalho é uma tecnologia de *Smart Parkings* e o módulo de precificação dinâmica, um componente deste sistema.

O gerenciamento de diferentes tipos de estacionamentos é uma área importante para criação de *Smart Parkings* e SMAs, principalmente para este trabalho. Entre os apresentados nesta subseção, o método mais comum é a consideração da posição geográfica da vaga nas decisões do sistema. A tabela 2 apresenta uma comparação entre as estratégias apresentadas previamente. O processo de mudança de preço considera as dependências temporais e históricas do preço, assim como por variações do estoque e disponibilidade do recurso ou produto em questão. O módulo de precificação desenvolvido contém dinamismo ao considerar a quantidade de pedidos por vagas, horário de uso das vagas e a condição do clima no estacionamento. O módulo, porém, não possui aprendizado sobre os dados produzidos em operações passadas.

O *Smart Parking* desenvolvido para este trabalho utilizada de uma função de demanda dinâmica com aprendizado e inventário finito. A demanda em questão muda ao decorrer do uso das vagas que possuem total máximo determinado no início da

simulação.

**Quadro 1 – Comparação entre estratégias de estacionamento**

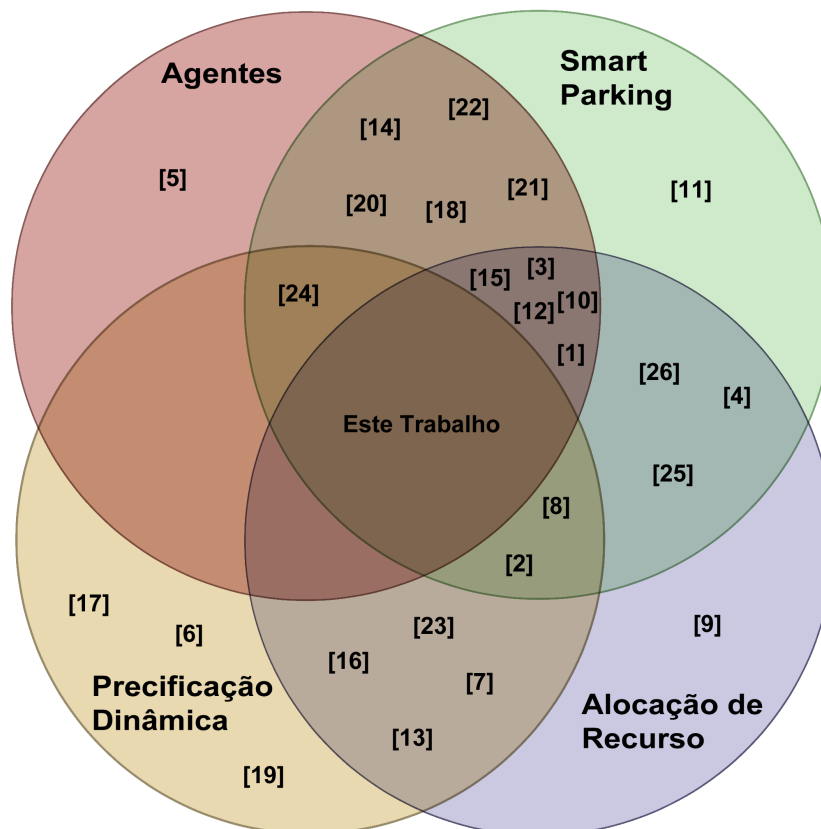
Autor	Metodologia
(CASTRO, L. F. S. d., 2015)	Grau de confiança de cada agente influencia a velocidade de atendimento no estacionamento.
(BENENSON; MARTENS; BIR-FIR, 2008)	Regras pré-estabelecidas determinam a reação de agentes em relação a falta de vagas, diferença de preços e comportamento de outros agentes.
(SHIN; JUN, 2014)	A orientação do modelo considera o tempo de viagem até o estacionamento, distância do estacionamento até o destino, custo da vaga, probabilidade de disponibilidade de vagas e congestionamento de tráfego até o estacionamento.
(PIERCE; SHOUP, 2013)	O preço muda de acordo com o horário do dia, o local da vaga e outros fatores.
(ALIEDANI; LOKE, 2018)	Compartilhamento de informações entre agentes de modo a encontrar a vaga disponível mais próxima de seu destino utilizando DSRC.

**Fonte: Autoria Própria (2021).**

## 4 TRABALHOS RELACIONADOS

Neste capítulo serão apresentados trabalhos que possuem conteúdo relevante. Os trabalhos seguintes foram divididos por quatro focos: *Smart Parking*, agentes, alocação de recurso e precificação dinâmica. A Figura 9 mostra um diagrama de Venn que representa o agrupamento dos trabalhos relacionados estudados.

Figura 9 – Agrupamento dos trabalhos relacionados



[1] - (BENESON; MARTENS; BIRFIR, 2008)

[2] - (PIERCE; SHOUP, 2013)

[3] - (NAPOLI; NOCERA; ROSSI, 2014)

[4] - (SHIN; JUN, 2014)

[5] - (ASADZADEH, 2015)

[6] - (CHEN; SHELDON, 2015)

[7] - (PIERCE; WILLSON; SHOUP, 2015)

[8] - (SFPARK, 2015)

[9] - (AMINI; MOGHADDAM; KARABASOGLU, 2017)

[10] - (CASTRO; ALVES; BORGES, 2017)

[11] - (LIN; RIVANO; MOUËL, 2017)

[12] - (ALIEDANI; LOKE, 2018)

[13] - (CAO et al., 2018)

[14] - (DUCHEIKO; BORGES; ALVES, 2018)

[15] - (RIZVI; ZEHRA; OLARIU, 2018)

[16] - (TIAN et al., 2018)

[17] - (WILLIAMS, 2018)

[18] - (BOTELHO; BORGES; ALVES, 2019)

[19] - (LI et al., 2019)

[20] - (SAKURADA; BARBOSA; LEITÃO, 2019)

[21] - (BOTELHO; ALVES; BORGES, 2020)

[22] - (CASTRO et al., 2020)

[23] - (LIAO et al., 2021)

[24] - (NAPOLI; ROSSI, 2021)

[25] - (PARKWHIZ, 2021)

[26] - (SAID; KAMAL; AFIFI, 2021)

Fonte: Autoria Própria (2021).

## 4.1 Smart Parking

Os trabalhos neste foco apresentam como característica principal o desenvolvimento ou pesquisa de tecnologias de gerenciamento dos recursos de um estacionamento.

(BENENSON; MARTENS; BIRFIR, 2008) apresentam o PARKAGENT, um modelo espacial que constrói em SIG (Sistema de Informação Geográfica) um modelo de ambiente urbano com camadas que representam cada elemento da infraestrutura de tráfego pertinente ao processo de estacionamento. O modelo também é baseado em agentes que funcionam como os motoristas que dirigem até o destino, procuram por vaga, estacionam e deixam o local quando suas atividades terminam. O modelo contém regras que descrevem o comportamento destes agentes *drivers* (motoristas) e incluem uma descrição da reação de cada agente *driver* à falta de estacionamentos, diferenças de preços e comportamentos de outros agentes. As ações de direção, busca de estacionamento, estacionar e saída do agente, também estão contidas nestas regras.

(RIZVI; ZEHRA; OLARIU, 2018) utilizam um agente de software em nuvem que determina a vaga de estacionamento mais apropriada para um motorista em uma rede de estacionamentos. Cada veículo no sistema contém um agente local com as preferências de seu motorista. Um agente *parker* na nuvem gerencia a alocação de vagas de cada agente local. Este agente é capaz de interagir com serviços de tráfego, clima e emergência relacionados para encontrar a vaga que confere com as preferências do agente local: categoria da vaga, o preço máximo aceitável do estacionamento e tempo tolerado de caminhada dessa vaga até o destino.

O sistema, proposto em (SHIN; JUN, 2014), destina o carro à uma vaga usando uma regra de despacho baseada na avaliação dos valores de várias funções de utilidade de estacionamento. A atribuição do carro solicitante à um determinado estacionamento deve ser feita considerando as características dinâmicas do estacionamento, a condição do tráfego, a conveniência do usuário e assim por diante. São considerados vários fatores dinâmicos para a alocação do veículo: tempo de viagem do local atual para o estacionamento, distância do estacionamento até o destino, custo do estacionamento, probabilidade de disponibilidade de vagas e congestionamento de tráfego até o estacionamento. O uso do algoritmo mostra que o tempo de viagem, número que não receberam vaga e utilização dos recursos da cidade podem ser melhorados.

A pesquisa (LIN; RIVANO; LE MOUËL, 2017) mostra que os trabalhos existentes e em andamento sobre *Smart Parking* são bastante complicados e transdisciplinares. Neste trabalho é apresentado sistemas de *Smart Parking* com perspectivas acadêmicas, comerciais e municipais. Este trabalho introduz um ecossistema de Smart Parking e propõem uma classificação abrangente, identificando funcionalidades e focos problemáticos nestes estacionamentos. Os sistemas estudados foram então categorizados e tabelados de acordo com três macro-temas: coleta de informações, implantação de sistemas e disseminação de serviços. As principais metodologias utilizadas nos trabalhos são sintetizadas e são resumimos seus objetivos e visões comuns que resolvem atuais dificuldades de estacionamentos.

(SAID; KAMAL; AFIFI, 2021) apresenta uma proposta moderna de Smart Parking com foco em sustentabilidade. A alocação de vagas do sistema, baseado em Internet das Coisas, foi modelada pela proposição do modelo matemático da teoria dos jogos. O sistema desenvolvido considera o preço das vagas, a distância entre a vaga e o destino do motorista e duração de estadia no estacionamento em seu processo de alocação. Os resultados apresentados indicam redução no tempo de encontrar uma vaga vazia, incentivo financeiro a indivíduos e proprietários com a aderência ao sistema e maior sustentabilidade do estacionamento.

## 4.2 Agentes

Os trabalhos a seguir focam especificamente nas funcionalidades de um agente inteligente para suas soluções.

(ASADZADEH, 2015) propõem uma solução ao problema de agendamento de *job shop* (Escalonamento de processos) utilizando agentes com Algoritmo Genético. Um SMA contendo vários agentes, com comportamentos especiais, foi desenvolvido para implementar um algoritmo genético de busca local. Essa busca foi utilizada para melhorar a eficiência deste algoritmo genético. Este método proposto mostra que um algoritmo genético de busca local baseado em agentes é capaz de encontrar soluções ótimas globais ou quase ótimas para diversas possibilidades do problema de *job shop*.

(DI NAPOLI; DI NOCERA; ROSSI, 2014) apresentam a possibilidade de usar a negociação entre agentes para resolver o problema de estacionamento. Essa negociação leva em consideração as preferências dos motoristas, em relação à vaga

desejada, dos vendedores, quanto à ocupação do estacionamento, e da cidade, em relação ao congestionamento do centro da cidade. Para isso, propõe-se a implantação de um Sistema para negociação de vagas entre agentes em uma cidade inteligente usando um protocolo de negociação controlado por um agente centralizador. É mostrado que este mecanismo de negociação automatizado entre o agente centralizado e os agentes motoristas, permite gerenciar essas diferentes necessidades dos elementos envolvidos.

(DI NAPOLI; ROSSI, 2021), desenvolveram um *marketplace* com mecanismo de negociação distribuída em Smart Parking criado em um SMA. A negociação distribuída entre os gerentes de estacionamento e motoristas permite que as vagas sejam escolhidas de acordo com suas preferências após soluções aceitas por ambas as partes. É utilizado um método de precificação dinâmica no valor das vagas como incentivo aos motoristas utilizarem melhor estas e limitar a circulação do tráfego em áreas específicas da cidade. A negociação do *marketplace* foi avaliada de acordo com um grau de satisfação das partes envolvidas. Foi usada uma função de utilidade encontrada pela soma ponderada entre a disponibilidade de vagas no momento do pedido e a distância delas até áreas de grande ocupação. Os resultados mostraram que quando os pedidos são inferiores ao número de vagas disponíveis, aumentar a quantidade de gerentes eleva a competição e causa menor utilidade média entre eles. Essa competição retorna mais opções para os motoristas e sua utilidade média aumenta. Em caso inverso, mais pedidos que o total de vagas disponível levou a uma média de utilidades constantes de gerentes e motoristas.

(SAKURADA; BARBOSA; LEITÃO, 2019) analisam diferentes implementações de interfaces entre agentes de um sistema de Smart Parking com os dispositivos de controle físico das vagas. Foram implementadas quatro interfaces: Remoto / Cliente-Servidor, onde o agente da vaga e controlador estão em diferentes plataformas e comunicam-se por um canal direto; Remoto / Publicar-Assinar, também em diferentes localizações porém comunicando-se por protocolo MQTT (*Message Queue Telemetry Transport*); No dispositivo / Cliente-Servidor, onde o agente da vaga está embarcado na mesma plataforma que o controlador, sendo executados juntamente; e No dispositivo / Publicar-Assinar, onde o agente da vaga e o controlador também são executados no mesmo ambiente embarcado, porém realizam sua comunicação pelo protocolo MQTT. Essas interfaces foram testadas e avaliadas de acordo com seus tempos de resposta,

escalabilidade e reutilização em cenários com diferentes esquemas de interação e tecnologias. Os testes realizados mostram que os tempos de resposta foram mais rápidos na interface No dispositivo / Cliente-Servidor e mais escaláveis na interface No dispositivo / Publicar-Assinar.

### 4.3 Alocação de Recurso

Nestes trabalhos o foco encontra-se primariamente no desenvolvimento e otimização de sistemas de alocação de recursos.

O aplicativo da empresa *BestParking* (PARKWHIZ, 2021) é um sistema de alocação e previsão de vagas de vários estacionamentos nos Estados Unidos. Este sistema online aloca e reserva vagas em diversos estacionamentos de acordo com a preferência do cliente (preço da vaga e horário de estacionar). Esta plataforma, integrada com 12.000 instalações de estacionamentos parceiros, facilita o processo de procura por vagas e é utilizada por mais de 750.000 motoristas.

(ALIEDANI; LOKE, 2018) desenvolveram um SMA aberto no qual os veículos (agentes) entram e saem frequentemente do estacionamento e comunicam entre si oportunamente quando estão em uma distância para usar canais de comunicações dedicadas de curto alcance (do inglês, *Dedicated Short-Range Communications*) ou DSRC um do outro. O objetivo dessa comunicação entre agentes é compartilhar informações e conselhos entre si com o intuito de ajudar uns aos outros a encontrar uma vaga o mais próxima possível do destino utilizando DSRC (Comunicações dedicadas de curto alcance) como método de troca de dados. Os resultados deste trabalho mostram a capacidade deste sistema de alocar boas vagas de forma rápida contendo informações iniciais limitadas.

(AMINI; MOGHADDAM; KARABASOGLU, 2017) propõem uma abordagem de dois estágios para a alocação de estacionamentos de veículos elétricos e recursos renováveis fornecidos em rede de distribuição de energia. O método desenvolvido tem em mente favorecer os desejos econômicos do investidor em estacionamentos e colaborar com as restrições técnicas do operador da rede de distribuição. O investidor oferece os ônibus candidatos para a instalação do estacionamento para o operador com base em seus objetivos econômicos e a tomada de decisão da rede de distribuição é obtida para reduzir a perda do sistema. Para resolver o problema de otimização

formulado, foi utilizado um algoritmo genético e de otimização de enxame de partículas a fim de minimizar de perdas na rede de distribuição. O método proposto melhora a perda da rede de distribuição e a disponibilidade do estacionamento do ponto de vista econômico.

#### 4.4 Precificação Dinâmica

As propostas a seguir mostram diversos métodos e casos de uso do paradigma de modificação do preço dinamicamente.

(CHEN; SHELDON, 2015) realizam uma medição de como preços dinâmicos afetam a oferta de trabalho de motoristas do Uber. Foram realizadas medições e caracterizações das elasticidades do número de motoristas para encontrar as elasticidade de ofertas significativas e substancialmente positivas. Observou-se que o surto de preços das viagens diminuem significativamente a taxa de parada. Motoristas da Uber passaram a ajustarem suas programações para dirigir por mais tempo e fornecer mais viagens nos momentos com maior demanda. Essa conclusão sugere que o aumento do preço aumenta o número de viagens ocorridas e melhora a eficiência do sistema.

Em (WILLIAMS, 2018) foi criado um modelo de cálculo dinâmico de preços de companhias aéreas. Para isso, é definido que as tarifas são determinadas por discriminação intertemporal de preços e ajuste dinâmico à demanda estocástica. Este modelo de precificação dinâmica contabiliza essas duas forças com dados novos de voos para criar estimativas de preço. Com essas é separado principais interações entre o padrão de chegada dos tipos de consumidores e a capacidade restante sob demanda estocástica. O resultados com uso do modelo mostraram um aumento da receita de companhias aéreas e lucro do consumidor em comparação com um regime de preços mais restritivo.

(TIAN *et al.*, 2018) propõe um modelo de precificação dinâmica para reserva de vagas em estacionamentos a fim de maximizar a receita esperada. Neste modelo, o preço varia de acordo com a demanda de chegada e número de vagas. As requisições de vagas chegam seguindo um processo de Poisson e o preço variável é utilizado para alterar a intensidade dessa chegada. Foi mostrado que essa alteração dinâmica de preços pode resultar em melhor uso dos recursos do estacionamento durante períodos de pico e aumento significativo da receita.



(PIERCE; SHOUP, 2013) revisam o aplicativo (SFPARK, 2015), um programa que ajusta preços para a aumentar a disponibilidade de vagas. De modo a descobrir a influência do preço sobre a ocupação da rua, calculou-se a elasticidade do preço sobre a demanda em mais de 5000 alterações do preço e medido a ocupação durante um ano de uso do programa. Foi visto que o preço varia de acordo com a hora do dia, o local da vaga e outros fatores. A utilização do (SFPARK, 2015) resultou em uma redução de 1% do preço médio durante o primeiro ano e foi possível ajustar os preços sem aumentá-lo, em geral.

(LIAO *et al.*, 2021) propõem uma estratégia de precificação dinâmica para uma rede de computação de borda multi-acesso auxiliada por veículos multi-usuário. Esta estratégia propõe minimizar o consumo de energia de equipamentos de usuários (EUs), enquanto maximiza a receita dos provedores de serviços móveis. É usado um algoritmo de Evolução Diferencial (ED) para escolher a decisão refinada do *offloading* para o EU. Este é auxiliado por um algoritmo de *Q-learning* na seleção do preço adequado para o serviço. Os resultados deste artigo mostram que essa estratégia de preços dinâmica pode ter desempenho superior à de preços fixos e diferenciados em relação ao custo cumulativo geral. Além disso, é indicado que essa abordagem promove maior eficiência no uso de recursos comparado a outros algoritmos de execução local e de descarregamento aleatório.

(PIERCE; WILLSON; SHOUP, 2015) mostram como as cidades dos EUA gerenciam precariamente suas estruturas de estacionamento em meio-fio. É descrito como a gestão atual ignora benefícios públicos e lógicas econômicas. Isso é apresentado usando um caso conceitual de como as cidades deveriam gerenciar estes a fim de maximizar estes benefícios. Os dados de 14 garagens que usadas no programa (SFPARK, 2015) em São Francisco, foram usados na criação do exemplo mais promissor de gestão de estacionamento em meio fio.

É apresentado em (CAO *et al.*, 2018), um modelo de precificação dinâmico para recursos de nuvem que considera o tempo de ocupação e os consumos simultâneos e singulares destes recursos. O objetivo é ajudar os usuários a economizar no custo de uso mantendo o lucro dos prestadores do serviço. A partir de sua implementação, usando uma árvore B+ como método de controle dos dados, foi encontrado que o modelo é capaz de calcular essa precificação ao conseguir consultar 10 milhões de dados em cerca de 0,2 segundos.

(LI *et al.*, 2019) propõe um modelo de preço dinâmico baseado no custo nivelado do calor e na demanda de calor por hora prevista realizado um estudo de caso em um sistema de energia na Suécia. Foram usados três métodos para associar o custo do combustível ao dos custos variáveis da produção de calor: a proporção da quantidade e exergia de calor; eletricidade gerada; e uma dedução do preço de mercado da eletricidade. Nos resultados é mostrado que o modelo proposto é capaz de refletir o custo de produção do calor.

Esses trabalhos mostraram o contexto atual de soluções utilizando agentes inteligentes, métodos de alocação de recursos, implementações de *Smart Parking*, paradigma de precificação dinâmica e combinações destas técnicas de desenvolvimento. Além de sua contribuição como expansão do projeto MAPS, este trabalho contém todas as características apresentadas previamente. O módulo de precificação dinâmico foi desenvolvido com foco em sua capacidade de implantação em qualquer SMA de estacionamentos inteligentes.

#### 4.5 Projeto MAPS

Estes trabalhos fazem parte do projeto MAPS que possui o objetivo de desenvolver SMAs para automatizar as funções de estacionamentos.

(CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, 2017) desenvolveu um SMA com a função de gerenciar vagas em um Smart Parking utilizando o *framework* JaCaMo. Este sistema é composto por dois tipos de agentes: *manager* que é responsável pela administração das vagas; e *drivers* que representam os motoristas querendo utilizar o estacionamento. As vagas são atribuídas a de acordo com um grau de confiança que cada *driver* possui. este grau de confiança é um valor que mostra o comprometimento do *driver* em relação ao estacionamento. Foram realizadas diversas simulações em cenários experimentais para determinar a efetividade do sistema. Os resultados mostraram o funcionamento do sistema e que *drivers* com grau de confiança maiores recebem vaga em menor tempo.

Em (DUCHEIKO; BORGES; ALVES, G. V., 2018), é realizado uma expansão do projeto MAPS com a criação de um modelo de raciocínio e protocolos para implementação de negociações de vagas descentralizado. Este trabalho descreve os protocolos e fórmulas elaboradas para a negociação entre os agentes, e é implementado um SMA

utilizando o *framework* JaCaMo. Destaca-se a organização social entre agentes no sistema com o uso de regras do Moise.

Em (BOTELHO; BORGES; ALVES, G. V., 2019) é proposto implantação dos agentes em uma arquitetura que utiliza Raspberry Pi para a captura das informações de um estacionamento. Os agentes serão desenvolvidos no *framework* JADE de forma descentralizada e serão integrados no sistema ciber-físico estudado. O trabalho em questão está em sua fase inicial e portanto alguns critérios do sistema não foram avaliados.

No trabalho de (CASTRO, L. F. S. *et al.*, 2020), é proposto uma abordagem de integração de um SMA, desenvolvido no *framework* JaCaMo com a ferramenta de simulação urbana (SUMO), sistemas embarcados e aplicações de IoT. Está é apresentada em seus níveis de agentes e artefatos, ambiente físico embarcado, simulação urbana e componentes IoT. O detalhamento de cada parte dessa integração apresenta sua capacidade de generalização. A baixa necessidade de acoplamento entre camadas permite que diferentes aplicações utilizem desta abordagem como suporte com a flexibilidade de utilizar toda a integração ou apenas pedaços dela.

(BOTELHO; ALVES, G. V.; BORGES, 2020) descreve a implementação e integração de um SMA, que gerencia as vagas, um aplicativo móvel para comunicação com o estacionamento, componentes IoT, que detectam veículos e um banco de dados em nuvem para o armazenamento de informações. A arquitetura propõe versatilidade ao permitir que qualquer dispositivo móvel realize uma requisição e facilidade da comunicação entre os dispositivos, necessitando apenas de acesso a internet. Os experimentos realizados mostram que esta arquitetura é capaz de ser executada em outros SMA, utilizando outros protocolos de negociação, desde que receba uma vaga como entrada em suas negociações e retorne a vaga negociada como saída.

O quadro 2 apresenta uma comparação detalhada com todos os trabalhos neste capítulo. Ela é organizada em cinco colunas descrevendo em sequência: a referência do trabalho na linha; a proposta do trabalho; seu objetivo; a metodologia utilizada; e seus resultados.

**Quadro 2 – Comparação entre os trabalhos relacionados**

Trabalho	Proposta	Objetivo	Metodologia	Resultados
(BENENSON; MARTENS; BIRFIR, 2008)	Modelo Espacial em SIG de ambiente urbano para processo de estacionamento.	Simular comportamento de múltiplos <i>drivers</i> em um ambiente espacialmente explícito.	Baseado em agentes que controlam os motoristas usando regras pré-estabelecidas para encontrar vagas.	Capaz de capturar a situações complexas da dinâmica de grandes conjuntos de agentes, bem como os impactos no ambiente urbano.
(PIERCE; SHOUP, 2013)	Revisão do programa <i>Sfpark</i>	Descobrir a influência do preço em relação à ocupação de estacionamento na rua.	Foi calculado a elasticidade sobre a demanda em mais de 5000 alterações de preços medidas ao decorrer de um ano de operação.	Identificado que preços alteram de acordo com horário, local da vaga e outros fatores. Seu uso resultou em redução média de 1% dos preços.
(DI NAPOLI; DI NOCERA; ROSSI, 2014)	<i>Smart Parking</i> baseado em Sistema Multi-agente.	Resolver problema de mobilidade de motoristas procurando por estacionamento.	A negociação de vagas no SMA considera a preferência dos motoristas, vendedores e da cidade.	Os resultados mostram que esse mecanismo de negociação de vagas pode gerenciar as necessidades dos usuários.
(SHIN; JUN, 2014)	Algoritmo de orientação de <i>Smart Parking</i>	Reduzir tempo e consumo de energia na procura por vagas de estacionamento.	Orientação considera tempo até vaga, distância da vaga até destino, custo da vaga, probabilidade de disponibilidade e congestionamento até vaga.	Resultados mostram que o tempo de viagem, número de vagas utilizadas e uso dos recursos da cidade podem ser melhorados.
(ASADZADEH, 2015)	Algoritmo genético de busca local implementado em um sistema-multiagente.	Resolver o problema de agendamento de <i>job shop</i> (escalonamento de processos).	Algoritmo genético implementado com duas buscas locais. Uma explora todas as soluções e a outra aprimora a melhor solução atual.	Capaz de encontrar soluções ótimas globais ou quase ótimas para diversas circunstâncias do problema de escalonamento de recursos.
(CHEN; SHELDON, 2015)	Pesquisa de como preços dinâmicos afetam motoristas de Uber.	Identificar as influências da alteração dinâmica dos preços de viagens nas ações dos motoristas de Uber.	Mediram e caracterizaram as elasticidades de ofertas significativas.	Motoristas ajustaram suas programações de viagem a fim de dirigir por mais tempo e procuraram fornecer serviço durante maior demanda.
(PIERCE; WILLSON; SHOUP, 2015)	Revisão do gerenciamento de estacionamentos em cidades dos EUA.	Apresentar um caso conceitual de como cidades deveriam gerenciar seus estacionamentos.	Utilizaram os dados de 14 garagens na cidade de São Francisco que usaram do programa (SFPARK, 2015).	Os resultados usando o programa mostram preços menores e maiores ocupações em estacionamentos, mantendo a receita para a cidade.
(AMINI; MOGHADDAM; KARABASOGLU, 2017)	Alocação de veículos elétricos e recursos renováveis.	Otimizar rede de distribuição de energia, disponibilidade e lucro do estacionamento.	Algoritmo genético e otimização de enxame são usados na minimização de perdas na rede de distribuição de acordo com desejos do investidor.	O método proposto melhora a distribuição de rede e aumentou a disponibilidade do estacionamento.
(CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, 2017)	Sistema multi-agente usando <i>framework</i> JaCaMo.	Desenvolver um sistema com a função de gerenciar as vagas em um <i>Smart Parking</i> .	O agente <i>Manager</i> gerencia as vagas que às atribuídas de acordo com um grau de confiança de cada <i>driver</i>	A simulações mostraram o funcionamento do sistema e que os <i>drivers</i> com os maiores graus de confiança receberam vagas em menor tempo.
(LIN; RIVANO; LE MOUËL, 2017)	Pesquisa de trabalhos envolvendo <i>Smart Parking</i> .	Classificar e identificar funcionalidades e focos problemáticos em estacionamentos inteligentes.	Sistemas categorizados de acordo com três macrotemas: coleta de informações, implantação de sistemas e disseminação de serviços.	Classificações abrangem áreas sensoriais, organizacionais, predição, análise de dados, conectividade e competição de <i>Smart Parkings</i> .
(ALIEDANI; LOKE, 2018)	Sistema multi-agente descentralizado.	Compartilhar informações entre agentes para encontrar vagas próximas a seu destino.	Troca de informações e alocação de vagas entre os veículos do sistema realizado utilizando <i>DSRC</i> .	Os resultados mostraram que o sistema possui a capacidade de alocar boas vagas rapidamente com informações iniciais limitadas.
(CAO <i>et al.</i> , 2018)	Precificação Dinâmica do uso de recursos em nuvem.	Economizar custo e manter lucro.	Uso de uma <i>Árvore B+</i> no controle de dados.	O modelo desenvolvido pode consultar 10 milhões de dados em 0,2 segundos.
(DUCHEIKO; BORGES; ALVES, G. V., 2018)	Modelo de raciocínio e protocolos de negociação de vagas descentralizado.	Desenvolver um sistema de negociação descentralizada entre agentes em um <i>Smart Parking</i> .	Protocolos e fórmulas de negociações em agentes de um SMA implementado usando JaCaMo.	Simulações mostraram o funcionamento dos protocolos de negociações para o funcionamento do estacionamento.
(RIZVI; ZEHRRA; OLARIU, 2018)	Agente Inteligente em nuvem.	Determinar a vaga apropriada para um motorista em uma rede de estacionamentos.	Um agente possui interação com serviços da cidade e gerência a alocação de vagas de acordo com preferências de usuários.	Simulações indicam que o sistema desenvolvido é escalável e adaptável a diversos cenários.
(TIAN <i>et al.</i> , 2018)	Precificação Dinâmica de vagas em estacionamento.	Alterar dinamicamente o preço de reserva de vaga e maximizar a receita esperada do gerente.	Preço alterado de acordo com demanda e número de vagas. Solicitações usam processo de Poisson e sua intensidade de acordo com preço.	O módulo pode proporcionar uma maior receita e fazer pleno uso dos recursos de estacionamento durante os períodos de pico.
(WILLIAMS, 2018)	Modelo de cálculo dinâmico de preços	Determinar preços para viagens aéreas de acordo com características temporais e dinâmicas dos voos.	O modelo contabiliza as interações chave entre o padrão de chegada dos tipos de consumidores e a capacidade restante sob demanda estocástica.	Receitas significativamente mais altas, além de aumento do excedente do consumidor comparado com um regime de preços mais restritivo.
(BOTELHO; BORGES; ALVES, G. V., 2019)	Propõem uma implantação de agentes em ambiente embarcado.	Estudo de agentes implementados em ambiente físico.	Implantação em uma arquitetura que utiliza <i>Raspberry Pi</i> na captura de informações para o estacionamento.	Trabalho em fase inicial e logo não foi avaliado.
(LI <i>et al.</i> , 2019)	Modelo de predição de preço do combustível dinamicamente.	Modelar predição de custo da produção de calor.	Modelo associa este custo a quantidade e energia de calor, eletricidade gerada e preço estimado de eletricidade.	O modelo corretamente refletiu preços reais.
(SAKURADA; BARBOSA; LEITÃO, 2019)	Análise de diferentes interfaces <i>Smart Parking</i> com controle físico das vagas.	Comparar os tempos de resposta, escalabilidade e reutilização das interfaces analisadas.	Comparação entre interfaces categorizadas entre combinações de sistema remoto e no dispositivo com cliente-servidor e publicar-assinar.	Tempos de resposta foram mais rápidos na interface No dispositivo / Cliente-Servidor e mais escaláveis No dispositivo / Publicar-Assinar.
(BOTELHO; ALVES, G. V.; BORGES, 2020)	Integração de um SMA a um ambiente embarcado <i>IoT</i> .	Gerenciar vagas por aplicativo móvel e banco de dados em nuvem para armazenamento de informações.	Arquitetura que utiliza <i>Raspberry Pi</i> na captura de informações para o estacionamento em qualquer dispositivo móvel.	Os experimentos mostram que esta arquitetura pode ser executada em vários SMAs com diferentes protocolos de negociação.
(CASTRO, L. F. S. <i>et al.</i> , 2020)	Abordagem de integração de um SMA em sistemas embarcados e aplicações de <i>IoT</i> .	Novas tecnologias para resolver questões de mobilidade, heterogeneidade e baixo acoplamento.	Desenvolvido no <i>framework</i> JaCaMo com a ferramenta de simulação urbana (SUMO).	Diferentes aplicações com está abordagem permite a flexibilidade de utilizar toda ou parte da integração.
(LIAO <i>et al.</i> , 2021)	Precificação Dinâmica para MEC.	Minimizar energia gasta e maximizar receita.	Uso de algoritmo diferencial no <i>offloading</i> , auxiliado por <i>Q-Learning</i> na seleção do preço.	Maior eficiência no uso de recursos e desempenho superior às estratégias de preços fixos e diferenciados.
(DI NAPOLI; ROSSI, 2021)	<i>Marketplace</i> com negociação distribuída em <i>Smart Parking</i> criado em SMA.	Escolha de vagas de acordo com preferências e soluções aceitas por ambos gerentes e motoristas.	Precificação dinâmica no valor das vagas e avaliação da negociação do <i>marketplace</i> de acordo com um grau de satisfação.	Um número de pedidos menor que de vagas disponíveis gera aumento na competição entre gerentes. Ao contrário, mostra-se um equilíbrio da média de utilidade entre usuários e gerentes.
(PARKWHIZ, 2021)	Sistema alocação, previsão e reserva de vagas de estacionamentos	Facilitar processo de procura e reserva de vagas de estacionamentos	Sistema online que aloca e reserva vagas de diversos estacionamentos parceiros conforme as preferências dos motoristas	A Plataforma é utilizada por mais de 750 mil motoristas e possui parceria com mais de 12 mil instalações.
(SAID; KAMAL; AFIFI, 2021)	<i>Smart Parking</i> sustentável	Reduzir tempo de procura por vaga, aumentar lucro e maximizar sustentabilidade	Alocação de vagas modelado usando ramo de teoria dos jogos e baseado em internet das coisas	Simulações indicam que o <i>Smart Parking</i> consegue cumprir seus objetivos.

Fonte: Autoria própria (2021).

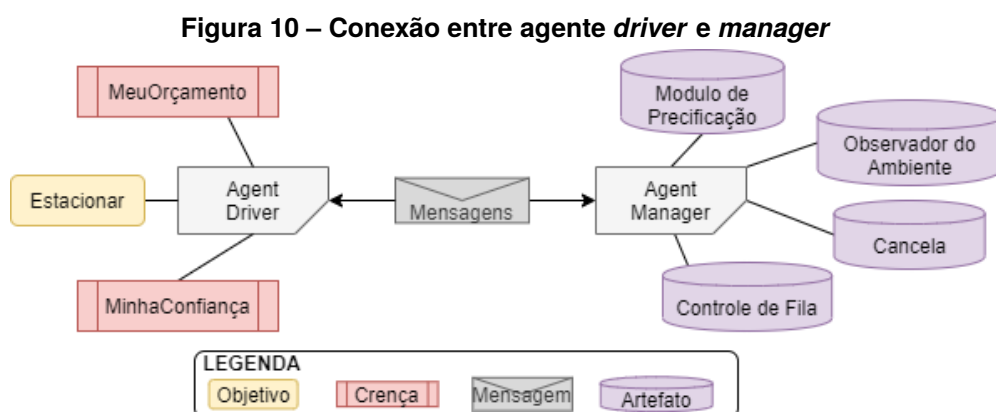
## 5 DESENVOLVIMENTO DO TRABALHO

Este trabalho visa desenvolver um mecanismo de controle dinâmico de preço de vagas que possa ser aplicado em qualquer *Smart Parking* com a capacidade de transmitir dados sobre demanda, horário e clima. A modificação do preço é realizada considerando características temporais do preço: o valor muda de acordo com o horário e condição atmosférica do dia; e de acordo com a quantidade de vendas: o valor varia de acordo com o número de pedidos de vagas realizados em cada período.

A seguir, na seção 5.1 é descrito o desenvolvimento do SMA. Na seção 5.5 são apresentadas as regras de alteração do preço. Na seção 5.2 tem-se a apresentação das variáveis do sistema. Os artefatos deste sistema são apresentados na seção 5.3. Depois, na seção 5.4, é descrito a reorganização de vagas. O processo de precificação é discutido na seção 5.6. Um exemplo contendo o processo de atualização do preço é realizado na seção 5.7. Por fim são apresentados as considerações finais do capítulo.

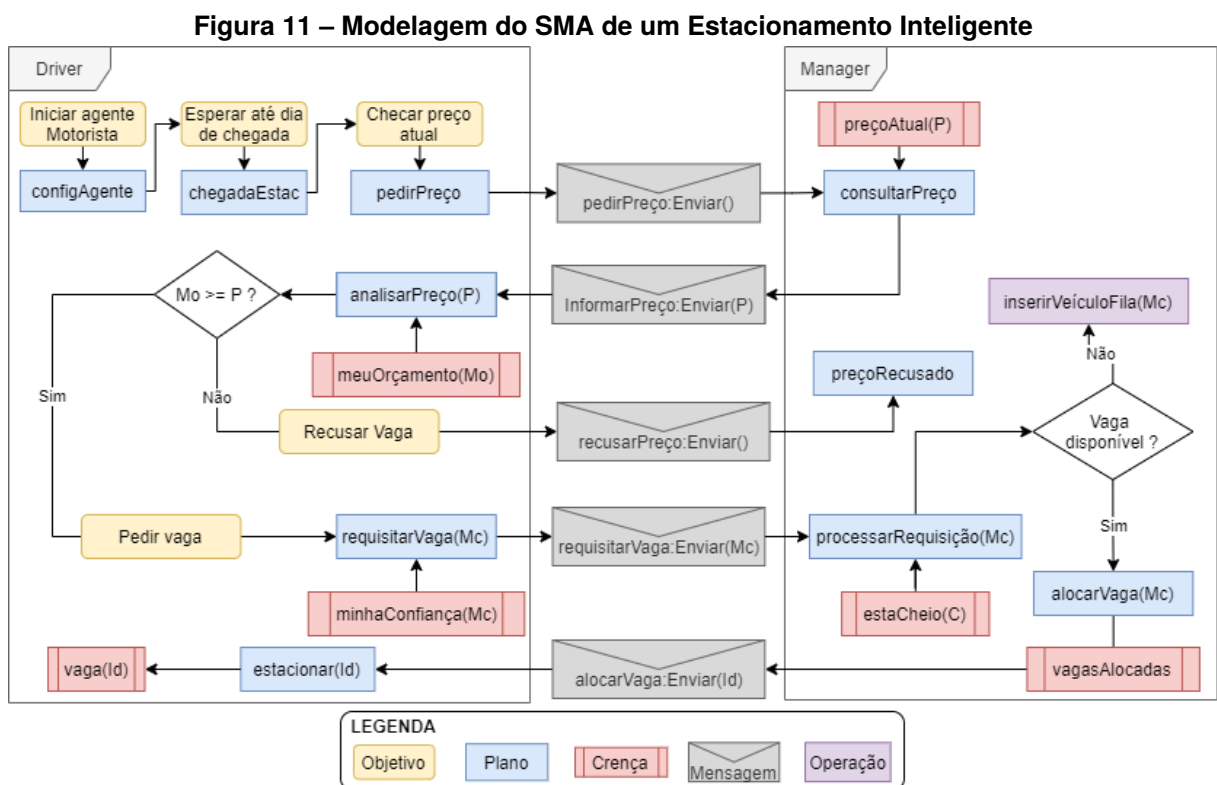
### 5.1 SMA Desenvolvido

Para implantar este módulo foi necessário o desenvolvimento de um SMA. Foi utilizado como base o sistema criado por (CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, 2017) e foram adicionadas funções para permitir a negociação entre os agentes *driver* e *manager* e a aplicação de preços nas vagas. A Figura 10 ilustra a conexão entre um agente *driver* o *manager* de forma simplificada. Neste SMA centralizado, o agente *manager* controla a entrada, alocação e pagamento dos agentes *drivers* interessados em estacionar. Os agentes *drivers* são encarregados de simular motoristas que decidem entre aceitar ou não o preço das vagas.



Fonte: Autoria Própria (2021).

A Figura 11 ilustra a modelagem do Smart Parking utilizado na implantação do módulo de precificação. Este diagrama mostra o processo de um motorista requisitar por uma vaga. O processo tem início com o *driver*. Após sua inicialização, ele espera até o dia de chegada para verificar com o *manager* o preço atual das vagas. Ao receber o valor, ele compara com seu orçamento e, caso este seja maior ou igual ao preço, requisitar uma vaga enviando seu grau de confiança. O agente *manager* processa esta requisição verificando se existem uma vaga disponível. Caso exista, ele atualiza seu conjunto de crenças e informa o motorista que estaciona na vaga.



Fonte: Autoria Própria (2021).

O agente *manager* é o ponto central do gerenciamento de todo o estacionamento. Ele está encarregado de receber, informar e alocar agentes *drivers*. O *manager* também é responsável por decidir quando alterar o preço de forma dinâmica (após observar a demanda, horário e condição climática do ambiente do estacionamento) ou de forma estática (utilizando planos internos). O módulo de precificação é encarregado de retornar o novo preço das vagas quando requisitado pelo agente *manager*. Esse valor é definido a partir das variáveis do ambiente atual e das regras definidas na configuração do estacionamento (ver Subseção 5.5.1).

## 5.2 Variáveis do Sistema

Durante a simulação do EI, os valores de algumas variáveis modificam relativamente o funcionamento do sistema. O agente *driver*, quando criado, possui quatro variáveis, ou crenças, que determinam suas ações. Essas variáveis recebem valores aleatórios dentro de um intervalo pré-definido. Seguem as descrições de suas funções e valores possíveis:

- *daysToArrive*: Determina em quantos dias em que o agente *driver* irá chegar ao estacionamento após ser criado. Definido entre 0 e 7 dias (tempo de simulação);
- *arriveTime*: Determina o horário em que o agente *driver* irá chegar no estacionamento no dia previamente determinado. Definido entre as 02:00h e 22:00h (tempo de simulação);
- *timeToSpend*: Determina o tempo em que o agente *driver* irá passar estacionado em sua vaga. Definido entre 1 hora e 3 dias (tempo de simulação);
- *myTrust*: Variável que determina o valor de confiança usada como meio de priorização de motoristas ao negociar com o agente *manager*. Definido entre 0 e 1000;
- *myBudget*: Valor dos créditos disponíveis ao agente *driver* para requerir uma vaga no estacionamento. Definido entre 5 e 20 créditos.

O agente *manager* possui quatro variáveis. Destas, duas são usadas no gerenciamento do estacionamento e as outras são utilizadas para o controle do preço das vagas:

- *spotData*: Contém o número total de vagas e quantas dessas vagas estão ocupadas no estacionamento. O número total de vagas é definido estaticamente e não muda durante a execução da simulação. O valor de vagas ocupadas é iniciado em zero e é incrementado a partir do uso dessas vagas por *drivers*;
- *sectorList*: Contém a configuração dos setores do estacionamento. Cada elemento nesta lista representa um setor que possui um nome e a porcentagem de vagas que serão identificadas por este nome. A soma dessas porcentagens deve ser igual a 100%;
- *spotTypeList*: Contém a configuração dos tipos de vagas no estacionamento. Cada elemento nesta lista representa um tipo de vaga que possui um nome



e porcentagens de vagas que serão identificadas por este nome. A soma dessas porcentagens deve ser igual a 100%;

- *basePrice*: Valor base do preço das vagas. Esta é a variável usada durante os cálculos de alteração de preço. O seu valor é estático durante a simulação;
- *currentPrice*: Determina o preço atual das vagas. Este valor começa igual ao valor base e é a variável alterada após execuções do módulo de precificação.

### 5.2.1 Planos do Agente Driver

O agente *driver* simula as ações de um motorista procurando por vaga. Nele são contidos planos para chegar a um estacionamento, negociar preço em relação a um orçamento pessoal, estacionar e deixar o estacionamento. Os planos a seguir são em parte modificados a partir do sistema de (CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, 2017). Os planos menos alterados são aqueles de estacionar e sair da vaga onde foram apenas adicionados indicações do tipo de vaga usada. Todos os planos restantes foram alterados ou criados para aceitar negociações de preço e tipo de vaga.

- *setupDriver*: Criação das crenças do agente *driver* e definição do tempo espera até chegar ao estacionamento (algoritmo 5). Nesta crenças são contidas: Quando usará uma vaga, nas linhas 2 e 3 (*daysToArrive* com valor entre 0 e 6 dias e *arriveTime* com possíveis horários entre 2:00 e 20:59), o tempo de uso da vaga, na linha 4 (*timeToSpend* recebendo de 1 hora até 70 horas e 59 minutos), a confiança do agente, na linha 6 (*myTrust* com valor entre 0 e 999), seus créditos disponíveis, na linha 7 (*myBudget* entre 5 e 10 créditos) e o seu tipo de vaga desejada, na linha 8 (*myType* com valor entre 0 e 9 para determinar ponderadamente o tipo de vaga). Após a criação das crenças, o agente define como objetivo chegar ao estacionamento (*!arriveParking*);

#### Algoritmo 5 – Definição das crenças do agente *driver*

---

```

1 +!setupDriver : true <-
2   +daysToArrive(math.random(7));
3   +arriveTime(2+math.random(19), math.random(60));
4   +timeToSpend(1+math.random(70), math.random(60));
5
6   +myTrust(math.random(1000));
7   +myBudget(5+math.random(6));
8   +myType(math.random(10));
9
10  !arriveParking.
```

---

Fonte: Autoria Própria (2021).



- *arriveParking*: Plano em que o agente espera até o dia de usar o estacionamento (algoritmo 6). Nas linhas 2, 3 e 4, a função interna *.wait()* faz o agente esperar respectivamente até o dia desejado (DA), até o horário escolhido (AH) e minuto específico (AM). Ao final do plano, na linha 5, é definido como objetivo requisitar o preço das vagas (*!askPrice*);

#### Algoritmo 6 – Plano de espera até chegar ao estacionamento

---

```

1 +!arriveParking : daysToArrive(DA) and arriveTime(AH, AM) <-
2   .wait(DA*24000)
3   .wait(AH*1000)
4   .wait(AM*17)
5   !askPrice.

```

---

Fonte: Autorial Própria (2021).

- *askPrice*: Agente *driver* envia uma mensagem para o agente *manager* solicitando o preço atual das vagas do estacionamento (algoritmo 7). Usando sua crença contendo seu grau de confiança *myTrust*, a função interna *.send()* é usada para adicionar o objetivo *consultPrice* no agente *manager*;

#### Algoritmo 7 – Pedir preço atual das vagas

---

```

1 +!askPrice: myTrust(MT) <-
2   .send(manager,achieve,consultPrice(MT)).

```

---

Fonte: Autorial Própria (2021).

- *confirmPrice*: Plano é iniciado quando o agente *manager* retorna uma mensagem contendo o preço das vagas (algoritmo 8). Neste plano é realizada a consideração deste valor em relação aos créditos disponíveis do agente. Caso este valor seja aceito, quando o preço atual da vaga P é menor ou igual aos créditos MP disponíveis, o agente define como próximo objetivo pedir uma vaga (*!requestSpot*). Caso contrário, o agente envia uma mensagem para o agente *manager* recusando o preço, espera 24 horas e redefine suas crenças para reutilizar o estacionamento;

#### Algoritmo 8 – Confirmar preço recebido

---

```

1 +!confirmPrice(P)[source(Manager)] : myBudget(MP) and P <= MP <-
2   !requestSpot.
3
4 +!confirmPrice(P)[source(Manager)] : myBudget(MP) and P > MP <-
5   .send(manager,achieve,priceRefused);
6   .wait(24000)
7   !setupDriver.

```

---

Fonte: Autorial Própria (2021).

- *requestSpot*: Plano é iniciado caso o agente tenha aceitado o preço de uma vaga do estacionamento. Neste plano é enviado uma mensagem ao agente *manager* pedindo por uma vaga disponível (algoritmo 9). A função interna *.send()* adiciona o objetivo *processRequisition* contendo o grau de confiança MT e tipo de vagas MTy desejado no *manager* a fim de requisitar uma vaga;

#### Algoritmo 9 – Requisitar por uma vaga

---

```

1 +!requestSpot : myTrust(MT) and myType(MTy) <-
2   .send(manager,achieve,processRequisition(MT,MTy)).

```

---

Fonte: Autoria Própria (2021).

- *park*: Plano iniciado quando o *manager* retorna uma mensagem contendo a vaga alocada (algoritmo 10). Neste plano o *driver* cria uma crença *spot* (linha 2) com o setor Sc e identificado S da vaga desta vaga indicada pelo *manager*. Neste mesmo plano o agente espera o período de tempo que deseja permanecer estacionamento indicado nas linhas 3 e 4 respectivamente aguardando SH horas e SM minutos. Ao fim dessa espera o objetivo de saída da vaga é iniciado na linha 5 com inserção da meta *!leaveSpot*;

#### Algoritmo 10 – Estacionar na vaga recebida

---

```

1 +!park(S,Sc,T)[source(Manager)] : timeToSpend(SH, SM) and spotOk <-
2   +spot(Sc,S);
3   .wait(SH*24000);
4   .wait(SM*17);
5   !leaveSpot.

```

---

Fonte: Autoria Própria (2021).

- *leaveSpot*: Plano iniciado quando o tempo de estadia do agente no estacionamento termina (algoritmo 11). Neste plano o *driver* começa enviando uma mensagem, na linha 2, para o *manager* que esta se retirando do estacionamento. Esta mensagem adiciona o objetivo *leavesSpot* no *manager*. Em seguida o *driver* remove sua crença de vaga, espera 24 horas e adiciona a meta *arriveParking* para reutilizar o estacionamento.

#### Algoritmo 11 – Sair da vaga

---

```

1 +!leaveSpot : spot(Sc,S) and myType(T) <-
2   .send(manager,achieve,leaveSpot(Sc,S,T));
3   spot(Sc,S);
4   .wait(24000);
5   !arriveParking.

```

---

Fonte: Autoria Própria (2021).

## 5.2.2 Planos do Agente Manager

O *manager* controla todos os aspectos do *Smart Parking*. Nele é implementada o controle do preço, recepção, alocação e remoção de agentes *drivers*. Os planos do *manager* foram criados e modificados usando como base o sistema em (CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, 2017). Todos os planos sofreram alterações devido a implementação de preço e reorganização de vagas. Os planos descritos a seguir que não existem no SMA base são de *!checkEnvironment*, *!saveRequest* e *!consultPrice*.

- *setupParking*: Inicialização dos artefatos utilizados pelo agente *manager* e criação de suas crenças (algoritmo 12). Os artefatos em questão são o sistema de checagem do ambiente na linha 2, a cancela de entrada do estacionamento na linha 4, o controle da fila de espera na linha 6, o módulo de precificação na linha 8 e o criador de *log* de eventos na linha 10. Após a inicialização destes artefatos são adicionadas as metas de criação das vagas *!createSpots* e organização destas *!organizeSpots*. O plano termina por ler o arquivo de configuração selecionado e preencher as matrizes de regras e de legendas nas linhas 16 e 17 com operações do artefato do módulo de precificação.

### Algoritmo 12 – Inicialização dos artefatos e organização das vagas

---

```

1 +!setupParking: basePrice(BP) <-
2   makeArtifact("Envrmt","mapsEngine.CheckEnvironment", [], ArtIdCE);
3   focus(ArtIdCE);
4   makeArtifact("Gate", "mapsEngine.Gate", [], ArtId);
5   focus(ArtIdG);
6   makeArtifact("Control", "mapsEngine.QueueControl", [], ArtIdQC);
7   focus(ArtIdQC);
8   makeArtifact("Price","mapsEngine.PriceEngine", [BP], ArtIdPE);
9   focus(ArtIdPE);
10  makeArtifact("Log","mapsEngine.Log", ["EventLog.csv"], ArtIdL);
11  focus(ArtIdL);
12
13  !createSpots;
14  !organizeSpots;
15
16  getMatrixData("ConfFile.csv");
17  getLegendData("ConfFile.csv").

```

---

Fonte: Aatoria Própria (2021).

- *checkEnvironment*: Neste plano é acionado o artefato de verificação do ambiente e o módulo de precificação (algoritmo 13). Caso tenha ocorrido alguma mudança no ambiente, prevista no arquivo de configuração (Subseção 5.5.1) o preço das vagas será atualizado conforme a nova situação. Como argu-

mento das operações, são necessários os dados das crenças de preço atual  $C_p$ , regra atual  $Cr$ , total de requisições  $NR_R$ , total de vagas  $NM$  e vagas usadas  $NU$ . A operação *check*, na linha 3, realiza observa os dados sobre o ambiente e retorna o conjunto de regras pertinentes na variável  $Rule$ . Na linha 5, a operação *updatePrice* remove as alterações do preço da regra anterior e a mesma na linha 6 aplica o retorna novo preço na variável  $Np$ . As linhas 7, 8 e 9 respectivamente reiniciam o total de requisições, atualiza o preço atual e a nova regra utilizada.

#### Algoritmo 13 – Observação do ambiente e atualização do preço

---

```

1 +!checkEnvironment: currentPrice(Cp) and currentRule(Cr) and
2     nrrequests(NRR)andspotData(NM,NU) <- check(NU, NRR, NM, Rule);
3 updatePrice(Cr, Cp, "rm", AuxP);
5 updatePrice(Rule, AuxP, "add", Np);
6
7 --nrrequests(0);
8 --currentPrice(Np);
9 --currentRule(Rule).
```

---

Fonte: Autoria Própria (2021).

- *consultPrice*: Plano iniciado após receber uma mensagem do agente *driver* pedindo pelo preço atual das vagas (algoritmo 14). O preço atual  $CP$ , contido nas crenças do agente *manager*, é enviado ao agente *driver* por uma mensagem por meio do plano interno *.send()*.

#### Algoritmo 14 – Consulta do preço e notificação do driver

---

```

1 +!consultPrice(TRUST)[source(AGENT)]: currentPrice(CP) <-
2     .send(AGENT,achieve,checkPrice(CP)).
```

---

Fonte: Autoria Própria (2021).

- *processRequisition*: Plano iniciado após receber uma mensagem do agente *driver* requisitando uma vaga (algoritmo 15). Neste plano adicionado a meta de alocar uma vaga ao agente *driver* que requisitou uma vaga na linha 2 (*!allocateSpot*). Esta meta contém o nome do agente, seu grau de confiança e o tipo de vaga desejada. A seguir, na linha 3, é adicionado o desejo de salvar esta requisição (*!saveRequest*) de acordo com o tipo de vaga que foi pedido.

#### Algoritmo 15 – Processamento de uma requisição por vaga

---

```

1 +!processRequisition(TRUST, TYPE)[source(AGENT)]: true <-
2     !allocateSpot(AGENT,TRUST,TYPE);
3     !saveRequest(TYPE).
```

---

Fonte: Autoria Própria (2021).

- *saveRequest*: Salva a requisição pelo tipo de vagas para ser usada durante a reorganização do estacionamento (algoritmo 16). A crença do tipo de vaga requerido *TYPE* é atualizada incrementando 1 no total de requisições deste tipo.

#### Algoritmo 16 – Salvando requisições por cada tipo de vaga

---

```

1 +!saveRequest(TYPE): spotTypeData(ID, TYPE, TReq, TPerc, TMax, TUsed) <-
2   -spotTypeData(ID, TYPE, TReq, TPerc, TMax, TUsed);
3   +spotTypeData(ID, TYPE, TReq+1, TPerc, TMax, TUsed).
```

---

**Fonte: Autoria Própria (2021).**

- *releaseSpot*: Plano iniciado quando um agente *driver* estacionado decide sair do estacionamento (algoritmo 17). Este agente é retirado da vaga nas linhas 2 e 3 e a lotação das vagas é atualizado na linha 4.

#### Algoritmo 17 – Retirando *driver* da vaga

---

```

1 +!releaseSpot(Sector,Id,Type)[source(AGENT)] : spotData(NM,NU) <-
2   -spot(Id,Sector,T,AGENT);
3   +spot(Id,Sector,T,"free");
4   +-spotData(NM,NU-1).
```

---

**Fonte: Autoria Própria (2021).**

- *allocateSpot*: Neste plano é feita uma busca e alocação de uma vaga disponível do tipo desejado pelo *driver* em questão (algoritmo 18). Para a execução deste plano são necessários os dados das crenças de lotação das vagas (*spotData*), número de requisições de vagas (*nr\_requests*) e do tipo de vaga desejado pelo *driver* (*spotTypeData*). O plano até a linha 6 é executada caso o estacionamento esteja cheio ou o tipo de vaga desejado não esteja disponível. Na linha 5 é incrementado o número de requisições por vaga e na linha 6 o *driver* e seus dados são inseridos na lista de espera. A partir da linha 8 ocorre caso ao menos uma vaga do tipo desejado esteja disponível. Entre as linhas 12 e 17 são encontradas todas as vagas disponíveis deste tipo e guardadas na lista *freeSpots*. Da linha 18 em diante, uma destas vagas é escolhida randomicamente sendo inseridos o identificador e setor nas variáveis *ID* e *SECTOR*. Estes dados são usados na remoção da crença da vaga vazia e adição de uma contendo o nome do agente *driver* nas linhas 23 e 24. A seguir são atualizadas as crenças de lotação das vagas e de número de requisições. No fim do plano, a cancela é aberta na linha 28 com a operação *openGate* e uma mensagem cria uma crença no *driver* indicando que ele pode estacionar.

Uma segunda mensagem adiciona no *driver* a meta de estacionar na vaga habilitada e na linha 31 a cancela é fechada com *closeGate*.

**Algoritmo 18 – Alocação de uma vaga, se disponível, a um *driver***

---

```

1 +!allocateSpot(AGENT,TRUST,TYPE) : spotData(NM,NU) and
2                               nrrequests(NRR) and
3                               spotTypeData(TyID, Ty, TReq, TPerc, TMax, TU) and
4                               (TU*100/TMax >= 100 | NU*100/NM >= 100) <-
5   --nrrequests(NRR+1);
6   insertDriverQueue(AGENT,TRUST,TYPE).
7
8 +!allocateSpot(AGENT,TRUST,TYPE) : spotData(NM,NU) and
9                               nrrequests(NRR) and
10                              spotTypeData(TyID, Ty, TReq, TPerc, TMax, TU) and
11                              TU*100/TMax < 100 NU*100/NM < 100 <-
12   +freeSpots(AGENT,[]);
13   for(spot(ID,SECTOR,TYPE,"free"))
14     -freeSpots(AGENT,Spots);
15     .concat(SpList,[[ID,SECTOR]],NewSpList);
16     +freeSpots(AGENT,NewSpList);
17
18   .random(FS,Spot);
19   -freeSpots(AGENT,FS);
20
21   .nth(0, Spot, ID);
22   .nth(1, Spot, SECTOR);
23   -spot(ID,SECTOR,TYPE,"free");
24   +spot(ID,SECTOR,TYPE,AGENT);
25   +-spotData(NM, NU+1);
26   +-nrrequests(NRR+1);
27
28   openGate;
29   .send(AGENT,tell,spotOk);
30   .send(AGENT,achieve,park(ID,SECTOR,TYPE));
31   closeGate.

```

---

Fonte: Autoria Própria (2021).

### 5.3 Artefatos do Sistema

Artefatos, como definidos na subseção 2.3.2, são recursos construídos para serem usados ou manipulados por agentes com o objetivo de realizar suas atividades individuais e coletivas. Neste sistema centralizado, apenas o agente *manager* possui acesso às operações dos artefatos para reduzir o número necessário de troca de mensagens entre agentes.

#### 5.3.1 Leitor do Arquivo de Configuração

Este artefato contém as operações de leitura e aplicação das regras encontradas no arquivo de configuração. A Figura 12 apresenta como exemplo o arquivo

de configuração 6 e o código de leitura deste arquivo encontra-se no Apêndice A. As propriedades e a construção deste arquivo são mostradas na subseção 5.5.1. As tabelas encontradas no arquivo são divididas em quatro matrizes:

- *legendHeader* contendo o cabeçalho da tabela de legenda;
- *legendaData* contendo o conteúdo da tabela de legenda;
- *rulesHeader* contendo o cabeçalho da tabela de regras;
- *rulesData* contendo as regras do arquivo de configuração;

**Figura 12 – Criação das matrizes usando o arquivo de configuração 6**

LegendHeader	ID	String	Intervalos
LegendData	1	Demanda	0-0.3-0.6-1
	2	CondClima	0-1-2-3
	3	HoraDia	0-6-12-15-19-24
RulesHeader	Categoria	Condicao	Valor
RulesData	HoraDia	Madrugada	-0.2
	HoraDia	Manha	0
	HoraDia	MeioDia	0.1
	HoraDia	Tarde	0
	HoraDia	Noite	-0.1
	Evento	Promocao	-0.5
	Demanda	Baixa	-0.2
	Demanda	Normal	0
	Demanda	Alta	0.3
	CondClima	Sol	0.1
	CondClima	Nublado	0
	CondClima	Chuva	-0.05
	CondClima	Tempestade	-0.1

**Fonte: Autoria Própria (2021).**

### 5.3.2 Checagem do Ambiente

Este artefato verifica a situação do ambiente e utiliza as matrizes criadas pelo artefato leitor a fim de retornar as regras adequadas para a situação atual. A Figura 13 ilustra as regras selecionadas de acordo com situações do ambiente. Usando a matriz contendo a tabela de legenda, as operações identificam quais características do ambiente serão observadas.

Estas características podem envolver a demanda do estacionamento, condição climática e horário do dia. A demanda por vagas é calculada baseada no último período de simulação e condição climática atual é determinada pseudo-randomicamente. Essas

características são usadas para encontrar o intervalo e identificar a condição atual de cada categoria.

**Figura 13 – Exemplo de verificação do ambiente usando o arquivo de configuração 6**

LegendHeader	ID	String	Intervalos	
LegendData	1	Demanda	0-0.3-0.6-1	
	2	CondClima	0-1-2-3	
	3	HoraDia	0-6-12-15-19-24	
RulesHeader	Categoria	Condicao	Valor	Ambiente
RulesData	HoraDia	Madrugada	-0.2	
	HoraDia	Manha	0	
	HoraDia	MeioDia	0.1	
	HoraDia	Tarde	0	
	HoraDia	Noite	-0.1	20:30
	Evento	Promocao	-0.5	
	Demanda	Baixa	-0.2	
	Demanda	Normal	0	
	Demanda	Alta	0.3	72%
	CondClima	Sol	0.1	
	CondClima	Nublado	0	Sensor 1
	CondClima	Chuva	-0.05	
	CondClima	Tempestade	-0.1	

**Fonte: Autoria Própria (2021).**

Um exemplo de execução é mostrado na Figura 13. Neste ambiente pensado, o horário atual é 20:30, a demanda do estacionamento no último período se encontra em 72% e o sensor 1 foi ativado pseudo-randomicamente. Nestas circunstâncias, pela tabela de legenda a demanda se encontra entre 60 e 100%, e logo a demanda esta alta de acordo com a tabela de regras; Com o sensor 1 ligado, a tabela de regras infere que o clima atual é nublado; E o horário de 20:30 se encontra no intervalo entre 19h e meia-noite, portanto a regra atual indica que está de noite.

O artefato retorna ao *manager* uma *String* formatada contendo as regras a serem usadas na próxima mudança de preço. A partir das características encontradas, demanda alta, clima nublado e que está de noite, pode-se determinar a nova regra de preço. No exemplo da Figura 13, essa regra é enviada da seguinte forma: "*Demanda:Alta/CondClima:Nublado/HoraDia:Noite*".

### 5.3.3 Alteração do Preço

A regra retornada do artefato de checagem do ambiente é usada para alterar o preço das vagas usando a operação mostrada no algoritmo 19. Usando como exem-



plo a regra estabelecida na subseção anterior na Figura 13, o novo valor da vagas é encontrado da seguinte maneira: *Demanda:Alta* aumenta o preço em 30%; *Cond-Clima:Nublado* não altera o valor; e *HoraDia:Noite* reduz o preço em 10%. Supondo um preço base de 5 créditos, o novo preço é encontrado usando a seguinte cálculo  $(5 + (0,3 * 5) + (0 * 5) + (-0,1 * 5))$  que resultaria em 6 créditos.

#### Algoritmo 19 – Operação de atualização do preço

---

```

1 @OPERATION
2 public void updatePrice(String terms, double currentPrice, String option, OpFeedbackParam<Object>
  Res) {
3     String[] auxTerms = terms.split("/");
4     String[] tempTerms;
5     int rule = 0;
6     double value;
7     int columns = this.matrixData[0].length;
8
9     for(int i = 0; i < auxTerms.length; i++) {
10        rule = findTerm(auxTerms[i]);
11
12        tempTerms = auxTerms[i].split(":");
13
14        for(int j = 0; j < tempTerms.length; j++) {
15            System.out.print(tempTerms[j]+);
16        }
17
18        value = Double.parseDouble(this.matrixData[rule][columns-1]);
19
20        if(option.equals("add")){
21            currentPrice += this.basePrice*value;
22        }else if(option.equals("rm")) {
23            currentPrice -= this.basePrice*value;
24        }
25    }
26    Res.set(currentPrice);
27 }

```

---

Fonte: Autoria Própria (2021).

#### 5.3.4 Criação do Log de Eventos

Este artefato é utilizado após cada evento realizado no estacionamento. Desde um agente *driver* perguntando o preço atual até o seu termino do usa da vaga, cada ação resulta no *manager* usando a operação deste artefato para salvar este evento em um *log*. Um trecho do arquivo de *log* criado após cada execução do sistema é mostrado na Figura 14. Os dados guardados no *log* foram escolhidos a fim de demonstrar funcionamento do sistema. Para isso são necessários os dados da vaga utilizada em cada evento (*Sector*, *SpotId* e *SpotType*), o agente e sua ação no sistema (*Agent* e *Description*), o momento em que o evento ocorreu (*WeekDay*, *Year*, *Date(M/D)* e *EventTime(H:M:S)*),

a regra utilizada (*Rule*), o preço atual (*CurrentPrice*), sua variação em cada momento (*PriceVariation(%)*) e a taxa de ocupação do estacionamento ao decorrer da simulação (*OccupationRate(%)*). Nesta Figura é observado que a linha 14 o momento em que o sistema realiza a alteração do preço exemplificada nas subseções 5.3.2 e 5.3.3.

**Figura 14 – Log de eventos criado durante execução do sistema usando o arquivo de configuração 6**

Sector	SpotId	SpotType	Agent	Description	WeekDay	Year	Date(M/D)	EventTime(H:M:S)	Rule	CurrentPrice	PriceVariation(%)	OccupationRate(%)
?	?	?	?	StartingOrganization	Segunda	2020	01/01	00:00:00	?	5	.00	0
?	?	?	?	OrganizationChanged	Segunda	2020	01/01	00:00:00	?	5	.00	0
?	?	?	driver71	PredictingPrice	Segunda	2020	01/01	00:50:50	?	5	.00	0
?	?	?	driver71	SpotBooked	Segunda	2020	01/01	01:22:20	?	5	.00	0
?	?	?	?	PriceChanged	Segunda	2020	01/01	03:01:50	Demanda:Baixa / CondClima:Sol / HoraDia:Madrugada/	3.5	-30.00	0
?	?	?	driver121	AskedPrice	Segunda	2020	01/01	03:05:30	?	3.5	.00	0
?	?	?	driver121	RequestedSpot	Segunda	2020	01/01	03:06:10	?	3.5	.00	0
A	5	Normal	driver121	SpotAllocated	Segunda	2020	01/01	03:07:10	?	3.5	.00	1
...												
?	?	?	driver99	AskedPrice	Sexta	2020	01/05	20:06:10	?	5	.00	71
?	?	?	driver99	RequestedSpot	Sexta	2020	01/05	20:06:30	?	5	.00	71
B	35	Reserved	driver99	SpotAllocated	Sexta	2020	01/05	20:07:40	?	5	.00	72
?	?	?	?	PriceChanged	Sexta	2020	01/05	20:30:20	Demanda:Alta / CondClima:Nublado / HoraDia:Noite/	6	20.00	72
A	1	Reserved	driver107	LeavingSpot	Sexta	2020	01/05	20:52:00	?	6	.00	71
A	45	Disabled	driver131	LeavingSpot	Sexta	2020	01/05	21:42:10	?	6	.00	70
B	17	Elderly	driver91	LeavingSpot	Sexta	2020	01/05	21:51:10	?	6	.00	69
B	49	Normal	driver97	LeavingSpot	Sexta	2020	01/05	22:59:20	?	6	.00	68
?	?	?	?	StartingOrganization	Sabado	2020	01/06	00:02:00	?	6	.00	68
?	?	?	?	OrganizationChanged	Sabado	2020	01/06	00:30:20	?	6	.00	68
A	31	Normal	driver98	SpotReserved	Sabado	2020	01/06	00:30:30	?	6	.00	68
A	6	Normal	driver133	SpotReserved	Sabado	2020	01/06	00:30:50	?	6	.00	68

Fonte: Autoria Própria (2021).

### 5.3.5 Controle da Fila de Espera

Em casos onde não há mais vagas livres, seja estacionamento lotado ou o tipo de vaga desejada indisponível, este artefato é usado para manter uma lista de espera dos agentes *drivers*. O algoritmo 20 apresenta a operação de inserção do *driver* na lista de espera. São registrados o nome, grau de confiança e tipo de vaga desejado do agente nesta lista. Após a saída de um *driver* do estacionamento, a operação no algoritmo 21 é o primeiro método utilizado para designar uma nova vaga para um agente, caso exista algum em espera. O primeiro *driver* a sair da lista e receber uma vaga é aquele com o maior grau de confiança.

#### Algoritmo 20 – Inserção de driver na lista de espera

```

1 @OPERATION
2 public void insertDriverQueue(Object idDriver, Object tDriver, Object tyDriver){
3     driver = new Driver(idDriver.toString(),
4         Double.parseDouble((tDriver.toString())),
5         tyDriver.toString(),new Date());
6     waitingQueue.add(driver);
7 }

```

Fonte: Autoria Própria (2021).

### Algoritmo 21 – Remoção de driver na lista de espera

---

```

1 @OPERATION
2 public void freeDriver(OpFeedbackParam<Object> idDriver, OpFeedbackParam<Object> tDriver,
   OpFeedbackParam<Object> tyDriver){
3     Driver d = greatestTrust();
4
5     idDriver.set(d.getId());
6     tDriver.set(d.getBackground());
7     tyDriver.set(d.getType());
8
9     waitingQueue.remove(d);
10 }

```

---

Fonte: Autoria Própria (2021).

#### 5.3.6 Simulador de Progressão do Tempo

As operações deste artefato são usadas como um meio de simular a ocupação de um *Smart Parking*. Em uma situação real este artefato não seria utilizado e existe com o intuito de reduzir o tempo de simulação. O tempo simulado progride mais rápido que o normal. Cada segundo de tempo real avança a simulação em aproximadamente 1 hora.

#### 5.4 Processo de Reorganização de Vagas

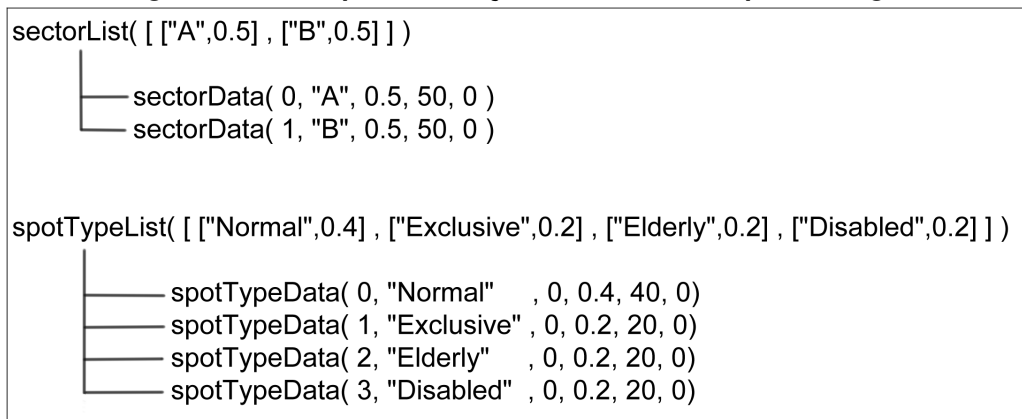
O processo de reorganização de vagas faz parte dos planos do agente *manager* e nesta seção o processo será discutido de forma mais detalhada.

Para realizar uma reorganização do estacionamento, é necessário estabelecer a organização inicial das vagas. No plano *CreateSpots* as crenças iniciais *sectorList* e *spotTypeList*, vistas na seção 5.2, são usadas para criar as crenças de dados dos setores e dos tipos de vagas. Para cada elemento dessas listas, uma crença que contém os dados estatísticos dos diferentes setores e tipos de vagas é criada. Nomeadas respectivamente como *sectorData* e *spotTypeData*, essas crenças contém a porcentagem de representação no estacionamento, o numero inteiro que essa porcentagem representa e quantas dessas vagas estão ocupadas no momento.

A Figura 15 mostra um exemplo de criação das crenças onde o estacionamento possui dois setores, A e B, cada um ocupando 50% do estacionamento e quatro tipos de vagas: *Normal* que representa 40% das vagas como normais; *Exclusive* representando 20% das vagas como exclusivas; *Elderly* atribuída 20% vagas para idosos; e *Disabled* definindo 20% das vagas como de deficientes. Essas porcentagens for escolhidas

de forma arbitrária considerando apenas indicar existência de vagas especiais. Neste exemplo é suposto um estacionamento de 100 vagas. Cada setor contém 50 e os tipos compõem respectivamente 40, 20, 20 e 20 dessas vagas. O terceiro elemento em cada *spotTypeData* mostra o número de requisições para o tipo e será utilizado durante a reorganização do estacionamento.

**Figura 15 – Exemplo de Crenças dos Setores e Tipos de Vagas**



**Fonte: Autoria Própria (2021).**

O plano *CreateSpots* continua por criar as crenças das vagas, inserindo qual setor cada um pertence usando as porcentagens encontradas em *sectorData*. Com as vagas criadas, o plano *OrganizeSpots* é iniciado. Neste plano as crenças *spotTypeData* são usadas como referência para atribuir e, futuramente, alterar os tipos das vagas. O algoritmo 22 mostra um exemplo de criação e atribuição de elementos nas crenças das vagas. A primeira linha representa o esqueleto da crença que contém o número identificador da vaga, setor que ela pertence, o seu tipo e o agente que a está usando. A segunda linha mostra um exemplo de crença após a finalização do plano *CreateSpots* em que ainda não possui um tipo de vagas atribuído. A terceira linha mostra a crença após a primeira chamada do plano *OrganizeSpots* que atribui tipos para todas as vagas. A quarta linha mostra essa crença ocupada por um *driver*.

**Algoritmo 22 – Exemplo de crenças das vagas**

---

```

1 spot(ID, Sector, Type, User)
2 spot(23, "B", "None", "Free")
3 spot(23, "B", "Normal", "Free")
4 spot(23, "B", "Normal", "Driver5")

```

---

**Fonte: Autoria Própria (2021).**

No começo de cada dia, o agente *manager* inicia o plano de reorganização da composição dos tipos de vagas no estacionamento: *ChangeOrganization*. Durante o

dia anterior, as requisições de cada tipo foram guardadas em suas respectivas crenças, após cada pedido de vaga realizado por um *driver*. O plano de reorganização começa somando e encontrando o total dessas requisições e, a partir deste resultado, são definidas as novas porcentagens de cada tipo de vaga. É estabelecido internamente que cada tipo de vaga deve ter pelo menos 10% de representação do total. Em ocasiões que isso não ocorre, blocos de 10% das vagas do tipo com maior representação são transferidos para os tipos que ainda não possuem essa porcentagem mínima.

Concluindo a atualização da proporção de tipo, o plano *OrganizeSpots* é reiniciado. A mudança nas vagas ocorre porque as crenças dos tipos de vagas foram alteradas durante o plano anterior.

## 5.5 Regras de Precificação

Parte integral deste módulo de precificação dinâmica encontra-se no arquivo de configuração. Neste sistema foi utilizado um arquivo csv, contudo o método de processamento do arquivo deixa em aberto a possibilidade de utilizar outras extensões. Neste trabalho foram consideradas apenas três categorias observadas no ambiente. Pelo método em que o arquivo de configuração foi desenvolvido, é possível facilmente adicionar novas características. O próprio arquivo de configuração é escalável. É possível a criação de  $n$  subdivisões para cada categoria e  $m$  linhas para cada subdivisão.

### 5.5.1 Arquivo de Configuração

O arquivo de configuração é uma tabela composta por duas sub-tabelas que definem as regras utilizadas durante a mudança dos preços das vagas. A construção deste arquivo foi realizada da seguinte forma: Primeiro é criada uma sub-tabela de legenda que contém o ID, a *String* e os intervalos de cada tipo de regra. Foram considerados três tipos categorias de regras aceitas nesta legenda: regras de demanda do estacionamento, horário do dia e condição atmosférica definidas com os IDs 1 á 3 respectivamente. Na legenda, qualquer combinação contendo no mínimo uma entre essa três categorias é suportada pelo módulo. Em seguida, separado por uma linha vazia, é construído uma sub-tabela que contém as categorias, suas condições e as porcentagem de mudança do preço para cada condição.

**Tabela 1 – Configuração com duas categorias**

1	ID	String	Intervalos
2	1	Demanda	0-0.3-0.6-0.8-1
3	3	Horario	0-6-12-15-18-24
4			
5	Categoria	Condição	Valor
6	Horario	Madrugada	-0,30
7	Horario	Manha	0,10
8	Horario	MeioDia	0,30
9	Horario	Tarde	0,20
10	Horario	Noite	-0,10
11	Demanda	Baixa	0
12	Demanda	Normal	0,10
13	Demanda	Alta	0,20
14	Demanda	Enorme	0,30

**Fonte: Autoria Própria (2021).**

**Tabela 2 – Configuração com três categorias**

1	ID	String	Intervalos
2	1	Dmd	0-0.4-0.7-1
3	2	CondClima	0-1-2
4	3	Hora	0-12-18-24
5			
6	Categoria	Condição	Valor
7	CondClima	Ensolarado	-0,10
8	CondClima	Nublado	0
9	CondClima	Chuvoso	0,05
10	Dmd	Baixa	0
11	Dmd	Media	0,10
12	Dmd	Alta	0,30
13	Hora	Manha	0
14	Hora	Tarde	0,10
15	Hora	Noite	-0,10

**Fonte: Autoria Própria (2021).**

Na sub-tabela de legendas, a coluna *ID* identifica o tipo da regra, onde *1* indica uma mudança por demanda; *2* por condição climática e *3* por horário do dia. A coluna *String* representa a palavra da categoria usada para cada *ID*. Essa palavra não precisa ser igual ao tipo de regra. A coluna *Intervalos* estabelece os valores para escolher entre as condições definidas em cada categoria usada. Na sub-tabela de regras, a primeira coluna contém as *Strings* das categorias escolhidas. A segunda contém as condições que especificam cada categoria. A terceira contém a porcentagem de mudança do preço em cada combinação categoria vs. condição.

Na tabela 1 por exemplo, são usadas apenas as categorias de demanda e horário. Na tabela 2 são usadas as de demanda, condição climática e horário. Os intervalos são usados como meio de conectar a situação do ambiente com as condições disponíveis na tabela de regras. Os valores dependem do contexto de cada tipo de categoria. Nas categorias de demanda são representados porcentagens e precisam conter os valores reais de 0 a 1 (0 a 100%). Valores de clima são identificadores de cada um observado no estacionamento. Por exemplo, a tabela 2 contém 0, 1 e 2 identificando 3 condições climáticas. Os intervalos de horário equivalem as horas do dia e devem conter de 00:00 a 24:00 que representam o começo e fim do dia respectivamente.

Cada linha na segunda tabela contém uma regra com uma das categorias usadas, uma condição em que o estacionamento deseja realizar uma mudança do preço nestas categorias e a porcentagem de quanto o preço mudará nesta condição. Por exemplo, a linha 9 da tabela 2 apresenta a categoria *CondClima*, condição *Chuvoso*

e incremento do preço em 5%. Observando a coluna de intervalos da tabela 1, uma demanda de 65% está entre 0.6 e 0.8 e logo se encaixa com a condição *Alta*. A combinação *Demanda Alta*, na linha 13, resulta em 20% do preço base somado ao valor atual das vagas. A mesma demanda de 65% encontra-se entre 0.4 e 0.7 na tabela 2 e se encaixa na condição *Media*. A combinação *Dmd Media* adiciona ao preço das vagas 10% do preço base.

Apenas uma regra de cada categoria pode estar ativa em simultâneo, e a porcentagem desta altera o valor da vaga usando o preço base, definido no cenário simulado. Antes de o preço ser alterado pela nova regra, o desconto ou aumento de uma regra anterior é removido.

$$\text{NovoPreço} = \text{PreçoBase} + \sum_{i=1}^n \text{Regra}_i * \text{PreçoBase}$$

Esta equação acima mostra como o preço é alterado em cada instancia de uso do módulo. Como mencionado anteriormente, o novo preço encontrado é inserido sobre o utilizado antes da chamada da operação. O preço base é determinado pelo gerente no início da execução e  $n$  é igual ao total de tipos de regras usadas.

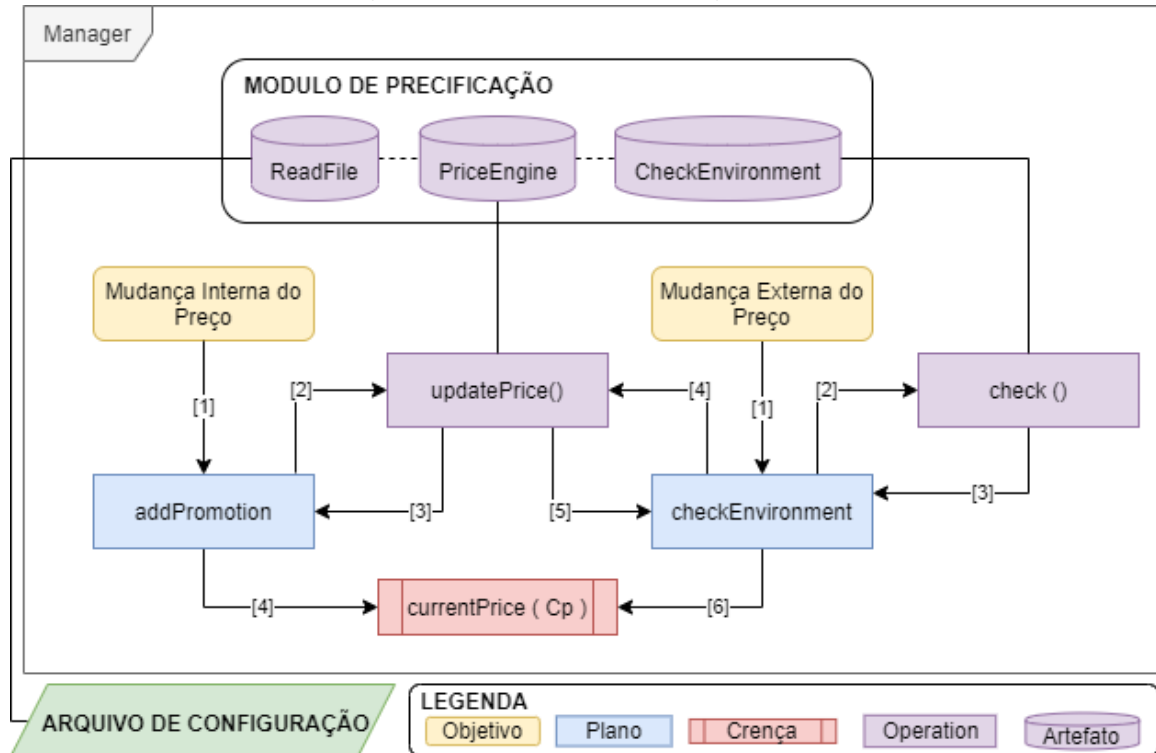
Como o valor base das vagas e o arquivo de configuração não mudam durante a simulação, os valores máximos e mínimos passíveis são determináveis no início da simulação. Estes valores são determinados considerando as regras do arquivo de configuração usado. Por exemplo, ao usar a Tabela 2 como configuração, o maior preço é encontrado ao ocorrer a combinação de regras *Hora:Tarde/Dmd:Alta/CondClima:Chuvoso*. Inserindo estas regras na equação e usando um preço base de 5 créditos, o seguinte cálculo e resultado é alcançado:  $(5 + (0,10 * 5) + (0,30 * 5) + (0,05 * 5)) = 7,25$  créditos. Usando a mesma tabela, o menor preço ocorre com o uso das regras *Hora:Noite/Dmd:Baixa/CondClima:Ensolarado*. Inserindo estas novas regras na equação e usando o mesmo preço base, o seguinte cálculo e resultado é alcançado:  $(5 + (-0,10 * 5) + (0 * 5) + (-0,10 * 5)) = 4$  créditos.

## 5.6 Processo de Precificação

O processo de precificação é realizado com a utilização dos artefatos de checagem do ambiente e o artefato de precificação que usa a *String* de regras formatada pelo artefato de checagem para alterar o preço das vagas. Essas regras, que contém as

porcentagens de quanto o preço deve ser modificado são definidas em um arquivo de configuração (subseção 5.5.1). Este arquivo revela como o Smart Parking em questão considera o efeito de mudanças do ambiente no preço de suas vagas.

**Figura 16 – Utilização do módulo de precificação pelo agente *manager***



Fonte: Autoria Própria (2021).

A Figura 16 ilustra o processo de utilização do módulo de precificação. A tomada de decisão de quando ocorrerá a mudança do preço é realizada pelo agente *manager*, o único agente que possui acesso aos três artefatos que compõem este módulo.

Para que o *manager* seja capaz de iniciar o processo de precificação é necessário a inicialização do arquivo de configuração por meio do artefato leitor (subseção 5.3.1). Neste artefato está contida a operação que identifica as configurações do arquivo especificado como argumento. O leitor captura as regras e dados e os insere em matrizes disponibilizadas para os outros artefatos do módulo.

Uma vez disponíveis as matrizes contendo as regras estabelecidas no arquivo de configuração, o *manager* é capaz de em qualquer momento recorrer ao artefato de checagem para verificar a situação atual do ambiente e retornar um conjunto de regras que melhor reflete o cenário. As propriedades do ambiente consideradas mudam de acordo com cada configuração, mas sempre são uma combinação entre a demanda,



horário e condição climática no estacionamento. A regra estabelecida neste artefato é retornada para o agente *manager*.

A regra criada pela observação do ambiente é usada pelo *manager* como argumento da operação do terceiro e último artefato. Cada pedaço desta *String* formatada, que pode conter entre 1 e 3 elementos, aponta para uma linha da tabela de regras do arquivo de configuração. Na última coluna desta linha está contida a porcentagem de mudança do preço para as combinações referidas nos elementos da regra.

#### 5.6.1 Mudança Interna do Preço

O caminho da esquerda da figura 16, começando pelo objetivo com nome da subseção, mostra que o agente *manager* pode iniciar alterações de preço de acordo com intenções próprias. Um exemplo é *addPromotion*, que envia uma regra de inclusão de promoção no valor das vagas. Essa regra deve estar incluída no arquivo de configuração, visto na subseção 5.5.1, por mais que não seja identificado no artefato de checagem do ambiente. Neste caminho, o plano *updatePrice* recebe uma regra como *Evento:Promocao* (Tabela 9) para procurar em sua matriz de regras a linha que contém essa *String*. Na última coluna desta linha, encontra-se a porcentagem de mudança do preço e o novo valor encontrado é retornado ao *manager* no fim da execução. O *manager* logo atualiza sua crença de preço atual das vagas *currentPrice* com o valor recebido da operação.

#### 5.6.2 Mudança Externa do Preço

No caminho da esquerda da figura 16, começando pelo objetivo com nome da subseção, é mostrado o processo completo do módulo de precificação. Ele começa com o agente *manager* iniciando o plano que concentra os dados sobre do ambiente (*checkEnvironment*). Durante as execuções do sistema, o cenário do estacionamento é verificado a cada 3 horas em tempo de simulação e logo este plano é iniciado em cada um destes momentos. A operação do artefato de checagem do ambiente (*check()*), descrito na subseção 5.3.2, identifica a situação atual do ambiente, que pode incluir dependendo do arquivo de configuração a demanda nas ultimas 3 horas, a condição climática e o horário atual e retorna a regra que reflete a situação. Uma possível condição do ambiente resulta na regra *Demanda:Normal/CondClima:Sol/HoraDia:Tarde* (Tabela

9). Essa regra é logo entregue ao plano *updatePrice* que usa as linhas contendo essas regras para alterar e retorna o novo preço de acordo com as porcentagens de sua matriz de regras criadas no leitor do arquivo de configuração (subseção 5.3.1). O novo valor é usado pelo *manager* para atualizar sua crença de preço das vagas *currentPrice*.

## 5.7 Exemplo do Processo Completo

A seguir encontra-se todas as etapas pertinentes a um exemplo de mudança do preço das vagas:

1. Inicialização dos artefatos do módulo de precificação (*PriceEngine* e *CheckEnvironment*);
2. Leitura do arquivo de configuração (neste caso 6 da tabela 9). Os dados são gravados no artefato de verificação do ambiente e de precificação;
3. Agente *manager* inicia a operação de verificação do ambiente enviando informações atuais do estacionamento. Por exemplo, relógio indicando 20:30, demanda em 72% e sensor 1 do clima ativado;
4. Esta operação verifica as informações pertinentes ao arquivo de configuração utilizado. A configuração 6 neste caso considera todas as três características;
5. A regra encontrada a partir destes dados é formatada e enviada de volta para o *manager*. Neste exemplo a regra é "*Demanda:Alta/CondClima:Nublado/HoraDia:Noite*";
6. Com essa regra, o *manager* primeiro retorna o valor de sua crença ao preço base e envia a nova regra para o artefato de precificação;
7. Na operação deste artefato a regra é desmembrada e o novo preço é alcançado utilizando as porcentagens do arquivo de configuração. Neste exemplo, a demanda alta incrementa 30% do preço base nele mesmo, o horário da noite reduz este valor por 10% do preço base e condição climática nublado não altera o preço. O novo preço é encontrado com a equação  $(5 + (0,3*5) + (0*5) + (-0,1*5))$  que resulta em 6 créditos;
8. O processo termina com o *manager* utilizando este valor em sua crença de preço das vagas.

## 5.8 Considerações Finais

Este capítulo apresenta o desenvolvimento do módulo de precificação e processo de reorganização de vagas do *Smart Parking* implementado. Foi descrito o SMA desenvolvido, suas variáveis e planos de seus agentes. Mostra-se como o agente *manager* é encarregado de gerenciar todo o estacionamento. Nele existem os planos que levam a alocação, reorganização e precificação de todas as vagas. O agente *driver* por sua vez contém sua função de simular apenas necessidades do motorista. Nele existe a capacidade de decidir horário de uso e tipo de vaga desejada, além de planos de requisição e consideração do preço de vagas.

O objetivo principal deste capítulo foi a descrição do módulo de precificação. A atualização do preço é realizada considerando uma combinação de três características do ambiente. Mudanças na demanda por vagas, condição climática e horário do dia são observadas para alterar o valor do preço. Como objetivo secundário, foi explicado o processo de reorganização de vagas. Neste é realizada uma alteração da composição do estacionamento de acordo com as necessidades do momento. Cada tipo de vaga é alterado usando sua demanda no último período observado.

## 6 RESULTADOS

Neste capítulo são mostrados os resultados encontrados após a execução de diversas simulações de um Smart Parking em diferentes cenários. Esses resultados são usados como auxílio na observação do funcionamento do módulo de precificação em um sistema fechado. A criação dos cenários foi realizada com o intuito de representar tipos e ocasiões diferentes de estacionamentos. Contudo, as características determinísticas do SMA utilizado podem não representar completamente como todo o processo ocorreria na realidade.

### 6.1 Cenários de Testes

Os cenários consistem de combinações entre diferentes valores de variáveis do sistema e arquivos de configuração para o módulo de precificação. Os períodos utilizados nas regras de demanda e de horário são baseados nos dados apresentados no trabalho de (PIERCE; SHOUP, 2013). Foram criados 6 arquivos de configurações distintos que representam maneiras pelas quais o preço pode ser alterado. Também foram considerados cenários sem usar o módulo. As variáveis do sistema alteradas foram o total de vagas no estacionamento; o número de agentes *drivers* criados no início da simulação; e conjuntos de preço base das vagas e créditos máximos de cada *driver*.

Foram determinados os valores de 100 e 200 vagas totais e 50, 100 e 200 *drivers* criados, visando comparar diferentes níveis de demanda base no estacionamento. Foram desenvolvidos os conjuntos de 5 e 10, 8 e 15, 10 e 20 de respectivos preços base das vagas e créditos máximos disponíveis aos *drivers* a fim de identificar a interação dos agentes com diferentes preços iniciais no estacionamento. As combinações dessas características geraram no total 70 cenários que foram executados no sistema cada um durante um mês (30 dias) em tempo de simulação.

**Tabela 3 – Variáveis e valores dos cenários de teste**

Variáveis	Valores						
	100	200					
Vagas Totais	100	200					
<i>Drivers</i> Criados	50	100	200				
Módulos	Nenhum	Conf. 1	Conf. 2	Conf. 3	Conf. 4	Conf. 5	Conf. 6
Preços Base	5	8	10				
Créditos Máximos	10	15	20				

**Fonte: Autoria Própria (2021)**

A seguir são apresentados os 6 arquivos de configuração desenvolvidos para a realização dos testes do módulo de precificação. Cada subseção descreve duas configurações semelhantes e como elas alteram o preço das vagas ao observar as características do ambiente. A funcionalidade de um arquivo de configuração é melhor detalhada na subseção 5.5.1.

#### 6.1.1 Arquivos de Configuração 1 e 2

Os arquivos de configurações a seguir representam estacionamentos que consideram apenas uma das características do ambiente como referência para a mudança do preço. Para esses arquivos foram escolhidas as características de demanda das vagas e horário no estacionamento. É implícito que em todas as alterações percentuais é utilizado o preço base no incremento ou redução no valor atual da vaga.

Foi definido para a configuração 1 na Tabela 4 que nos intervalos de demanda de 0 à 20%, o preço da vaga é reduzido em 10%; de 20 à 40%, é mantendo o preço atual; de 40 à 60%, é adicionando 15% ao valor da vaga; de 60 à 80%, é adicionando 20% ao valor da vaga; e de 80 a 100% de demanda o valor da vaga aumenta em 20%.

Para a configuração 2, vista na tabela 5, é considerado que nos períodos do dia entre 0 e 5 horas, O preço da vaga reduz em 30%; entre 5 e 12 horas é incrementado 10% ao preço da vaga; de 12 às 14 horas a vaga aumenta em 30%; de 14 às 18 horas o valor da vaga aumenta em 10%; e de 18 até o fim do dia o preço é reduzido em 15%.

**Tabela 4 – Configuração 1**

1	<b>ID</b>	<b>String</b>	<b>Intervalos</b>
2	1	Dmd	0-0.2-0.4-0.6-0.8-1
3			
4	<b>Categoria</b>	<b>Condição</b>	<b>Valor</b>
5	Dmd	MuitoBaixa	-0.10
6	Dmd	Baixa	0
7	Dmd	Normal	0.15
8	Dmd	Alta	0.20
9	Dmd	MuitoAlta	0.25

**Fonte: Autoria Própria (2021)**

**Tabela 5 – Configuração 2**

1	<b>ID</b>	<b>String</b>	<b>Intervalos</b>
2	3	Horario	0-5-12-14-18-24
3			
4	<b>Categoria</b>	<b>Condição</b>	<b>Valor</b>
5	Horario	Madrugada	-0.30
6	Horario	Manha	0.10
7	Horario	MeioDia	0.30
8	Horario	Tarde	0.10
9	Horario	Noite	-0.15

**Fonte: Autoria Própria (2021)**

#### 6.1.2 Arquivos de Configuração 3 e 4

Os estacionamentos que utilizam um dos arquivos de configuração a seguir observam a demanda e mais uma característica do ambiente para determinar o preço

das vagas. É implícito que em todas as alterações percentuais é utilizado o preço base no incremento ou redução no valor atual da vaga.

A configuração 3 na tabela 6, considera a demanda e o horário do dia no estacionamento. Para demandas de 0 a 40% o preço não é alterado; Entre 40 e 70%, o preço aumenta em 20%; e a partir de 70%, o valor das vagas aumenta em 30%. Entre as 0 e 6 horas, o preço é reduzido em 20%; das 6 às 12 horas, a vaga não tem desconto ou aumento; entre as 12 e 15 horas, o valor das vagas aumenta em 20%; de 15 até as 18 horas, o preço tem incremento de 10%; das 18 horas até o fim do dia, o preço das vagas tem um desconto de 10%.

Na configuração 4 na tabela 7 é considerado a demanda e a condição climática do estacionamento para alterar o preço das vagas. Em demandas de 0 a 30%, o preço é reduzido em 10%; em demandas entre 30 e 80%, o preço não aumenta nem diminui; durante demandas maiores que 80%, o preço das vagas aumenta em 10%. Quando está ensolarado, o id 0 é recebido e o valor das vagas aumenta em 15%; Em tempo nublado, o id 1 é recebido e o preço aumenta em 5%; Durante chuva, o id 2 é recebido e o preço das vagas é reduzido em 10%; Em meio a tempestades, o id 3 é recebido e o valor das vagas diminui em 25%.

**Tabela 6 – Configuração 3**

1	ID	String	Intervalos
2	1	PercDmd	0-0.4-0.7-1
4	3	HrDia	0-6-12-15-18-24
5			
6	<b>Categoria</b>	<b>Condição</b>	<b>Valor</b>
13	PercDmd	Baixa	0.0
14	PercDmd	Media	0.20
15	PercDmd	Alta	0.30
7	HrDia	Madrugada	-0.20
8	HrDia	Manha	0
9	HrDia	MeioDia	0.20
10	HrDia	Tarde	0.10
11	HrDia	Noite	-0.10

**Fonte: Autoria Própria (2021)**

**Tabela 7 – Configuração 4**

1	ID	String	Intervalos
2	1	Demanda	0-0.3-0.8-1
3	2	Temporal	0-1-2-3
4			
5	<b>Categoria</b>	<b>Condição</b>	<b>Valor</b>
6	Temporal	Sol	0.15
7	Temporal	Nublado	0.05
8	Temporal	Chuva	-0.10
9	Temporal	Tempestade	-0.25
10	Demanda	Baixa	-0.10
11	Demanda	Normal	0.00
12	Demanda	Alta	0.10

**Fonte: Autoria Própria (2021)**

### 6.1.3 Arquivos de Configuração 5 e 6

Nestes dois arquivos todas as três características do ambiente são observadas e usadas durante a alteração do preço das vagas dos estacionamentos que usam uma

dessas configurações. É implícito que em todas as alterações percentuais é utilizado o preço base no incremento ou redução no valor atual da vaga.

**Tabela 8 – Configuração 5**

1	ID	String	Intervalos
2	1	Dmd	0-0.4-0.7-1
3	2	Tmp	0-1-2
4	3	Hr	0-12-18-24
5			
6	Categoria	Condição	Valor
7	Tmp	Ensolarado	-0.1
8	Tmp	Nublado	0
9	Tmp	Chuvoso	0.05
10	Dmd	Baixa	0.0
11	Dmd	Media	0.20
12	Dmd	Alta	0.50
13	Hr	Manha	0.0
14	Hr	Tarde	0.05
15	Hr	Noite	-0.05

**Fonte: Autoria Própria (2021)**

**Tabela 9 – Configuração 6**

1	ID	String	Intervalos
2	1	Demanda	0-0.3-0.6-1
3	2	CondClima	0-1-2-3
4	3	HoraDia	0-6-12-15-19-24
5			
6	Categoria	Condição	Valor
7	HoraDia	Madrugada	-0.20
8	HoraDia	Manha	0
9	HoraDia	MeioDia	0.10
10	HoraDia	Tarde	0.00
11	HoraDia	Noite	-0.10
12	Evento	Promocao	-0.50
13	Demanda	Baixa	-0.20
14	Demanda	Normal	0.00
15	Demanda	Alta	0.30
16	CondClima	Sol	0.10
17	CondClima	Nublado	0.0
18	CondClima	Chuva	-0.05
19	CondClima	Tempestade	-0.10

**Fonte: Autoria Própria (2021)**

Para a configuração 5 na tabela 8 são considerados menos condições para cada característica e poucas regras de desconto. Para as categorias de demanda são considerados que de 0 a 40% o preço não aumenta ou diminui; de 40 a 70% o preço aumenta em 20%; e acima de 70% o valor aumenta em 50%. Em tempo ensolarado, o id 0 é recebido e o preço recebe um desconto de 10%; Em tempo nublado, o id 1 é recebido e o valor das vagas não aumenta ou diminui; Durante clima chuvoso, o id 2 é recebido e o preço aumenta em 5%. No período das 0 às 12 horas, o preço não aumenta ou diminui; das 12 às 18 horas, o preço aumenta em 5%; das 18 até o fim do dia, o preço das vagas é reduzido em 5%.

A configuração 6, na tabela 9, possui número maior de condições por categoria além de maior número de descontos. É estabelecido nesta configuração uma regra de promoção que não observada no ambiente e pode ser acionada pelo agente *manager* durante a execução dos cenários com esta configuração. Para a demanda entre 0 e 30%, o preço é reduzido em 20%; entre 30 e 60%, o valor das vagas não aumenta ou diminui; Em demanda acima de 60%, o preço das vagas aumenta em 30%. O recebido id 0 representa temporal ensolarado e resulta em aumento de 10% no preço; Receber o id 1 representa a condição nublada e o preço não aumenta ou diminui; Durante chuva,

é recebido o id 2 e o preço diminui em 5%; e durante tempestades, o id 3 é recebido e resulta em redução de 10% no valor das vagas. No horário de 0 até as 6 horas, o preço é reduzido em 20%; das 6 até o meio dia, o valor das vagas não aumenta ou diminui; durante a 12 até as 15 horas, o preço aumenta em 10%; das 15 às 19 horas, o preço das vagas também não movimenta; após as 19 horas, o valor é reduzido em 10%.

## 6.2 Resultados Obtidos

O objetivos dos dados a seguir é avaliar como a inclusão do módulo de precificação dinâmico altera o total de fundos recebidos e a identificar seu impacto na ocupação do estacionamento em comparação com seu desuso. Em cada execução dos cenários é criado um *log* de eventos (Subseção 5.3.4) que contém todas as operações realizadas no sistema. As informações deste *log* são usadas para as criações dos gráficos e no preenchimento da tabela de resultados.

**Tabela 10 – Tabela dos Conjuntos**

Conjunto	Preço Base	Crédito Máximo
Baixo	5	10
Médio	8	15
Alto	10	20

**Fonte: Autoria Própria (2021)**

A tabela 10 deve ser usada como uma legenda dos conjuntos usados nos resultados. A tabela 11 mostra os resultados da execução de todos os 70 cenários. Para cada um foi encontrado o total de créditos recebidos e a ocupação média do estacionamento durante um período de 30 dias de simulação.

Os créditos adquiridos nos cenários sem a utilização do módulo de precificação foram, em média, 1,61% maiores que os recebidos em cenários com o uso de um arquivo de configuração. Essa porcentagem é encontrada após comparar a soma dos créditos recebidos em todos os cenários de cada módulo com o total recebido nos cenários que não usam um módulo. Os cenários sem módulo e de conjunto baixo receberam 4,08% a mais de créditos; os de conjunto médio, 10,03% a mais; e nos de conjunto alto, os cenários sem módulo receberam, em média, -0.98% de créditos que os com uso de uma configuração.

O total recebido nas configurações 3 e 5 foram em média superiores do que os créditos recebidos nos cenários sem módulo. A configuração 3 tem valor total 3,64%



Tabela 11 – Resultados dos cenários simulados

Módulo	# de Vagas	# de Drivers	Conjunto	Crédito Recebido	Ocupação Média	
10*Sem módulo	5*100 vagas	50 drivers	Baixo	1115	15,49%	
		2*100 drivers	Baixo	2355	32,70%	
			Médio	3304	28,92%	
		2*200 drivers	Baixo	4525	61,47%	
	Alto		7350	56,00%		
	5*200 vagas	50 drivers	Baixo	1090	9,85%	
		2*100 drivers	Baixo	2250	14,85%	
			Médio	3400	14,77%	
		2*200 drivers	Baixo	4370	30,78%	
			Alto	7140	28,29%	
10*Configuração 1		5*100 vagas	50 drivers	Baixo	1045	15,48%
	2*100 drivers		Baixo	2270,5	39,03%	
			Médio	3191,2	29,99%	
	2*200 drivers		Baixo	5000,75	67,57%	
		Alto	7553,5	55,98%		
	5*200 vagas	50 drivers	Baixo	931,5	11,27%	
		2*100 drivers	Baixo	2096,5	15,58%	
			Médio	2944	15,80%	
		2*200 drivers	Baixo	4679,75	32,88%	
			Alto	7090	25,21%	
		10*Configuração 2	5*100 vagas	50 drivers	Baixo	1107
	2*100 drivers			Baixo	2016,75	38,31%
				Médio	3397	28,57%
	2*200 drivers			Baixo	4275,75	58,84%
Alto			7350	45,40%		
5*200 vagas	50 drivers		Baixo	1144	5,76%	
	2*100 drivers		Baixo	2285,5	19,04%	
			Médio	2990	12,79%	
	2*200 drivers		Baixo	4421,5	35,16%	
			Alto	7105,5	27,56%	
	10*Configuração 3		5*100 vagas	50 drivers	Baixo	1044
2*100 drivers				Baixo	2210,5	29,97%
		Médio		3200	29,20%	
2*200 drivers		Baixo		5168	68,19%	
		Alto	<b>8070</b>	51,42%		
5*200 vagas		50 drivers	Baixo	1066	9,21%	
		2*100 drivers	Baixo	2164,5	17,34%	
			Médio	3256	14,00%	
		2*200 drivers	Baixo	4573,5	25,53%	
			Alto	7489	28,18%	
		10*Configuração 4	5*100 vagas	50 drivers	Baixo	910,5
2*100 drivers				Baixo	2185,25	36,70%
	Médio			3065,2	31,98%	
2*200 drivers	Baixo			4339,25	<b>69,17%</b>	
	Alto		7239,5	66,05%		
5*200 vagas	50 drivers		Baixo	965	10,26%	
	2*100 drivers		Baixo	1995,25	12,26%	
			Médio	2910,8	17,78%	
	2*200 drivers		Baixo	4107	26,20%	
			Alto	7234,5	28,33%	
	10*Configuração 5		5*100 vagas	50 drivers	Baixo	1111,25
2*100 drivers				Baixo	2575,5	41,70%
		Médio		3130,8	31,06%	
2*200 drivers		Baixo		5389	56,09%	
		Alto	7317,5	52,51%		
5*200 vagas		50 drivers	Baixo	1139,75	8,21%	
		2*100 drivers	Baixo	1999,25	12,85%	
			Médio	3434	15,61%	
		2*200 drivers	Baixo	4590,25	27,16%	
			Alto	7673	21,64%	
	10*Configuração 6	5*100 vagas	50 drivers	Baixo	899,25	18,65%
2*100 drivers			Baixo	2073	36,94%	
			Médio	3193,6	35,31%	
2*200 drivers			Baixo	4992	60,53%	
		Alto	7270,5	57,32%		
5*200 vagas		50 drivers	Baixo	<i>845,5</i>	8,82%	
		2*100 drivers	Baixo	1732,75	14,13%	
			Médio	2265,2	18,04%	
		2*200 drivers	Baixo	4030	30,93%	
			Alto	6585	33,57%	

Fonte: Autoria Própria (2021)

maior enquanto a configuração 5 possui valor total 3,96% maior. A configuração 1 teve em geral os mesmos créditos recebidos ficando apenas 0,26% abaixo do total dos créditos sem módulo equivalentes. As configurações 2, 4 e 6 tiveram seus valores totais abaixo dos cenários sem módulo. Os piores sendo da configuração 6 com -8,89% de créditos seguido das configurações 4 e 2 com respectivamente -5,57% e -2,23% de créditos totais recebidos em relação aos cenários equivalentes sem o uso do módulo de precificação.

Em comparações usando apenas os cenários com módulo de precificação, aqueles que utilizaram as configurações 3 e 5 receberam o maior número de créditos ganhando em média 6,36% e 6,76% a mais que outras configurações respectivamente. Considerando apenas os cenários de conjunto baixo e médio, a configuração 5 recebeu 10,72% e 9,48%, respectivamente, a mais de créditos recebidos que as outras. Para os cenários de conjunto alto, a configuração 3 foi superior com 7,53% de créditos acima de outras configurações.

De todos os dados encontrados, o cenário com o maior número de créditos recebidos foi o de configuração 3, 100 vagas, 200 *drivers* e de conjunto alto com 8070 créditos. A menor quantidade de créditos ocorreu no cenário de configuração 6, 200 vagas, 50 *drivers* e de conjunto baixo com apenas 845,5 créditos recebidos. A média de créditos recebidos, incluindo os cenários sem uso de módulo, foi de 3646,22 créditos. Considerando apenas os cenários de cada conjunto, as médias foram: 2597,28 para os de conjunto baixo; 3120,13 para os de conjunto médio; e 7319,14 para os de conjunto alto.

Observando as ocupações médias nos resultados, os cenários com o uso de uma configuração tiveram, em geral, preenchimento maior das vagas dos estacionamentos comparados com os sem uso do módulo. Olhando todos os resultados equivalentes, os cenários com uma configuração alcançaram 2,09% a mais de ocupação média. Ambos os cenários com conjuntos baixos e médios tiveram ocupação maior com 0,31% e 5,31% respectivamente mais que sem o uso do módulo. Apenas nos cenários com conjunto alto a falta do módulo foi favorecida com 4,23% maior ocupação média para os cenários sem configuração.

Entre as simulações usando um módulo de precificação, as configurações 1 e 4 se destacaram alcançando respectivamente 3,77% e 7,15% maiores ocupações médias que outras configurações. Considerando apenas os cenários com conjunto baixo, a

configuração 1 foi a melhor com 10,77% maior ocupação média neste quesito. Para aqueles com conjunto médio e alto, as configurações 4 e 6 se destacaram alcançando 11,26% e 18,75% respectivamente maior ocupação nos cenários médios e 16,41% e 18,38% respectivamente maior ocupação média nos cenários de conjunto alto.

Dos cenários observados, o de configuração 4, 100 vagas, 200 *drivers* e conjunto baixo foi o que alcançou a maior ocupação média com 69,17% de ocupação. O cenário com a menor ocupação média foi aqueles com a configuração 2, 200 vagas, 50 *drivers* e conjunto baixo com apenas 5,76% de ocupação. A média de todos os cenários neste quesito foi de 29,90%, com médias específicas para cada conjunto em 28,38% em conjuntos baixos; 23,13 em conjuntos médios; e 41,25% de média das ocupações médias no cenários de conjunto alto.

A seguir serão apresentados os resultados das simulações em cada dia de execuções dos cenários. Os gráficos a seguir mostrarão comparações entre uso dos arquivos de configurações em um mesmo estacionamento. Em todos os gráficos serão mostrados as taxas de ocupação, começando em zero no primeiro dia, e o preço atual das vagas em cada dia. O objetivo destes gráficos é identificar como a ocupação e preço das vagas do estacionamento foram alteradas ao decorrer da simulação em diferentes arquivos de configuração. São comparados o funcionamento do estacionamento sem o uso do módulo de precificação e entre as configurações em si.

#### 6.2.1 Cenários com 100 vagas e 50 *drivers*

O foco nos cenários de 100 vagas e 50 *drivers* é observar um estacionamento com quantidade média de vagas as quais são utilizadas por poucos motoristas. Todas as vagas iniciam a simulação vazias e os *drivers*, criados ao mesmo tempo, decidem aleatoriamente em que dia as usarão (ver Subseção 5.2).

O Gráfico 1 mostra os cenários com uso de conjunto baixo: preço base da vagas em 5 créditos e crédito máximo dos motoristas em 10 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 5 créditos, o maior valor aceito por um *driver* é de 10 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 29% e ocorreu com o uso da configuração 4 no dia 27. As

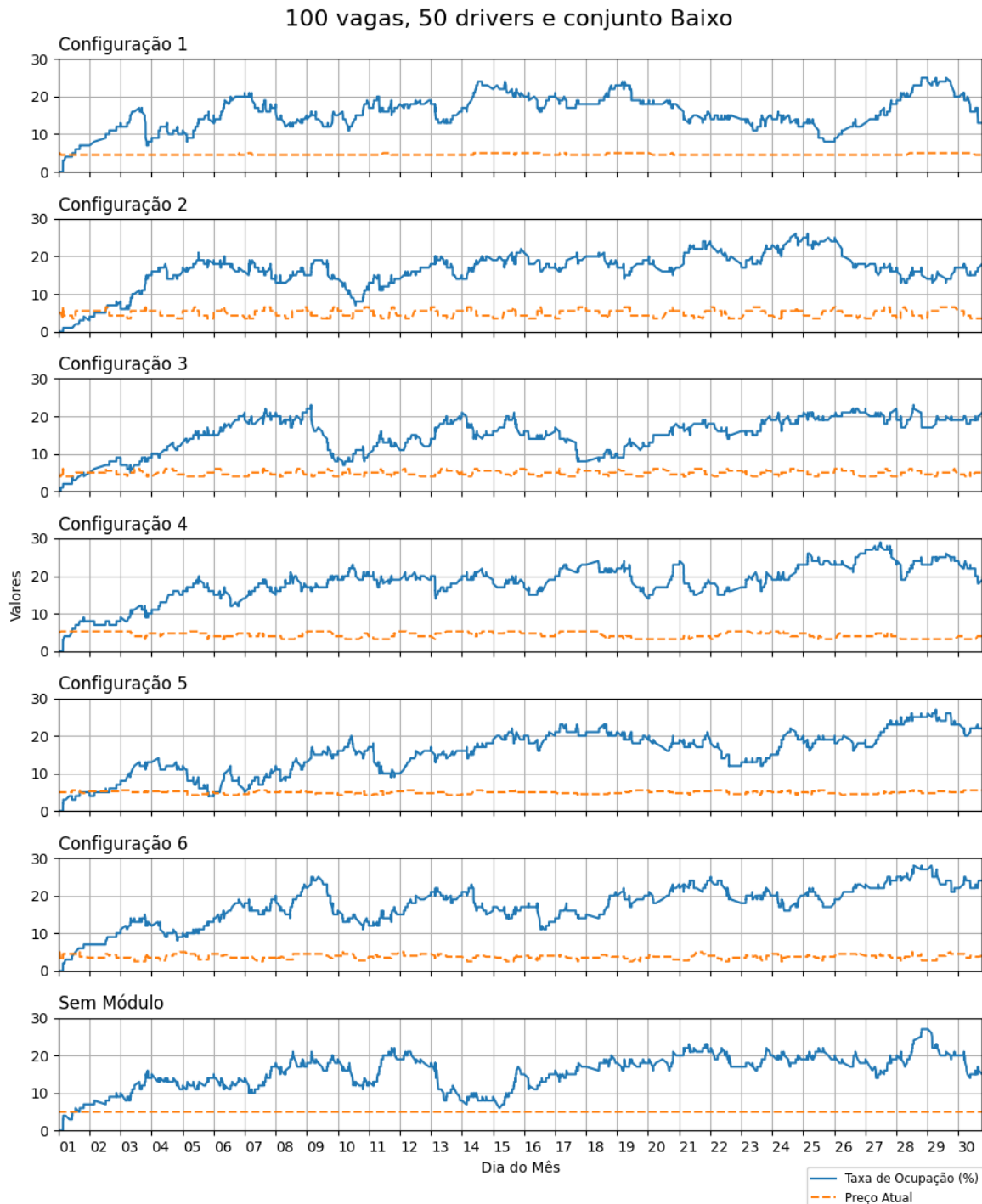
menores ocupações, após o primeiro dia, encontram-se na configuração 5 nos dias 2 e 5 em 4% de ocupação. Dos gráficos que são influenciados pela demanda, apenas a configuração 1 considera uma taxa menor que 30% em suas regras de mudança de preço. Contando com isso, o maior preço entre os gráficos encontra-se no de configuração 2 que alcançou 6,5 créditos durante o período do meio-dia. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 2,5 créditos. este preço das vagas foi alcançado especificamente durante tempestades de madrugada.

A configuração 1 possui a menor ocupação média entre os gráficos em 15,48%. Essa média é -0,05% menor que o cenário sem módulo, o qual possui a segunda pior ocupação média. Durante o mês, a taxa de ocupação permanece instavelmente no intervalo de 10 a 20% com subidas no dias 14 a 16 e de 19 a 20. Uma queda de ocupação, começando no dia 20, leva a uma taxa abaixo de 10% no dia 25 seguido pelo maior crescimento do cenário no fim do mês. A maior taxa de ocupação alcançada foi de 25% nos dias 28 e 29 e a menor depois do primeiro dia foi de 7% no dia 3. O gráfico de preço apresenta a influência da demanda nos poucos dias que a ocupação alcança ou ultrapassa o intervalo de 20% estabelecido nas regras do arquivo de configuração. O total de créditos recebidos neste cenário foi 1045, valor -6,7% menor que o total do sem módulo. O maior preço de 5 créditos encontra-se no dias de demanda baixa e o menor valor de 4,5 créditos ocorre nos dias restantes de demanda muito baixa.

A configuração 2 mostrou-se no meio da colocação tanto em preço recebido como em ocupação média. A ocupação está frequentemente ao redor de 20%, ocorrendo uma queda do dia a 13, e possui uma média de 17,08%, 10,27% maior que a de sem módulo. Este cenário é o único com taxas abaixo de 20% em toda a última semana do mês. A maior taxa de ocupação neste cenário foi de 26% nos dias 24 e 25 e a menor, depois do primeiro dia, foi de 7% nos dias 3 e 10. No seu gráfico de preço é mostrado a mudança de valor periódica que ocorre quando este é afetado apenas pela hora do dia. Neste cenário foi adquirido 1107 créditos, -0,72% menor que o valor do sem módulo. O maior preço das vagas atingido é de 6,5 sempre ao meio dia e o menor preço foi de 3,5 nas madrugadas.

Na configuração 3 possui, ao fim do mês, uma ocupação média de 15,51%, 0,14% maior que a de sem módulo. O gráfico de ocupação mostra grande variação nas taxas nos primeiros dois terços do mês e uma última semana estável em 20%. A maior taxa de ocupação neste cenário foi de 23% no dia 9 enquanto a menor taxa

**Gráfico 1 – Ocupações e preços em 100 vagas, 50 drivers e conjunto Baixo**



**Fonte: Autoria Própria (2021).**

foi de 5% no dia 3. O total de créditos adquiridos foi 1044, valor -6,8% menor do que no sem módulo. O preço das vagas não é influenciado pelas regras de demanda ao nunca alcançar 40% de ocupação, necessário para atingir o segundo intervalo. O maior preço durante a simulação neste cenário foi de 6 créditos e ocorreu durante o meio dia enquanto o menor foi de 4 créditos durante a madrugada.

A configuração 4 mostra o gráfico mais estável em questão de taxa de ocupação, circulando todo o mês a taxa de 20%. Novamente as regras de demanda não alteram o preço devido a ocupação não alcançar 30% de taxa de ocupação. A ocupação média de 18,51% deste cenário e é 19,51% maior que a de sem módulo. A maior taxa de ocupação durante a simulação foi de 29% no dia 27 e a menor taxa depois do primeiro dia foi de 7% no dia 2. O gráfico de preços apresenta os preços de cada dia de acordo apenas com o clima simulado. Durante a execução deste cenário foram recebidos 910,5 créditos no total, -22,46% menor que no do sem módulo. O maior valor da vaga atingiu 5,25 créditos durante clima ensolarado e o menor preço de 3,25 durante tempestades.

A configuração 5 contém o pior início de ocupação e o único que alcança taxa menor que 5% após o primeiro dia. Apesar disso, ao fim da primeira semana, este cenário vê um crescimento em sua ocupação e um preço atual estável em que o total recebido perde apenas ao comparar com sem uso do módulo. O resultado é uma ocupação média de 15,96%, valor 3,04% maior do que no sem módulo e a maior taxa de ocupação alcançou 27% no dia 29. Neste cenário o total de créditos recebidos foi 1111,25 e -0,34% menor que o valor do sem módulo. O maior preço das vagas neste cenário foi de 5,5 créditos em tempo chuvoso no horário da tarde. O menor preço das vagas foi de 4,25 créditos durante a noite com céu limpo.

Os gráficos da configuração 6 mostram a maior ocupação média e o menor total de créditos recebidos, 18,65% e 899,25 respectivamente. A maior taxa de ocupação alcançada foi de 28% nos dias 28 e 29 e a menor, após o primeiro dia, foi de 7% no dia 2. O gráfico de preços mostra a influência do valor das porcentagens de alteração do preço nas regras. Na execução deste cenário foram recebidos 899,25 créditos, valor -24% menor que o total do sem módulo. O maior preço durante a simulação foi de 5 créditos e ocorre durante o meio dia em clima ensolarado. Essa regra específica ocorreu apenas nos dias 10 e 21 do mês. O menor preço das vagas foi de 2,5 durante tempestades de madrugada.

O cenário sem módulo mostra o característico gráfico de preços sem alterações durante os dias. O gráfico de ocupação mostram taxas frequentemente abaixo de 20% acentuadas por um grande queda no meio do mês. Este cenário é o segundo pior em questão de ocupação, com média de 15,49%, mas possui o maior total de créditos recebidos em 1115. A maior taxa de ocupação foi de 27% no dia 28 e a menor ocupação depois do primeiro dia foi de 6% no dia 15. Ao fim de sua simulação, este cenário adquiriu

um total de 1115 créditos, o maior valor dentre as simulações equivalentes.

### 6.2.2 Cenários com 100 vagas e 100 *drivers*

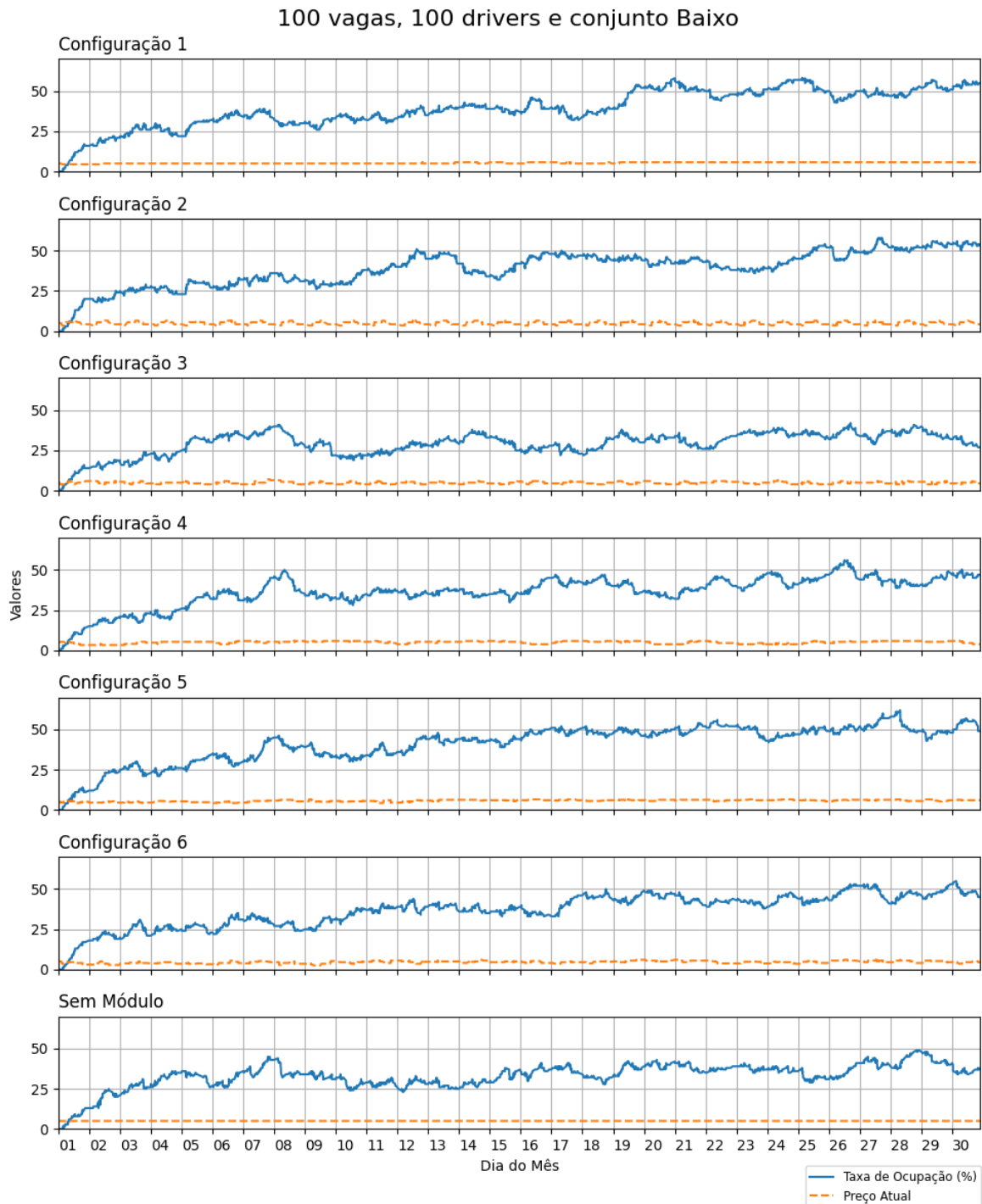
O foco nos cenários de 100 vagas e 100 *drivers* é observar um estacionamento com quantidade média de vagas as quais são utilizadas por um número médio de motoristas. Todas as vagas iniciam a simulação vazias e os *drivers*, criados ao mesmo tempo, decidem randomicamente em que dia as usarão (ver Subseção 5.2).

O Gráfico 2 mostra os cenários com uso de conjunto baixo: preço base da vagas em 5 créditos e crédito máximo dos motoristas em 10 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 5 créditos, o maior valor aceito por um *driver* é de 10 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 62% e ocorreu com o uso da configuração 5 no dia 28. A menor ocupação, após o primeiro dia, encontram-se na configuração 5 e no gráfico sem uso de módulo no dia 2 em 8% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 3 que alcançou 7 créditos durante o período de demanda média no horário de meio-dia. Tal regra ocorreu apenas no dia 7 durante a simulação do cenário, já que em todos outros dias a demanda é baixa durante o meio-dia. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 2,5 créditos. este preço das vagas foi alcançado especificamente em um período de demanda baixa, de madrugada durante uma tempestade no dia 9.

A ocupação média na configuração 1 foi 39,03%, segunda melhor média e 19,36% maior que a de cenário sem módulo. A maior taxa de ocupação durante a simulação foi de 58% nos dias 20, 24, 25 e 29. A menor taxa após o primeiro dia foi de 14% no dia 2. A escala do gráfico de ocupação reduz a clareza do gráfico de preço, porém ainda é observável a influência da demanda. Começando na metade do dia 2, a ocupação ultrapassa 20% e o desconto no preço é removido. A partir do fim do dia 13 até o começo do dia 19, a ocupação circula 40% e o preço muda de acordo. Do dia 19 em frente a demanda segue dentro do intervalo da regra normal e o preço das vagas permanece em seu valor mais alto de 6 créditos. O menor valor das vagas, 4,5 créditos, ocorre apenas no início da simulação durante o período de demanda muito baixa. O

**Gráfico 2 – Ocupações e preços em 100 vagas, 100 drivers e conjunto Baixo**



**Fonte: Autoria Própria (2021).**

total adquirido em toda a simulação foi de 2270,5 créditos, valor -3,72% menor do que no sem módulo.

Na simulação do cenário com configuração 2, a ocupação média foi de 38,31% a qual é 17,14% maior que a da sem módulo. Durante a execução, a maior taxa de ocupação foi de 58% no dia 27 e a menor, após o primeiro dia, foi de 18% no dia 2. O



gráfico de preço mostra a mudança periódica quando considerado apenas a hora do dia. Foram recebidos 2016,75 créditos ao fim do mês, total -16,77% menor que o valor do sem módulo e o menor total entre essas simulações equivalentes. O maior preço atinge 6,5 créditos ao meio dia e o menor preço é de 3,5 créditos durante a madrugada.

A configuração 3 possui neste cenário a pior ocupação média e o único gráfico com queda de ocupação no fim do mês. Sua ocupação média de 29,97% é a única pior que a de sem módulo estando 9,12% menor. Na execução deste cenário, a maior taxa de ocupação alcançada foi de 42% no dia 26 e a menor após o primeiro dia foi de 15% no dia 2. Neste cenário, o total de créditos recebidos foi 2210,5 que é -6,54% menor que o total do sem uso do módulo. O maior preço das vagas alcançado foi 7 créditos nos períodos de demanda média ao meio-dia e o menor preço atinge 4 créditos durante demanda baixa de madrugada.

A média das taxas de ocupação na configuração 4 foi 36,68%, maior que a do sem módulo por 12,21%. A maior taxa de ocupação foi de 56% no dia 26 e a menor após o primeiro dia foi de 18% no dia 2. O total de créditos adquiridos foram 2185,25 e -7,76% menor que o total do cenário sem módulo. O maior valor das vagas durante a simulação foi 5,75 créditos quando a demanda estava em nível normal e o clima ensolarado. O menor preço foi 3,25 créditos nos períodos de demanda baixa durante tempestades.

A configuração 5 possui a maior média de ocupação em 41,70% e créditos recebidos em 2575,5, 27,53% e 9,36% respectivamente maiores que os do sem uso de módulo. A maior taxa de ocupação foi de 62% no dia 28 e a menor, após o primeiro dia, foi de 11% no dia 2. O maior preço das vagas atingiu 6,5 créditos no período de demanda média, durante chuva ao meio-dia. O menor preço das vagas foi 4,25 créditos durante demanda baixa a noite com céus limpos.

A ocupação média na configuração 6 foi de 36,94%, 12,96% maior que a média do cenário sem módulo. A maior ocupação foi 55% no dia 30 e, após o primeiro dia, a menor ocupação foi 17% no dia 2. Neste cenário foram recebidos um total de 2073 créditos. O maior preço alcançou 6 créditos nos períodos ensolarados ao meio-dia em média demanda. O menor preço foi de 2,5 créditos e ocorre durante uma tempestade de madrugada em demanda baixa.

O cenário sem módulo possui o característico gráfico de preço sem alterações ao decorrer dos dias. Junto com a configuração 3, este cenário não alcança ocupação

de 50%, além de queda abaixo de 25% durante o mês. A sua média de ocupação foi 32,70%, segunda pior média na frente da configuração 3, com a maior taxa em 49% no dia 28 e menor em 14% no dia 2. O total de créditos adquiridos foi 2355, segundo melhor total atrás da configuração 5.

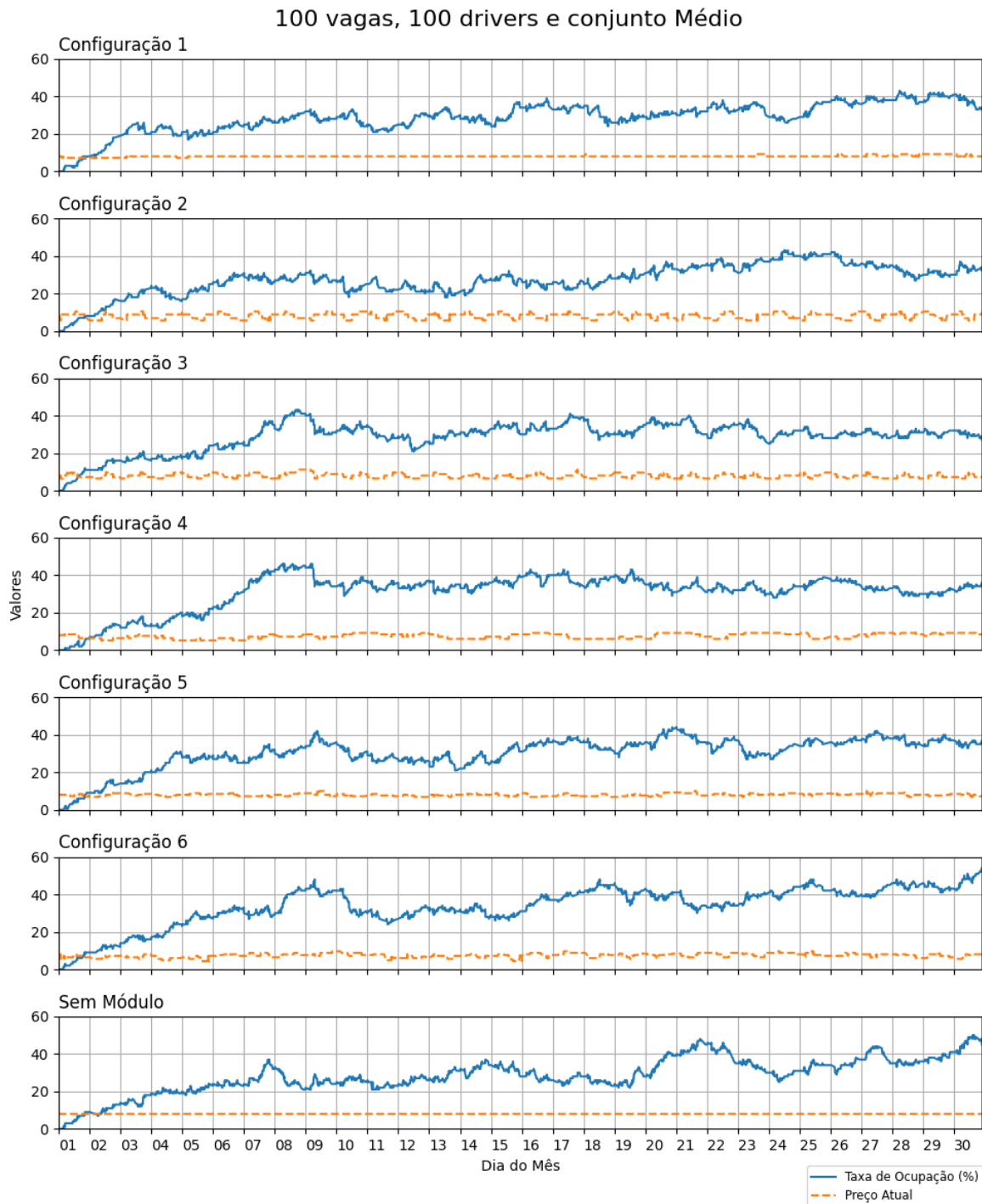
O Gráfico 3 mostra os cenários com uso de conjunto médio: preço base da vagas em 8 créditos e crédito máximo dos motoristas em 15 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 8 créditos, o maior valor aceito por um *driver* é de 15 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 54% e ocorreu com o uso da configuração 6 no dia 30. A menor ocupação, após o primeiro dia, encontram-se na configuração 1, 4 e no gráfico sem uso de módulo no dia 2 em 7% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 3 que alcançou 11,2 créditos durante o período de demanda média no horário de meio-dia. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 4,4 créditos. este preço das vagas foi alcançado especificamente em um período de demanda baixa, em clima chuvoso de madrugada.

A configuração 1 possui uma média de ocupação igual a 29,99%, 3,69% maior que a média do cenário sem módulo de precificação. Durante a execução, a maior taxa de ocupação foi 43% no dia 28 e a menor foi 7% no dia 2 desconsiderando o primeiro dia. O total de créditos acumulados foi 3191,2, uma vez que o maior preço foi de 9,2 créditos nos dias de demanda normal e o menor preço foi de 7,2 créditos em dias de demanda muita baixa. O gráfico de preços atuais mostra a influencia da demanda do estacionamento no preço da vagas. As modificações mais frequentes acontecendo no fim do mês quando a demanda flutua em 40%, valor inicial do intervalo considerado como demanda normal.

A ocupação média na configuração 2 foi de 28,57%, a pior média dentre os cenários do Gráfico e -1,23% menor que a média do cenário sem módulo. A maior taxa de ocupação ao longo da simulação foi 43% e a menor taxa, desconsiderando o primeiro dia, foi 10% no dia 2. O total de créditos recebidos neste cenário, o melhor entre os da Figura, foi 3397, valor 2,81% maior do que no sem módulo. O preço mais alto foi de 10,4 créditos ao meio-dia e o preço da vagas mais baixo foi de 5,6 créditos nas madrugadas.

**Gráfico 3 – Ocupações e preços em 100 vagas, 100 drivers e conjunto Médio**



**Fonte: Autoria Própria (2021).**

A configuração 3 possui 29,20% como média de ocupação, total 0,97% maior que a média do cenário sem módulo. A maior taxa de ocupação atinge 43% no dia 9 e a menor taxa chega a 14% no dia 2 ao desconsiderar o primeiro dia. Ao decorrer da simulação, foram acumulados 3200 créditos, -3,25% menor que o total do cenário sem módulo. O maior preço foi 11,2 créditos e ocorreu nos períodos de demanda média ao

meio-dia e o menor foi 6,4 créditos e ocorre durante demanda baixa de madrugada.

A média das taxas de ocupações na configuração 4 foi igual a 31,98%, 10,57% maior que a da sem módulo. Dentre as taxas de ocupação, a maior foi de 46% nos dias 8 e 9 e a menor foi 7% no dia 2, desconsiderando o primeiro dia. A soma de todos as requisições por vaga resultou em 3056,2 créditos no fim da simulação deste cenário. este total é -7,80% menor que o do sem módulo e é o pior resultado entre os cenários. O maior preço alcançando 9,2 créditos durante demanda normal e tempo ensolarado e o menor foi a 5,2 créditos nos momentos de demanda baixa durante tempestades.

A configuração 5 apresenta um média das ocupações igual a 31,06% e é 7,41% maior que a média do sem módulo. A maior taxa de ocupação é igual a 44% nos dias 9, 20 e 21 e a menor taxa é igual a 10% no dia 2. O total de créditos adquiridos foi 3130,8 e -5,53% menor que o do sem módulo. O maior preço durante a execução do cenário foi igual a 10,4 créditos no período de demanda média, em clima chuvoso a tarde especificamente no dia 9. O menor preço foi igual a 6,8 créditos e ocorreu em momentos de baixa demanda e de noite com céu limpo.

A configuração 6 possui uma média das ocupação de 35,31% que é maior média dentre as configurações e 22,11% maior que a média do cenário sem módulo. A maior taxa de ocupação foi de 54% no dia 30 e a menor foi de 10% no dia 2, desconsiderando o primeiro dia. Neste cenário foram recebidos 3193,6 créditos, -3,46% menor que o total do cenário sem módulo. O maior preço durante a simulação foi 9,6 créditos nos momentos de demanda normal em tempo ensolarado ao meio dia. O menor preço foi de 4,4 créditos durante chuva de madrugada meio a demanda baixa.

O cenário sem módulo mostra um gráfico de preço e taxas de ocupações nos quais o ambiente não possui influência. A ocupação média neste cenário foi 28,92%, superior ao resultado da configuração 2. A maior taxa de ocupação foi de 50% no dia 30 e a menor, depois do primeiro dia, foi de 7% no dia 2. Ao fim da execução este cenário acumulou 3304 créditos, segundo maior valor atrás do total recebido na configuração 2.

### 6.2.3 Cenários com 100 vagas e 200 drivers

O foco nos cenários de 100 vagas e 200 *drivers* é observar um estacionamento com quantidade média de vagas as quais são utilizadas por um número alto de motoristas. Todas as vagas iniciam a simulação vazias e os drivers, criados ao mesmo tempo,

decidem randomicamente em que dia as usarão (ver Subseção 5.2).

O Gráfico4 mostra os cenários com uso de conjunto baixo: preço base da vagas em 5 créditos e crédito máximo dos motoristas em 10 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 5 créditos, o maior valor aceito por um *driver* é de 10 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

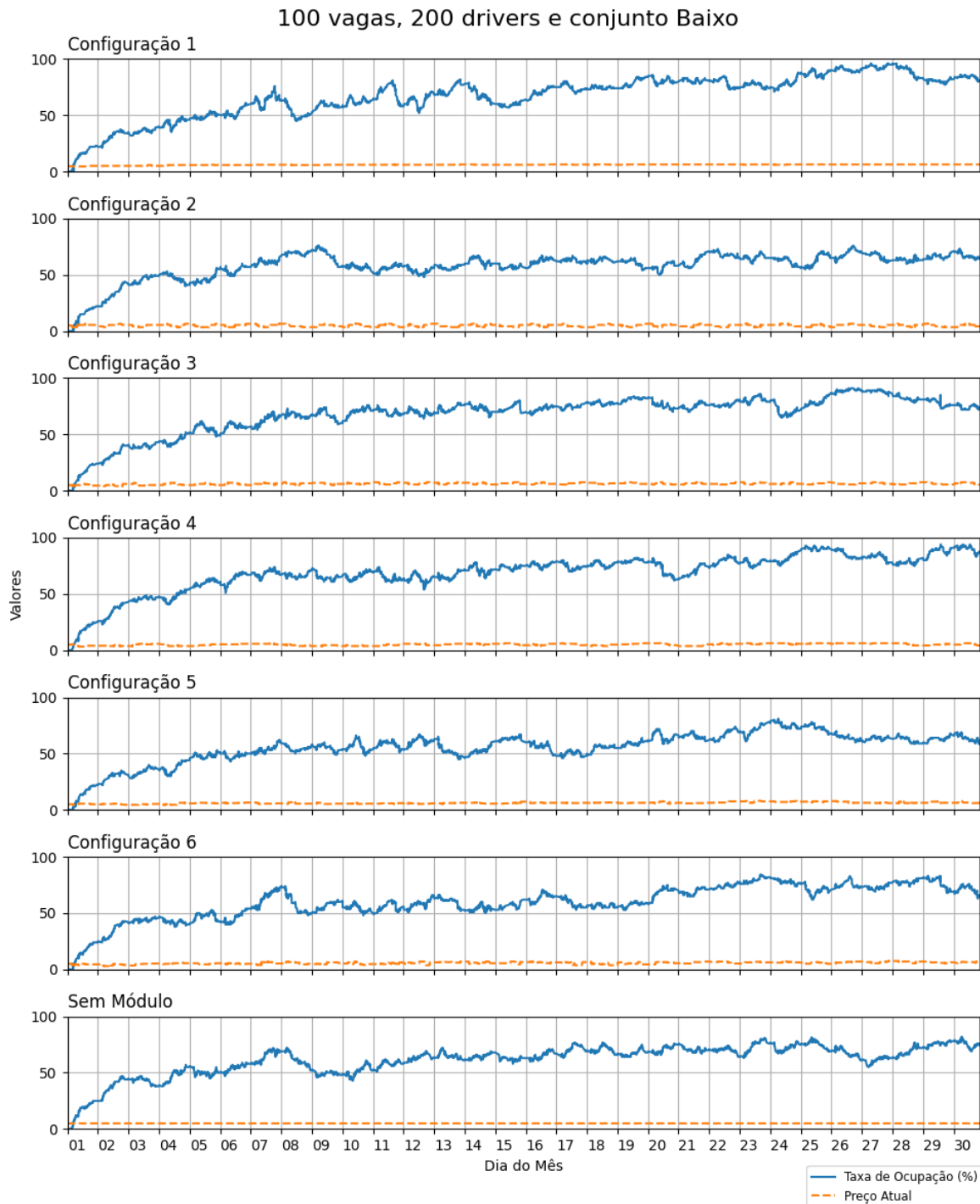
Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 96% e ocorreu com o uso da configuração 1 nos dias 27 e 28. A menor ocupação, após o primeiro dia, encontram-se na configuração 2 no dia 2 em 17% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 5 que alcançou 8 créditos durante o período de demanda alta, em uma tarde chuvosa apenas no dia 23. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 3 créditos. este preço das vagas foi alcançado especificamente em um período de demanda baixa, de madrugada com céu limpo no dia 2.

A configuração 1 possui uma média de ocupação igual a 67,57%, 9,92% maior que a média no cenário sem módulo. O gráfico de ocupação mostra grande flutuação de ocupações no começo do mês e estabilidade no fim. A maior taxa de ocupação foi de 96% nos dias 27 e 28 e menor, desconsiderando o primeiro dia, foi de 21% no dia 2. Neste cenário foram recebidos 5000,75 créditos no total, 10,51% a mais que o recebido no cenário sem módulo. O maior preço ao decorrer da simulação foi 6,25 créditos nos momentos de demanda muito alta e o menor preço de 4,5 créditos durante demanda muito baixa que só ocorre no primeiro dia de simulação.

A média de ocupação no cenário da configuração 2 foi de 58,84%, -4,47% menor que o valor do sem módulo, e manteve-se em uma mesma faixa de ocupação durante todo o mês. A maior taxa de ocupação foi de 78% no dia 9 e a menor taxa foi de 17% no dia 2. No fim da execução deste cenário, 4275,75 créditos foram recebidos, -5,83% menor que o total do cenário sem módulo, o pior resultado entre os cenários. O maior preço durante a simulação foi 6,5 créditos ao meio dia e o menor foi 3,5 créditos de madrugada.

A configuração 3 contém uma média de ocupação de 68,19% a qual é 10,92% maior que a média do cenário sem módulo. O gráfico de ocupações mostra constante crescimento do valor das taxas com uma pequena queda na última semana. A maior taxa de ocupação neste mês foi 91% no dia 26 e a menor foi 22% no dia 2, ao desconsiderar

**Gráfico 4 – Ocupações e preços em 100 vagas, 200 drivers e conjunto Baixo**



**Fonte: Autoria Própria (2021).**

o primeiro dia. Os créditos recebidos ao fim deste cenário foram 5168, resultado 14,21% maior que os recebidos no sem módulo. O maior preço na simulação foi de 7,5 créditos nos períodos de demanda alta ao meio-dia e o menor preço foi de 4 créditos durante demanda baixa de madrugada que ocorre apenas nos três primeiros dias.

Entre os cenários da Figura, o da configuração 4 retém a maior ocupação média

em 69,17%, 12,52% maior que a do sem módulo. A maior taxa de ocupação alcançou 94% no dia 29 e 30 e a menor taxa 22% no dia 2, desconsiderando as vagas começando vazias. O total de créditos no fim do mês foi 4339,25, valor -4,28% menor que o total no cenário sem módulo. O maior preço das vagas foi de 6,26 créditos durante clima ensolarado e demanda alta. O menor preço foi de 3,25 em períodos de demandas baixa durante tempestades.

A configuração 5 possui uma média de ocupação igual a 56,09%, a pior média entre os cenários e -9,59% menor que o valor do sem módulo. O gráfico de ocupação deste cenário mostram as taxas de ocupação fluando dois terços do mês em 50%, um pequeno aumento e novamente queda no fim do mês. A maior taxa de ocupação foi de 81% no dia 24 e a menor, após o primeiro dia, foi de 25% no dia 2. Com um total de 5389 créditos, este cenário recebeu 19,10% mais que o do sem módulo e é o melhor dentre todos os da Figura. O maior preço foi 8 créditos e ocorreu somente no dia 23 em um período de demanda alta e chuva de tarde. O menor preço de 4,25 créditos ocorreu apenas no dia 4 no período de noite com céus limpos e demanda baixa.

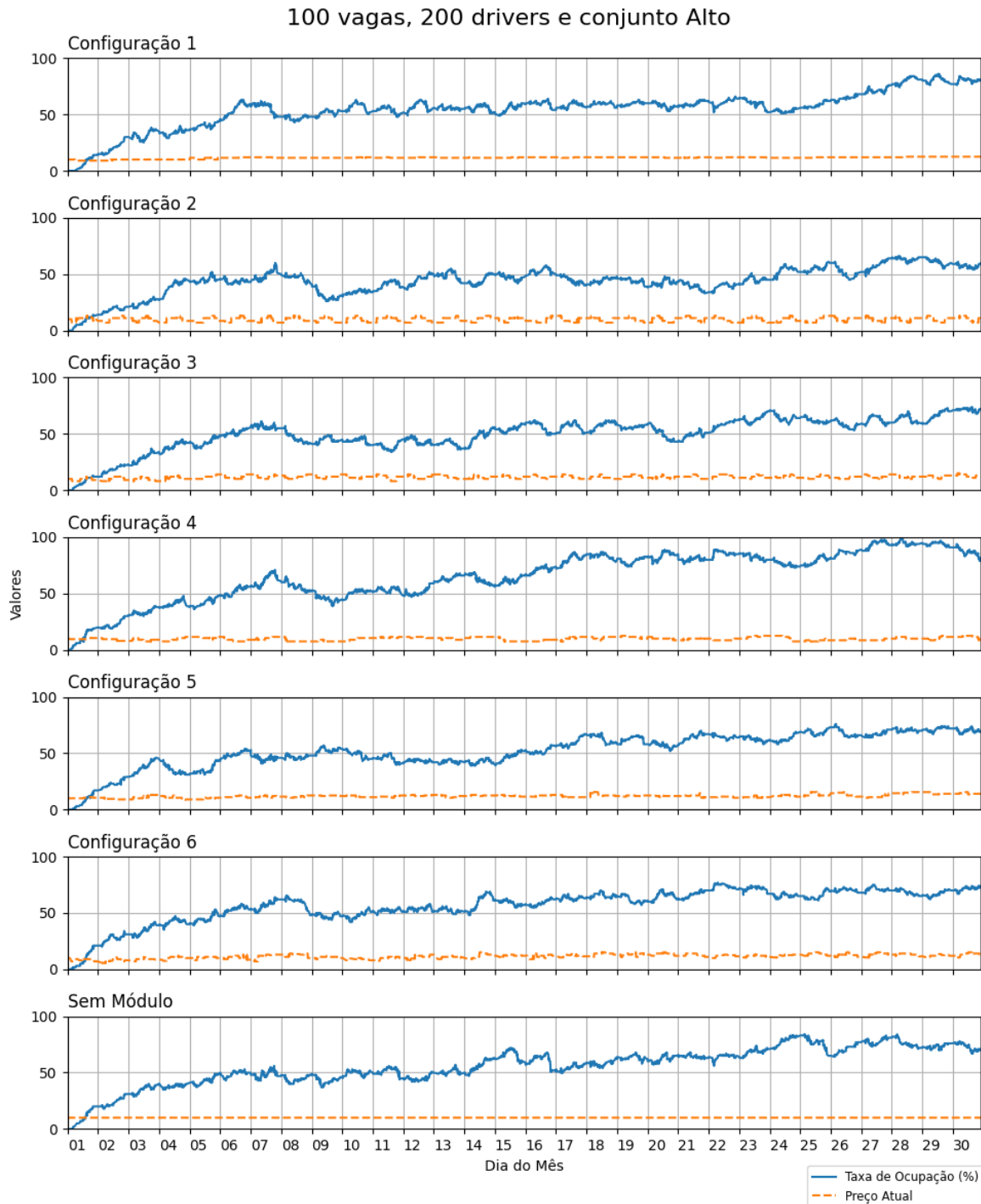
A configuração 6 possui uma média de ocupação igual a 60,53% e -1,55% menor que a média do cenário sem módulo. A maior taxa de ocupação foi de 86% no dia 29 e menor foi de 28% no dia 2 desconsiderando o primeiro dia. Este cenário acumulou 4992 créditos ao fim de sua execução, valor 10,32% maior que o total do sem módulo. O maior preço das vagas de 7,5 créditos ocorreu em um período de demanda alta ao meio-dia em clima nublado exclusivamente no dia 28. O menor preço de 3 créditos ocorreu apenas no dia 2 durante a madrugada, com demanda baixa e tempo nublado.

Os gráficos do cenário sem módulo mostram o funcionamento do estacionamento sem a alteração do preço. A ocupação das vagas inicialmente circunda os 50%, com um curto aumento nos dias 7 e 8, porém para a crescer estavelmente até o fim do mês, com uma queda entre os dias 25 e 28. A média de ocupação foi 61,47%, valor maior que de duas configurações, 2, 5 e 6. A maior taxa de ocupação no mês foi de 82% no dia 30 e a menor, após o primeiro dia, foi de 25% no dia 2. Ao fim deste cenário foram recebidos 4525 créditos, valor maior que de duas configurações, 2 e 4. Ambas as configurações 1 e 3 possuem ocupação média e total de créditos recebidos maiores do que no cenário sem módulo.

O Gráfico 5 mostra os cenários com uso de conjunto alto: preço base da vagas em 10 créditos e crédito máximo dos motoristas em 20 créditos. Logo, nos cenários

com este conjunto, o valor da vaga começa em 10 créditos, o maior valor aceito por um *driver* é de 20 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

**Gráfico 5 – Ocupações e preços em 100 vagas, 200 *drivers* e conjunto Alto**



Fonte: Autoria Própria (2021).

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 100% e ocorreu com o uso da configuração 4 nos dias 27 e 28.



A menor ocupação, após o primeiro dia, encontram-se nas configuração 2 e 3 no dia 2 em 16% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 5 que alcançou 15,5 créditos durante o período de demanda alta de tarde em tempo nublado e também durante demanda alta de manhã em tempo chuvoso. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 5,5 créditos. este preço das vagas foi alcançado especificamente em um período de demanda baixa, de madrugada durante tempo chuvoso no dia 2.

A ocupação média na configuração 1 foi 55,89%, valor -0,03% menor do que no cenário sem módulo. O gráfico de ocupações mostra taxas predominantemente maiores que 50% e um maior crescimento na última semana do mês. A maior taxa de ocupação durante a simulação foi de 86% no dia 29 e a menor após o primeiro dia foi de 20% no dia 2. O total adquirido em toda a simulação foi de 7553,5 créditos, valor 2,77% maior do que no sem módulo. O maior preço das vagas foi de 12,5 créditos durante demanda muito alta que ocorre apenas do dia 28 á 30. O menor valor das vagas, 9 créditos, ocorre apenas nos dias 1 e 2 durante o período de demanda muito baixa.

Na simulação do cenário com configuração 2, a ocupação média foi de 45,40% e -23,35% menor que a da sem módulo e a pior média entre os cenários na Figura. O gráfico de ocupações mostram os poucos momentos que ela passa de 50% e é único que contém porcentagem menor que 30 depois da primeira semana. Durante a execução, a maior taxa de ocupação foi de 67% no dia 28 e a menor, após o primeiro dia, foi de 16% no dia 2. Foram recebidos 7350 créditos ao fim do mês, total igual ao do sem módulo. O maior preço atinge 13 créditos ao meio dia e o menor preço é de 7 créditos durante a madrugada.

A configuração 3 possui neste cenário uma ocupação média de 51,42% e -8,91% menor que média do cenário sem módulo. O gráfico de ocupações mostra taxas circulando os 50% e permanecendo acima deste durante a última semana. Na execução deste cenário, a maior taxa de ocupação alcançada foi de 76% no dia 30 e a menor após o primeiro dia foi de 16% no dia 2. Neste cenário, o total de créditos recebidos foi 8070 que é 9,79% maior que o total do sem uso do módulo e o melhor resultado entre todos os cenários. O maior preço das vagas alcançado foi 15 créditos nos períodos de demanda alta ao meio-dia e o menor preço atinge 8 créditos durante demanda baixa de madrugada.

A média das taxas de ocupação na configuração 4 foi 66,05%, a maior média

entre os cenários e 17,94% maior que a do sem módulo. O gráfico de ocupação mostra um estacionamento que passou dois terços do mês com mais da metade de suas vagas ocupadas e o único que atinge lotação máxima. A maior taxa de ocupação foi de 100% nos dias 27 e 28 e a menor após o primeiro dia foi de 24% no dia 2. O total de créditos adquiridos foram 7239,5 e -1,52% menor que o total do cenário sem módulo. Este resultado é o menor total entre os cenários, ficando a baixa da configuração 6 por 0,42%. O maior valor das vagas durante a simulação foi 12,5 créditos quando a demanda estava em nível alto e o clima ensolarado. O menor preço foi 7,5 créditos nos períodos de demanda normal durante tempestades.

A configuração 5 possui uma média de ocupação de 52,51%, valor -6,65% menor que no cenário sem uso de módulo. Neste cenário, as taxas de ocupações passaram a primeira metade do mês abaixo de 50% e a segunda metade acima. A maior taxa de ocupação foi de 76% no dia 26 e a menor após o primeiro dia, foi de 29% no dia 2. Este cenário acumulou 7317,5 créditos ao fim de sua execução, valor -0,44% menor que o total do sem módulo. O maior preço das vagas atingiu 15,5 créditos com duas regras diferentes: em períodos de demanda alta, a tarde e tempo nublado e durante demanda alta, de manhã e tempo chuvoso. O menor preço das vagas foi 8,5 créditos durante demanda baixa a noite com céus limpos que ocorreu apenas no dia 4.

A ocupação média na configuração 6 foi de 57,32%, 2,35% maior que a média do cenário sem módulo. Além da queda nos dias 8 a 10 e 14, a ocupação deste cenário foi predominantemente superior a 50%. A maior ocupação foi 77% no dia 22 e, após o primeiro dia, a menor ocupação foi 20% no dia 2. Neste cenário foram recebidos um total de 7270,5 créditos, -1,09% menor que os créditos recebidos no cenário sem módulo. O maior preço alcançou 15 créditos nos períodos ensolarados ao meio-dia durante demanda alta. O menor preço foi de 5,5 créditos e ocorre durante chuva de madrugada em demanda baixa.

O cenário sem módulo possui o característico gráfico de preço sem alterações ao decorrer dos dias. A sua média de ocupação foi 56%, com a maior taxa em 84% no dia 28 e menor em 20% no dia 2. O total de créditos adquiridos foi 7530. A média de ocupação sem uso do módulo supera as configurações 1, 2, 3 e 5 e o total de créditos recebidos é maior que o das configurações 4, 5 e 6 e igual ao total da configuração 2. Dentre os gráficos, nenhuma configuração possui ambas ocupação média e créditos recebidos maiores que os do cenário sem módulo.

#### 6.2.4 Cenários com 200 vagas e 50 *drivers*

O foco nos cenários de 200 vagas e 50 *drivers* é observar um estacionamento com quantidade alta de vagas as quais são utilizadas por poucos motoristas. Todas as vagas iniciam a simulação vazias e os *drivers*, criados ao mesmo tempo, decidem aleatoriamente em que dia as usarão (ver Subseção 5.2).

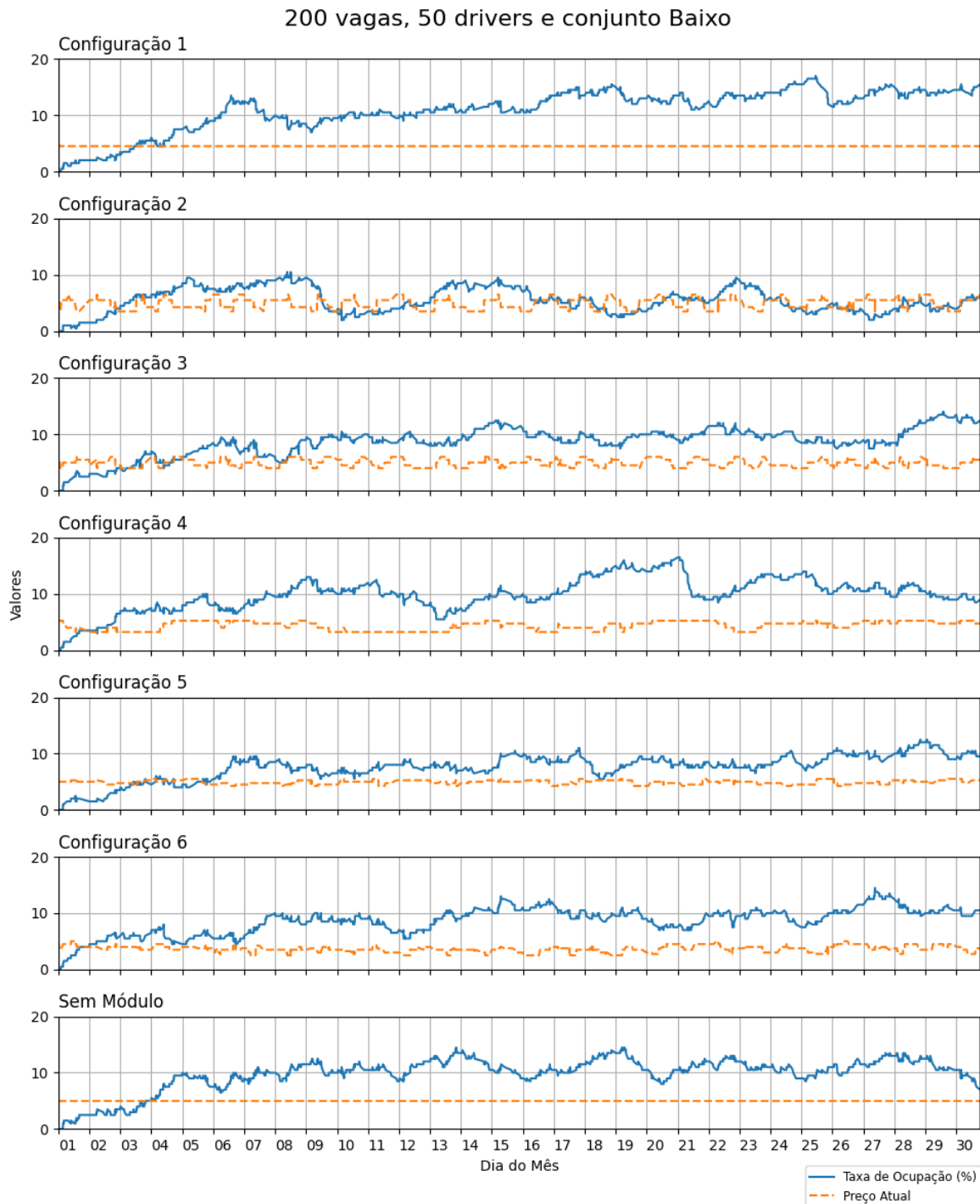
O Gráfico 6 mostra os cenários com uso de conjunto baixo: preço base da vagas em 5 créditos e crédito máximo dos motoristas em 10 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 5 créditos, o maior valor aceito por um *driver* é de 10 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 17% e ocorreu com o uso da configuração 1 no dia 25. A menor ocupação, após o primeiro dia, encontram-se na configuração 2 no dia 2 em 0,5% de ocupação. Dos gráficos que são influenciados pela demanda, nenhuma alcança ocupação necessária para alterar o preço. Contando com isso, o maior preço entre os gráficos encontra-se no de configuração 2 que alcançou 6,5 créditos durante o período do meio-dia. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 2,5 créditos. este preço das vagas foi alcançado especificamente durante tempestades de madrugada.

A configuração 1 possui uma ocupação média de 11,27%, a maior média e 14,47% a mais que o valor do cenário sem módulo. A maior taxa de ocupação alcançada foi de 17% no dia 25 e a menor depois do primeiro dia foi de 1,5% no fim do dia 2. Como em nenhum momento do mês a ocupação alcança 20%, o gráfico de preço apresenta um comportamento semelhante com o do de sem uso do módulo. A diferença entre eles é que, devido a baixa demanda, o preço durante o resto do mês é igual 4,5 créditos, -0,5 menor que o preço no cenário sem uso do módulo. O total de créditos recebidos neste cenário foi 931,5 e -17% menor que o valor do sem módulo. O maior preço de 5 créditos encontra-se nas primeiras 3h da simulação, antes da primeira chamada do módulo de precificação. O menor preço de 4,5 créditos ocorre em todos os dias restantes.

A configuração 2 possui o maior total de créditos recebidos, 1144, e a menor ocupação média, 5,76%. O total recebido é 4,95% maior e a média -71,04% menor que o valor do cenário sem módulo. O gráfico de ocupações contém, na maioria do mês,

**Gráfico 6 – Ocupações e preços em 200 vagas, 50 drivers e conjunto Baixo**



**Fonte: Autoria Própria (2021).**

taxas abaixo de 10% apenas alcançando valor superior em sua maior taxa de 10,5% no dia 7. A menor taxa de ocupação, depois do primeiro dia, foi de 0,5% no dia 2. No seu gráfico de preço é mostrado a mudança de valor periódica que ocorre quando este é afetado apenas pela hora do dia. O maior preço das vagas atingido é de 6,5 sempre ao meio dia e o menor preço foi de 3,5 nas madrugadas.

Na configuração 3 o preço das vagas não é alterado pelas regras de demanda ao nunca alcançar 40% de ocupação, necessário para atingir o segundo intervalo. Logo o gráfico de preços atuais mostra apenas a mudança de preços devido ao horário do dia. Ao fim do mês a média das taxas de ocupação foi de 9,21% e -6,94% menor que a do sem módulo. A maior taxa neste cenário foi 15% no dia 29 enquanto a menor taxa foi 2,5% no dia 2. O total de créditos adquiridos foi 1066 e -2,25% menor que o valor do sem módulo. O maior preço durante a simulação neste cenário foi de 6 créditos e ocorreu durante o meio dia enquanto o menor foi de 4 créditos durante a madrugada.

A configuração 4 mostra novamente gráficos onde as regras de demanda não alteram o preço devido a taxa de ocupação não alcançar 30%. A ocupação média de 10,26% deste cenário é a segunda melhor e 4,18% maior que a de sem módulo. A maior taxa de ocupação durante a simulação foi de 16,5% no dia 21 e a menor taxa depois do primeiro dia foi de 3% no dia 2. O gráfico de preços apresentam o preços de cada dia de acordo apenas com o clima simulado. Durante a execução deste cenário foram recebidos 965 créditos no total, -12,95% menor que o valor do sem módulo. O maior valor da vaga atingiu 5,25 créditos durante clima ensolarado e o menor preço de 3,25 durante tempestades.

A configuração 5 possui uma ocupação média de 8,21% e -19,94% menor que o valor do sem módulo. A maior taxa de ocupação alcançou 13,5% nos dias 28 e 29 e a menor, após o primeiro dia, foi de 1,5% no dia 2. Devido a ocupação nunca alcançar 40%, a alteração do preço nesta configuração considerou apenas a condição climática e o horário do dia. O total de créditos recebidos foi 1139,75, 4,56% a mais que o do sem módulo. O maior preço das vagas neste cenário foi de 5,5 créditos em tempo chuvoso no horário da tarde e o menor foi de 4,25 créditos durante a noite com céu limpo.

Os gráficos da configuração 6 mostram uma ocupação média de 8,82% e -11,70% menor que de sem módulo. A maior taxa de ocupação alcançada foi de 14,5% no dia 27 e a menor, após o primeiro dia, foi de 5% no dia 2. O gráfico de preços, comparados com o da configuração 5, mostram a influência da quantidade de descontos na alteração do preço. Na execução deste cenário foram recebidos 845,5 créditos, o pior resultado e -28,91% menor que o total do sem módulo. O maior preço durante a simulação foi de 5 créditos e ocorre durante o meio dia em clima ensolarado e o menor foi de 2,5 durante tempestades de madrugada.

O cenário sem módulo mostra o característico gráfico de preços sem alterações

durante os dias. A sua ocupação média de 9,85%, valor maior que a das configurações 2, 3, 5 e 6. A maior taxa de ocupação foi de 14,5% no dia 19 e a menor ocupação depois do primeiro dia foi de 2,5% no dia 2. Ao fim de sua simulação, este cenário adquiriu um total de 1090 valor maior que os das configurações 1, 3, 4 e 6. Nenhuma configuração teve resultados superiores ao cenário sem módulo em ambos quesitos.

#### 6.2.5 Cenários com 200 vagas e 100 *drivers*

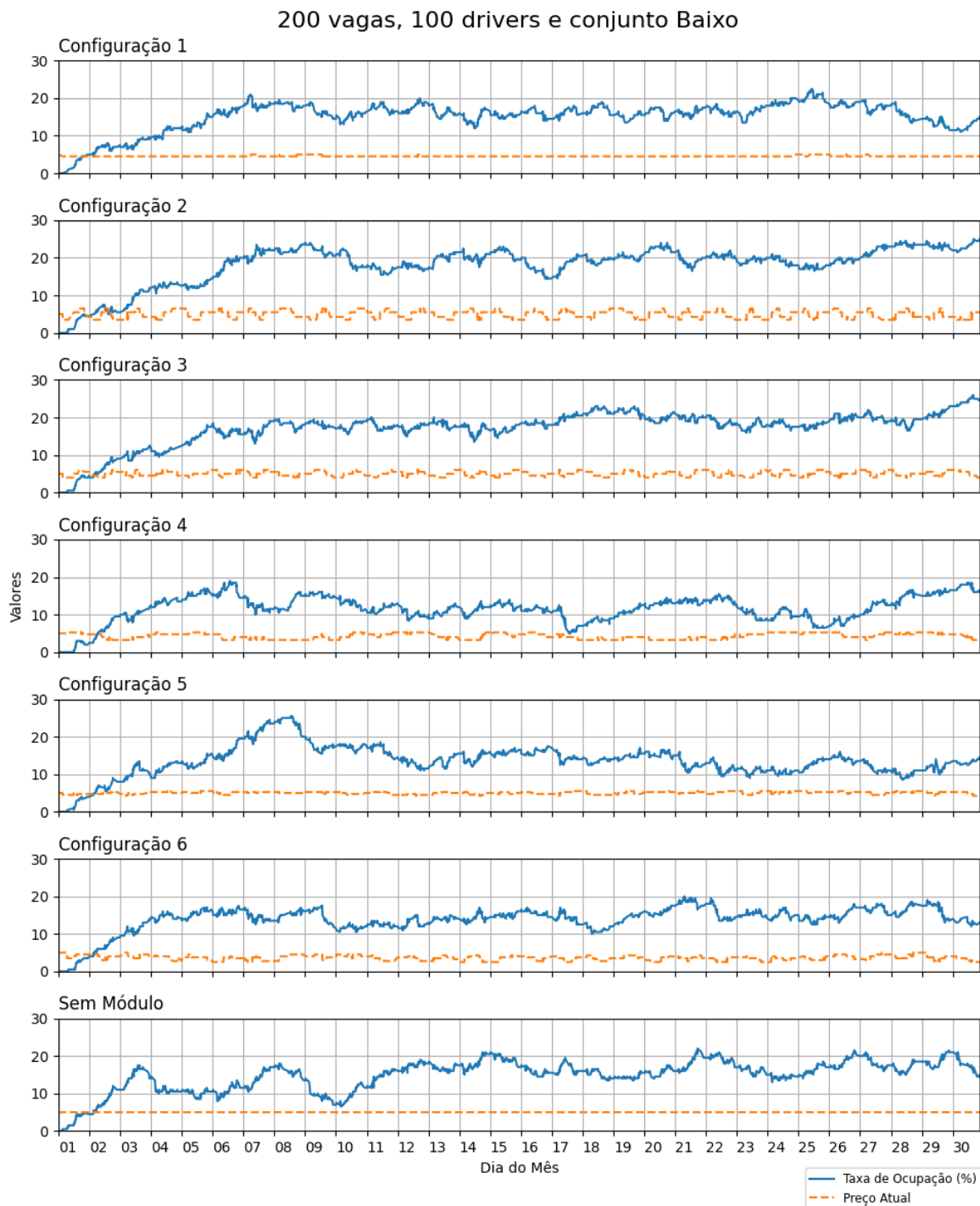
O foco nos cenários de 200 vagas e 100 *drivers* é observar um estacionamento com quantidade alta de vagas as quais são utilizadas por um número médio de motoristas. Todas as vagas iniciam a simulação vazias e os drivers, criados ao mesmo tempo, decidem randomicamente em que dia as usarão (ver Subseção 5.2).

O Gráfico 7 mostra os cenários com uso de conjunto baixo: preço base da vagas em 5 créditos e crédito máximo dos motoristas em 10 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 5 créditos, o maior valor aceito por um *driver* é de 10 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 26% e ocorreu com o uso das configuração 3 no dia 30. A menor ocupação, após o primeiro dia, encontram-se na configuração 4 no dia 2 em 2% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 2 que alcançou 6,5 créditos durante o meio-dia e o menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 2,5 créditos durante tempestades de madrugada. Ao comparara com os cenários proporcionais de 100 vagas e 50 *drivers*, identifica-se que as ocupações médias mantiveram-se semelhantes com uma diferença em geral de 0,90% entre todos os cenários. Já no total de créditos recebidos, os cenários do Gráfico 7 receberam no total o dobro de créditos do que aqueles da Gráfico 1.

A configuração 1 possui uma média de ocupação igual a 15,58%, 4,91% maior que a média no cenário sem módulo. O gráfico de ocupações mostra valores no intervalo de 10 e 20% na maioria dos dia contendo picos nos dias 7 e 25. A maior taxa de ocupação foi de 22,5% nos dia 25 e a menor, desconsiderando o primeiro dia, foi de 5% no dia 2. Neste cenário foram recebidos 2096,5 créditos no total, -7,32% menor que o recebido no cenário sem módulo. O maior preço ao decorrer da simulação foi 5 créditos nos

**Gráfico 7 – Ocupações e preços em 200 vagas, 100 drivers e conjunto Baixo**



**Fonte: Autoria Própria (2021).**

poucos períodos em que a ocupação alcança 20% e entra na regra de demanda baixa. O menor preço de 4,5 créditos ocorre nos dias restantes de ocupação menor que 20% que define a regra de demanda muito baixa.

A média de ocupação no cenário da configuração 2 foi de 19,04%, 2,15% maior do que no sem módulo, e a melhor média entre os cenários da Figura. Após o

crescimento inicial da simulação, as taxas de ocupações neste cenário possuem valor circulando os 20% durante todo o mês. A maior taxa de ocupação foi de 25,5% no dia 30 e a menor taxa foi de 5,5% no dia 2. No fim da execução deste cenário 2285,5 créditos foram recebidos, 1,57% a mais que o total do cenário sem módulo, o melhor resultado entre os cenários da Figura. O maior preço durante a simulação foi 6,5 créditos ao meio dia e o menor foi 3,5 créditos de madrugada.

A configuração 3 contém uma média de ocupação de 17,34% a qual é 16,72% maior que a média do cenário sem módulo. O gráfico de ocupações mostra constante crescimento do valor das taxas sem queda significativa durante o mês. A maior taxa de ocupação neste mês foi 26% no dia 30 e a menor foi de 4% no dia 2, ao desconsiderar o primeiro dia. As alterações dos valores no gráfico de preço são afetadas apenas pelo horário do dia já que a ocupação nunca alcançou 40%. Os créditos recebidos ao fim deste cenário foram 2164,5, resultado -3,95% menor que os recebidos no sem módulo. O maior preço na simulação foi de 6 créditos ao meio-dia e o menor preço foi de 4 crédito de madrugada.

Entre os cenários da Figura, o da configuração 4 retém a menor ocupação média em 12,26% e -21,20% menor que o valor do sem módulo. A ocupação neste cenário nunca alcança 20% e está em queda a maior parte do mês chegando a valores menores que 10% após a primeira semana. A maior taxa de ocupação foi 19% no dia 6 e a menor taxa, após o primeiro dia, foi de 2% no dia 2. O gráfico de preço mostra apenas as mudança por influencia da condição climática. O total de créditos no fim do mês foi 1994,25 e -12,76% menor que o total no cenário sem módulo. O maior preço das vagas foi de 5,25 créditos durante clima ensolarado e o menor preço foi de 3,25 durante tempestades.

A configuração 5 possui uma média de ocupação igual a 12,85%, valor -15,60% menor do que no sem módulo. O gráfico de ocupação deste cenário mostram um crescimento na primeira semana seguido por queda e baixa ocupação no resto do mês. A maior taxa de ocupação foi de 25,5% no dia 8 e a menor, após o primeiro dia, foi de 4,5% no dia 2. Com um total de 1999,25 créditos, este cenário recebeu -12,54% créditos que o do sem módulo. O gráfico de preços foram influenciados apenas pelo horário do dia e condição climática já que a ocupação nunca chegou ao valor de 40%. O maior preço foi 5,5 créditos durante chuva de tarde e o menor preço de 4,25 créditos ocorre de noite com céus limpos.



A configuração 6 possui uma média de ocupação igual a 14,13% e -5,11% menor que a média do cenário sem módulo. A maior taxa de ocupação foi de 20% nos dias 21 e 22 e a menor foi de 4% no dia 2 desconsiderando o primeiro dia. O gráfico de ocupações nunca alcança um valor de 30% e portanto as regras de demanda não são consideradas na mudança do preço. Este cenário acumulou 1732,75 créditos ao fim de sua execução, a menor quantidade entre os cenários e -27,85% menor que o total do sem módulo. O maior preço das vagas de 5 créditos ocorre durante o meio-dia em tempo ensolarado e o menor preço de 2,5 créditos ocorre durante tempestades de madrugada.

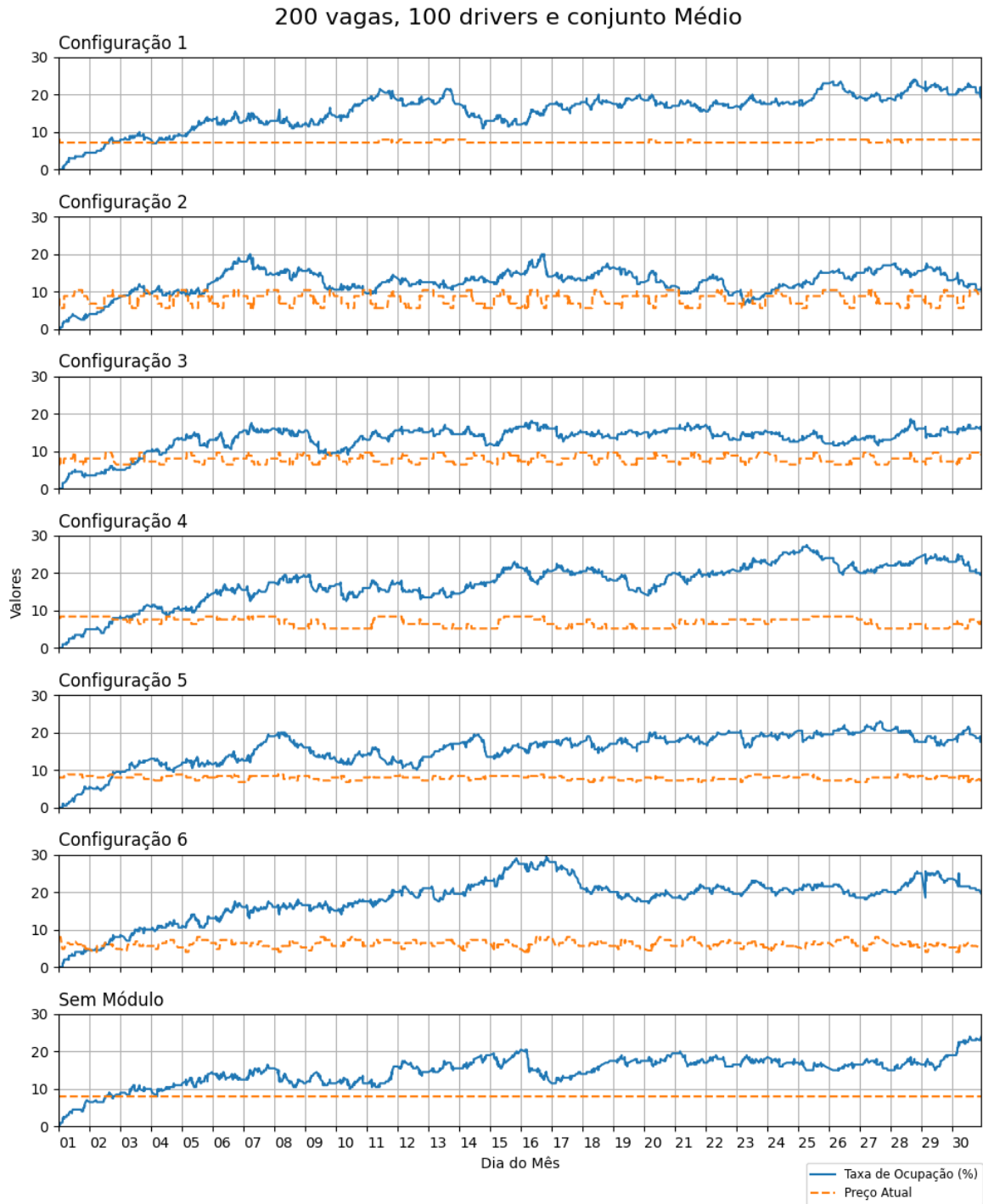
Os gráficos do cenário sem módulo mostram o funcionamento do estacionamento sem a alteração do preço. A média de ocupação foi 14,86%, valor maior que das configurações 4, 5 e 6. A maior taxa de ocupação no mês foi de 22% no dia 21 e a menor, após o primeiro dia, foi de 4,5% no dia 2. Ao fim deste cenário foram recebidos 2250 créditos, valor apenas menor que o da configuração 2, a única superior em ambos quesitos.

O Gráfico 8 mostra os cenários com uso de conjunto médio: preço base da vagas em 8 créditos e crédito máximo dos motoristas em 15 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 8 créditos, o maior valor aceito por um *driver* é de 15 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 29,5% e ocorreu com o uso da configuração 6 nos dias 15 e 16. A menor ocupação, após o primeiro dia, encontram-se na configuração 3 no dia 2 em 3% de ocupação. O maior preço de vagas entre os gráficos encontra-se no de configuração 2 que alcançou 10,4 créditos durante o período do meio-dia. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 4 créditos durante tempestades de madrugada.

A configuração 1 possui uma ocupação média de 15,80%, 6,97% a mais do que no cenário sem módulo. O gráfico de ocupações mostra essas em crescimento na primeira metade do mês, queda e estabilização no meio e alcançando suas maiores taxas na última semana. A maior taxa de ocupação alcançada foi de 24,5% no dia 28 e a menor depois do primeiro dia foi de 3,5% no fim do dia 2. O total de créditos recebidos neste cenário foi 2944, valor -15,49% menor do que no sem módulo. Nos

**Gráfico 8 – Ocupações e preços em 200 vagas, 100 drivers e conjunto Médio**



**Fonte: Autoria Própria (2021).**

poucos momentos do mês em que a ocupação alcança 20%, o maior valor de 8 créditos é usando como preço das vagas. O menor preço de 7,2 créditos ocorre em todos os dias restantes de demanda muito baixa.

A configuração 2 possui uma ocupação média de 12,79%, a pior média e -15,49% menor que da sem módulo. O gráfico de ocupações mostra valores predomi-

nantemente ao redor de 10% e dias de taxas superiores, como 7 e 15, alcançado 20%. A menor taxa de ocupação, depois do primeiro dia, foi de 4% no dia 2. Neste cenário foram recebidos 2990 créditos ao fim da simulação, valor -13,71% menor que o do sem módulo. O gráfico de preço é mostra a mudança de valor periódica que ocorre quando afetado apenas pela hora do dia. O maior preço das vagas atingido é de 10,4 créditos sempre ao meio dia e o menor preço foi de 5,6 créditos nas madrugadas.

Na configuração 3 o preço das vagas não é alterado pelas regras de demanda ao nunca alcançar 40% de ocupação, necessário para atingir o segundo intervalo. Logo o gráfico de preços atuais mostra apenas a mudança de preços devido ao horário do dia. Ao fim do mês a média das taxas de ocupação foi de 14% e -15,49% menor que o valor do sem módulo. A maior taxa neste cenário foi 18,5% no dia 28 enquanto a menor taxa foi 5% no dia 2. O total de créditos adquiridos foi 3256 e -4,42% menor do que no sem módulo. O maior preço durante a simulação neste cenário foi de 9,6 créditos e ocorreu durante o meio dia enquanto o menor foi de 6,4 créditos durante a madrugada.

A configuração 4 mostra novamente gráficos onde as regras de demanda não alteram o preço devido a taxa de ocupação não alcançar 30%. A ocupação média de 17,78% deste cenário é a segunda melhor e 20,37% maior que a de sem módulo. A maior taxa de ocupação durante a simulação foi de 27,5% no dia 25 e a menor taxa depois do primeiro dia foi de 4% no dia 2. O gráfico de preços apresentam o preços de cada dia de acordo apenas com o clima simulado. Durante a execução deste cenário foram recebidos 2910,8 créditos no total, -16,81% menor que no do sem módulo. O maior valor da vaga atingiu 8,4 créditos durante clima ensolarado e o menor preço de 5,2 durante tempestades.

A configuração 5 possui uma ocupação média de 15,61%, 5,6% maior que a de sem módulo. A maior taxa de ocupação alcançou 23% no dia 27 e a menor, após o primeiro dia, foi de 4,5% no dia 2. Devido a ocupação nunca alcançar 40%, a alteração do preço nesta configuração considerou apenas a condição climática e o horário do dia. O total de créditos recebidos foi 3434, 1% a mais que o do sem módulo e o maior total entre os cenários da Figura. O maior preço das vagas neste cenário foi de 8,8 créditos em tempo chuvoso no horário da tarde e o menor foi de 6,8 créditos durante a noite com céu limpo.

Os gráficos da configuração 6 mostram uma ocupação média de 18,04%, 22,13% maior que de sem módulo e a melhor média entre os cenários. A maior taxa de

ocupação alcançada foi de 29,5% nos dias 15 e 16 e a menor, após o primeiro dia, foi de 4,5% no dia 2. O gráfico de preços, comparados com o da configuração 5, mostram a influência do valor dos descontos na alteração do preço. Na execução deste cenário foram recebidos 2265,2 créditos, valor -50,10% menor que o total do sem módulo e o pior total recebido. O maior preço durante a simulação foi de 8 créditos e ocorre durante o meio dia em clima ensolarado e o menor foi de 4 durante tempestades de madrugada.

O cenário sem módulo mostra o característico gráfico de preços sem alterações durante os dias. A sua ocupação média de 14,77%, foi maior que a das configurações 2 e 3. A maior taxa de ocupação foi de 25% no dia 30 e a menor ocupação depois do primeiro dia foi de 6,5% no dia 2. Ao fim de sua simulação, este cenário adquiriu um total de 3400 créditos, segundo maior valor, abaixo apenas do resultado da configuração 5 que é superior em ambos quesitos.

#### 6.2.6 Cenários com 200 vagas e 200 *drivers*

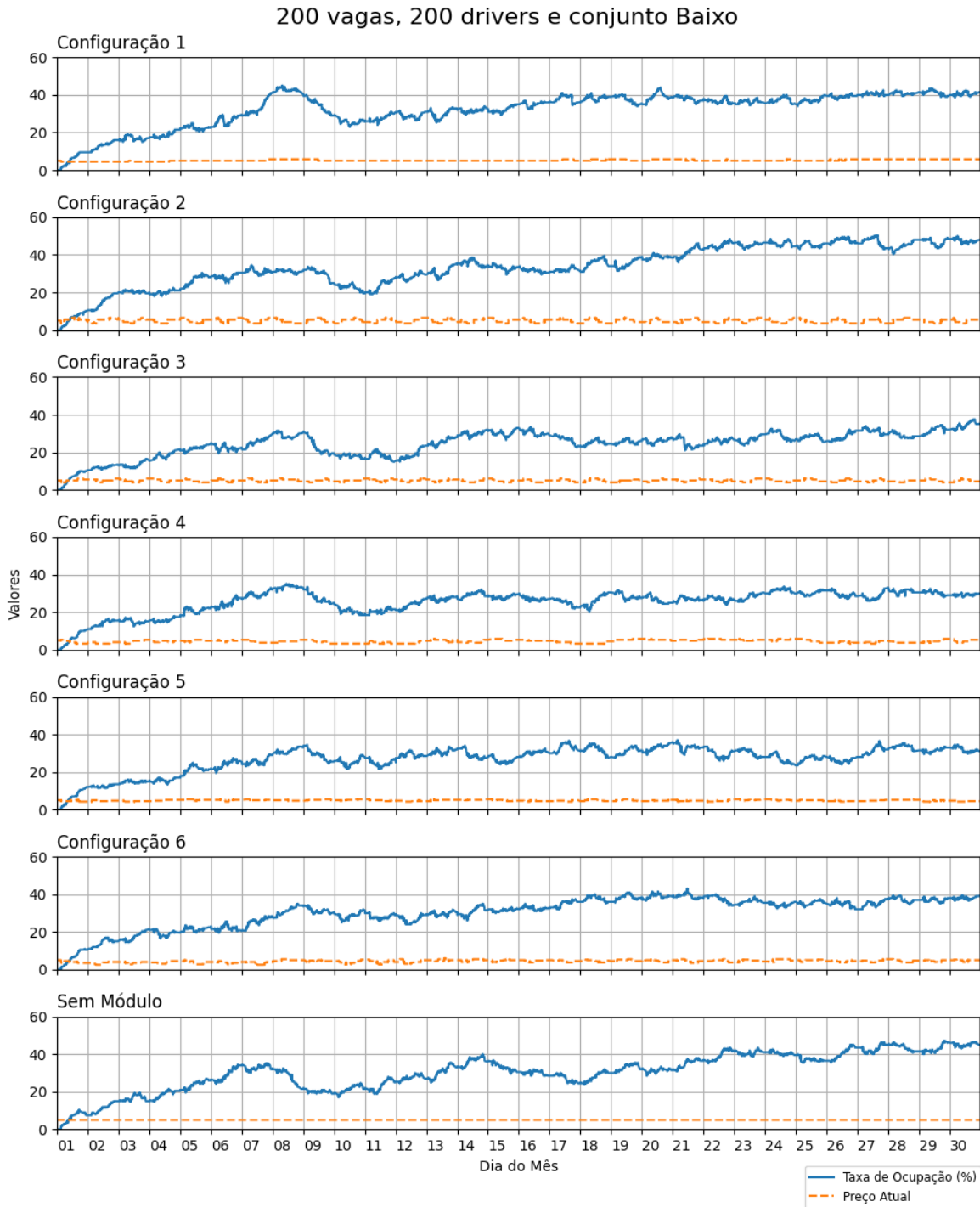
O foco nos cenários de 200 vagas e 200 *drivers* é observar um estacionamento com quantidade alta de vagas as quais são utilizadas por um número alto de motoristas. Todas as vagas iniciam a simulação vazias e os *drivers*, criados ao mesmo tempo, decidem aleatoriamente em que dia as usarão (ver Subseção 5.2).

O Gráfico 9 mostra os cenários com uso de conjunto baixo: preço base da vaga em 5 créditos e crédito máximo dos motoristas em 10 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 5 créditos, o maior valor aceito por um *driver* é de 10 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 51% e ocorreu com o uso da configuração 2 no dia 27. A menor ocupação, após o primeiro dia, encontram-se no gráfico do cenário sem módulo no dia 2 em 7,5% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 2 que alcançou 6,5 créditos durante o período do meio-dia. O menor preço de vaga é encontrado no gráfico de configuração 6 com o preço de 2,5 créditos nos períodos de demanda baixa, de madrugada durante tempestades. Comparando com cenários proporcionais de 100 vagas e 100 *drivers*, encontra-se as ocupações médias da Gráfico 9 foram 0,82% mais baixas porém adquiriram 1,96 vezes mais créditos no

total do que aqueles da Gráfico 2.

**Gráfico 9 – Ocupações e preços em 200 vagas, 200 drivers e conjunto Baixo**



**Fonte: Autoria Própria (2021).**

A ocupação média na configuração 1 foi 32,88%, 6,82% maior que a do cenário sem módulo. A maior taxa de ocupação durante a simulação foi de 45% nos dias 8, 20 e 29. A menor taxa após o primeiro dia foi de 9,5% no dia 2. A escala do gráfico de ocupação reduz a clareza do gráfico de preço, porém ainda é observável a influência da

demanda. Começando na tarde do dia 4 até o fim do mês, a ocupação ultrapassa 20% e o desconto no preço é removido. Durante todo dia 8 a ocupação é superior a 40% e o preço recebe um aumento de acordo com a regra de demanda normal. A partir do dia 17 a ocupação ocasionalmente passa 40%, especialmente na última semana, e nesses casos o preço das vagas atinge seu valor mais alto neste cenário de 5,75 créditos. O menor valor das vagas, 4,5 créditos, ocorre apenas no início da simulação durante o período de demanda muito baixa. O total adquirido em toda a simulação foi de 4679,75 créditos, valor 7,09% maior do que no sem módulo e o melhor total entre os cenários.

Na simulação do cenário com configuração 2, a ocupação média foi de 35,16% a qual é a melhor média e 14,24% maior que a da sem módulo. Durante os primeiros dois terços do mês, as taxa de ocupações permanecem no intervalo, 20 a 40%, comum entre os cenário da Figura. O último terço do mês contém apenas ocupações acima de 40%, a maior taxa de ocupação alcançando 51% no dia 27. A menor taxa de ocupação, após o primeiro dia, foi de 10,5% no dia 2. O gráfico de preço mostra a mudança periódica quando considerado apenas a hora do dia. Foram recebidos 4421,5 créditos ao fim do mês, total 1,18% maior que o valor do sem módulo. O maior preço atinge 6,5 créditos ao meio dia e o menor preço é de 3,5 créditos durante a madrugada.

A configuração 3 possui neste cenário a pior ocupação média de 25,53% e -20,53% menor que a do cenário sem módulo. Na execução deste cenário, a maior taxa de ocupação alcançada foi de 39,5% no dia 30 e a menor após o primeiro dia foi de 11,5% no dia 2. Neste cenário, o total de créditos recebidos foi 4573,5 que é -4,65% menor que o total do sem uso do módulo. Como o nenhuma taxa de ocupação alcançou 40%, as regras de demanda não afetaram a alteração do preço. O maior valor das vagas alcançado foi 6 créditos ao meio-dia e o menor atinge 4 créditos durante a madrugada.

A média das taxas de ocupação na configuração 4 foi 26,20%, menor que a do sem módulo por -17,47%. A ocupação do estacionamento ultrapassa 20% no dia 6 e, embora possuir uma queda entre os dias 10 e 11, permanece no intervalo de 20 e 40% a maioria do mês. A maior taxa de ocupação foi de 36,5% no dia 8 e a menor após o primeiro dia foi de 13,5% no dia 2. O total de créditos adquiridos foram 4107 e -6,40% menor que o total do cenário sem módulo. O maior valor das vagas durante a simulação foi 5,75 créditos quando a demanda estava em nível normal e o clima ensolarado. O menor preço foi 3,25 créditos nos períodos de demanda baixa durante tempestades.

A configuração 5 possui uma média de ocupação em 27,16% e -13,32% menor que o valor do sem uso de módulo. O gráfico de ocupações apresenta taxas que superam os 20% a partir do dia 6 e permanecem no intervalo de 20 e 40% até o fim do mês. A maior taxa de ocupação foi de 37% no dia 27 e a menor, após o primeiro dia, foi de 14,5% no dia 2. O total adquirido após a execução foi de 4590,25 créditos, 5,04% maior que o valor do sem módulo. O gráfico de preço foi influenciado apenas pelo horário do dia e condição climática, já que a ocupação nunca alcançou 40% e logo não saiu da regra de demanda baixa. O maior preço das vagas atingiu 5,5 créditos durante chuvas ao meio-dia e o menor foi 4,25 créditos durante noites com céus limpos.

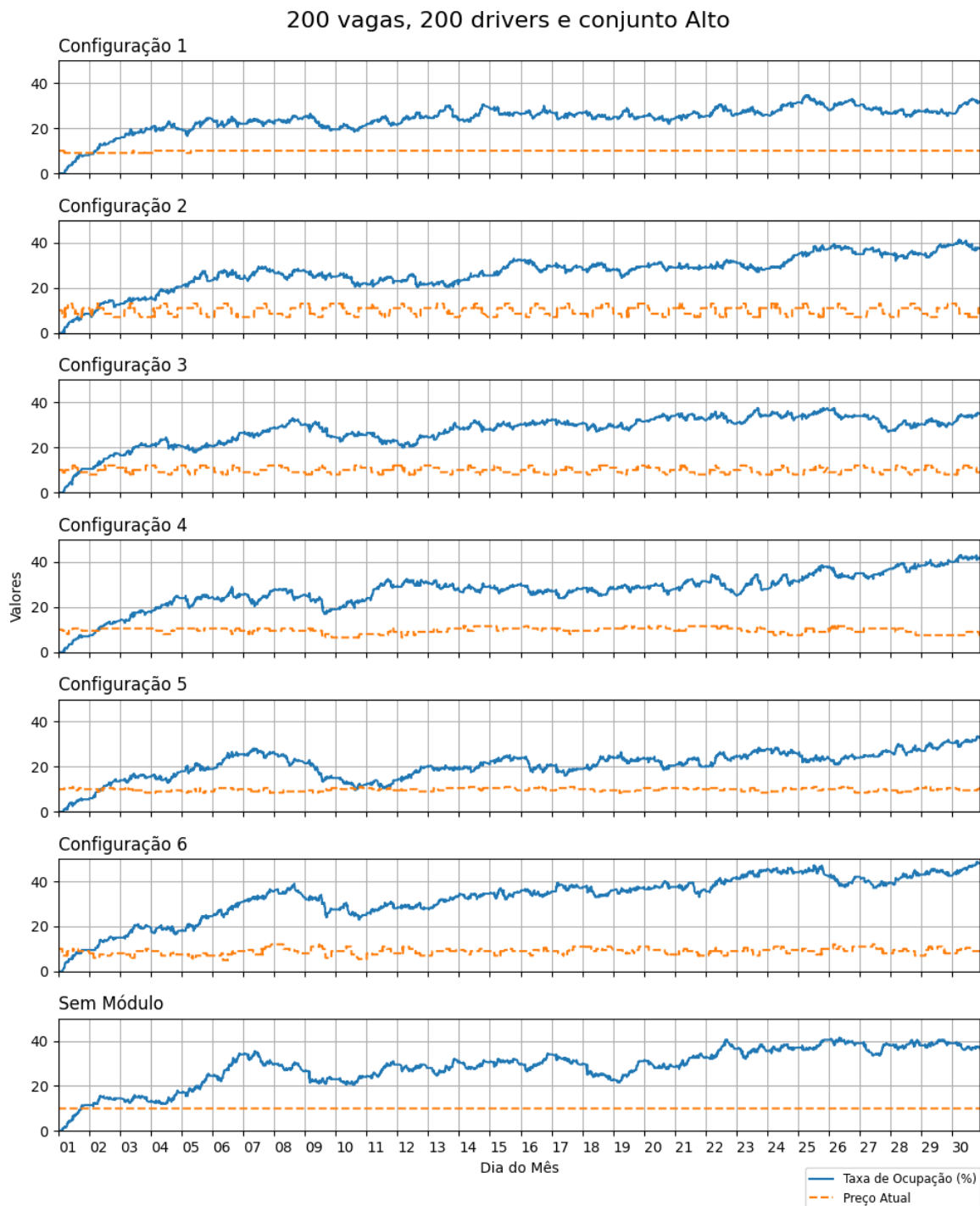
A ocupação média na configuração 6 foi de 30,93%, 0,49% maior que a média do cenário sem módulo. O gráfico de ocupações mostra taxas contidas em sua maioria no intervalo de 20 e 40%. As exceções ocorrendo abaixo de 20% na primeira semana e acima de 40% dos dias 18 a 21. A maior ocupação foi 43% no dia 21 e, após o primeiro dia, a menor ocupação foi 13% no dia 2. Neste cenário foram recebidos um total de 4030 créditos, o pior resultado e -8,44% menor que o do sem módulo. O maior preço alcançou 6 créditos nos períodos ensolarados ao meio-dia em média demanda. O menor preço foi de 2,5 créditos e ocorre durante uma tempestade de madrugada em demanda baixa.

O cenário sem módulo possui o característico gráfico de preço sem alterações ao decorrer dos dias. A sua média de ocupação foi 30,78%, valor superior aos das configurações 3, 4 e 5 e com a maior taxa em 47% no dia 29 e menor em 7,5% no dia 2. O total de créditos adquiridos foi 4370, resultado maior que o das configuração 4 e 6. As configurações 1 e 2 possuem resultados superiores em tanto total de créditos recebidos como média das taxa de ocupação.

O Gráfico 10 mostra os cenários com uso de conjunto alto: preço base da vagas em 10 créditos e crédito máximo dos motoristas em 20 créditos. Logo, nos cenários com este conjunto, o valor da vaga começa em 10 créditos, o maior valor aceito por um *driver* é de 20 créditos e o menor é de 5 créditos. O valor de créditos máximos são decididos aleatoriamente neste intervalo para cada *driver* em sua criação.

Dentre os cenários da Figura, a maior taxa de ocupação durante as execuções das simulações foi de 49,5% e ocorreu com o uso da configuração 6 no dia 30. A menor ocupação, após o primeiro dia, encontram-se nas configuração 5 no dia 2 em 5% de ocupação. O maior preço de vaga entre os gráficos encontra-se no de configuração 2

**Gráfico 10 – Ocupações e preços em 200 vagas, 200 drivers e conjunto Alto**



**Fonte: Autoria Própria (2021).**

que alcançou 13 créditos durante o meio-dia. O menor valor de vaga é encontrado no gráfico de configuração 6 com o preço de 5 créditos em um período de demanda baixa, de madrugada durante tempestades.

A ocupação média na configuração 1 foi 25,21%, valor -12,22% menor do que no cenário sem módulo. O gráfico de ocupações mostra taxas predominantemente no



intervalo de 20 a 40% com quedas nos dias 5, 9 e 10. A maior taxa de ocupação durante a simulação foi de 37,5% no dia 25. A menor taxa após o primeiro dia foi de 7,5% no dia 2. O total adquirido em toda a simulação foi de 7090 créditos, valor -0,70% menor do que no sem módulo. O gráfico de preço apresenta comportamento semelhante ao do sem uso do módulo devido a ocupação manter-se no intervalo de demanda baixa. O maior preço das vagas foi de 10 créditos durante demanda baixa e o menor valor foi 9 créditos que ocorrer durante demanda muito baixa.

Na simulação do cenário com configuração 2, a ocupação média foi de 27,56% e -2,65% menor que a da sem módulo. O gráfico de ocupações mostra taxas predominantemente no intervalo de 20 a 40% e alcançado seu maior valor de 41,5% no dia 30. A menor taxa de ocupação, após o primeiro dia, foi de 6% no dia 2. Foram recebidos 7100,5 créditos ao fim do mês, total -0,48% menor que o valor do sem módulo. O maior preço atinge 13 créditos ao meio dia e o menor preço é de 7 créditos durante a madrugada.

A configuração 3 possui neste cenário a pior ocupação média de 28,18% e -0,40% menor que média do cenário sem módulo. As taxas de ocupações neste cenário permaneceram predominantemente entre 20 a 40% com quedas nos dias 5 e 12. A maior taxa de ocupação alcançada foi de 37,5% no dia 26 e a menor após o primeiro dia foi de 10,5% no dia 2. Neste cenário, o total de créditos recebidos foi 7489 que é 4,88% maior que o total do sem uso do módulo. O gráfico de preço apresenta as alterações influenciadas apenas pelo horário do dia devido a ocupação não ultrapassa 40%. O maior preço das vagas alcançado foi 12 créditos ao meio-dia e o menor preço atinge 8 créditos durante a madrugada.

A média das taxas de ocupação na configuração 4 foi 28,33%, 0,15% maior que a do sem módulo. O gráfico de ocupação contém a maioria de suas taxa acima de 20% e abaixo de 40%, com uma queda no dia 9 e uma subida no dia 30 com a taxa de 45%. A menor ocupação após o primeiro dia foi de 6,5% no dia 2. O total de créditos adquiridos foram 7234,5, 1,32% a mais que o total do cenário sem módulo. O maior valor das vagas durante a simulação foi 11,5 créditos durante demanda normal e clima ensolarado e o menor preço foi 8,5 créditos nos períodos de demanda baixa durante tempestades.

A configuração 5 possui uma média de ocupação de 21,64%, a pior média neste estacionamento e -30,75% menor do que no cenário sem uso de módulo. O gráfico de

ocupações possui taxas flutuando 20% com a pior queda entre todos cenários nos dias 9 a 12. A maior taxa de ocupação foi de 34,5% no dia 30 e a menor após o primeiro dia foi de 5,5% no dia 2. Este cenário acumulou 7673 créditos ao fim de sua execução, valor 7,46% maior que o total do sem módulo e o melhor resultado dentre os cenários da Figura. Por causa das baixas taxas de ocupação durante sua simulação, seus preços não foram influenciados pelas regras de demanda. O maior preço das vagas atingiu 11 créditos durante chuvas de tarde e o menor preço foi de 8,5 créditos em noites de céus limpos.

A ocupação média na configuração 6 foi de 33,57%, a melhor média e 18,67% maior que a média do cenário sem módulo. O gráfico de ocupação apresenta nos primeiros dois terços taxas no intervalo entre 20 a 40% e no fim do mês a ocupação circula os 40%. A maior ocupação foi 49,5% no dia 30 e, após o primeiro dia, a menor ocupação foi 8,5% no dia 2. Neste cenário foram recebidos um total de 6585 créditos, -8,43% menor que os créditos recebidos no cenário sem módulo e o pior resultado dentre os cenários. O maior preço alcançou 12 créditos nos períodos ensolarados ao meio-dia durante demanda normal e o menor foi de 5 créditos e ocorre durante tempestade de madrugada em demanda baixa.

O cenário sem módulo possui o característico gráfico de preço sem alterações ao decorrer dos dias. A sua média de ocupação foi de 29,29% com a maior taxa em 41,5% nos dias 22 e 26 e menor em 12% no dia 2. O total de créditos adquiridos foi 7530. A média de ocupação sem uso do módulo supera as configurações 1, 2, 3 e 5 e o total de créditos recebidos é maior que o das configurações 1, 2 e 6. Dentre os gráficos, apenas a configuração 4 possui ambas ocupação média e créditos recebidos maiores que os do cenário sem módulo.

### **6.3 Discussão dos Resultados**

Os resultados obtidos mostraram, ao comparar todos juntos, que os cenários sem utilização do módulo tiveram uma vantagem de 1,60% a mais no total de créditos recebidos e uma desvantagem de 2,09% menor ocupação média. Essas informações indicam que a utilização do módulo de precificação em si não é automaticamente benéfica e mostram como pequenas decisões nas regras de alteração de preço nos arquivos de configuração mudam consideravelmente os resultados finais do estacionamento.

Observando apenas os resultados dos créditos recebidos, aqueles cenários com valores totais menores utilizaram de configurações contendo descontos no valor da vaga em regras de demanda baixa. As piores configurações neste quesito foram a 4 e 6 com, respectivamente, 10% e 20% de desconto neste nível de demanda e adquiriram 5,57% e -8,89% menor de créditos ao comparar com os cenários sem módulo. A configuração 3 corrobora essa hipótese ao possuir descontos grandes nas regras de horário do dia, com 20% e 10% de redução do preço na madrugada e noite respectivamente, e mesmo assim alcançar 3,64% a mais de créditos recebidos que os cenários sem o módulo. As configurações que em geral adquiriram mais créditos ao fim de suas simulações foram as 3 e 5 com respectivamente 3,63% e 3,96% a mais de créditos recebidos comparados aos cenários sem uso do módulo. Ambas configurações não possuem descontos em suas regras de demanda baixa, optando por apenas não alterar o preço quando a taxa de ocupação mantém-se inferior a 40%.

Observando apenas os resultados das ocupações médias, as configurações que não possuem descontos em suas regras de demanda dispuseram de média menor do que as que descontam o preço durante baixa ocupação. A pior configuração neste quesito foi especificamente a 5, com média 3,65% menor comparado aos cenários equivalentes não usando o módulo. As configurações 2 e 3 também possuem médias menores que os cenários sem módulo, -1,60% e -1,58% respectivamente, ao não conter desconto nas regras de demanda ou, no caso da configuração 2, não abranger regras de demanda. Isso é possível ao considerarem que a ocupação é determinada por quantos *drivers* possuíram a quantia necessária para estacionar. Ao possuir um preço de vaga maior, devido à falta de descontos durante baixa ocupação, menos *drivers* aceitam o preço resultando em menores taxas de ocupação e uma média baixa ao fim da simulação. Aquelas configurações que em geral obtiveram as melhores médias de ocupação foram as 4 e 6 com médias respectivamente 8,22% e 7,20% maiores ao comparar com os cenários sem utilização do módulo. Ambas configurações realizam descontos no preço das vagas durante períodos de baixa demanda.

A configuração 1 possui a melhor combinação entre créditos recebidos e ocupação média. Nos cenários com seu uso, os créditos recebidos foram em média 0,262% menores comparados aqueles que não usaram o módulo, porém alcançaram ocupações médias 5,348% maior que estes. Esta configuração considera apenas a demanda do estacionamento e abrange um número maior de intervalos de demanda além de

descontar o preço de forma leve e apenas nos mais baixos níveis de requisição.

Observando especificamente maiores e menores resultados de cada cenário, apresentados na Tabela 12, pode ser concluído em quais ocasiões cada configuração prevaleceu. Considerando todos os quesitos, a melhor configuração é a 2 com 12 maiores resultados e a pior é a 6 com 15 menores resultados. Porém a maioria dessa soma encontra-se nos resultados de preço de vaga onde a configuração 2 possui 6 cenários com o maior preço de vaga alcançado e a configuração 6 possui 9 cenários com o menor valor de vaga proposto.

Ao desconsiderar o quesito de preço de vaga e de taxa de ocupação ocorre um empate entre as configurações 2 e 5 que possuem ambas 5 resultados maiores. A configuração 2 alcança essa soma ao possuir o maior total de créditos recebidos em 3 cenários e a maior média em 2. A configuração 5, por sua vez, possui 4 cenários com o maior total de créditos e 1 com a maior média. Também acontece um empate na pior configuração entre a 2 e 6, ambas com 6 piores resultados. A configuração 2 foi contém os menores totais de créditos recebido em 2 cenários e as menores médias em 4. A configuração 6, por outro lado, possui toda os seus 6 menores resultados no total de crédito recebido de 6 distintos cenários. Apesar de péssima no quesito de crédito recebido e conter os menores preços de vagas, a configuração 6 possui o maior número de cenários com a melhor média e taxa de ocupação, respectivamente 4 e 3 cenários.

**Tabela 12 – Comparação dos maiores e menores resultados de cada cenário**

2°CenáriosQuesitos	Preço da Vaga		Crédito Recebido		Taxa de Ocupação		Ocupação Média	
	Maior	Menor	Maior	Menor	Maior	Menor	Maior	Menor
100 / 50 / Baixo	C2: 6,5	C6: 2,5	SEM: 1115	C6: 899,25	C4: 29%	C5: 4%	C6: 18,65%	C1: 15,48%
100 / 100 / Baixo	C3: 7	C6: 2,5	C5: 2575,5	C2: 2016,75	C5: 62%	C5/SEM: 8%	C5: 41,70%	C3: 29,97%
100 / 100 / Médio	C3: 11,2	C6: 4,4	C2: 3397	C4: 3056,2	C6: 54%	C1/4: 7%	C6: 35,31%	C2: 28,57%
100 / 200 / Baixo	C5: 8	C6: 3	C5: 5389	C2: 4275,75	C1: 96%	C2: 17%	C4: 69,17%	C5: 56,09%
100 / 200 / Alto	C5: 15,5	C6: 5,5	C3: 8070	C4: 7239,5	C4: 100%	C2/3: 16%	C4: 66,05%	C2: 45,40%
200 / 50 / Baixo	C2: 6,5	C6: 2,5	C2: 1144	C6: 845,5	C1: 17%	C2: 0,5%	C1: 11,27%	C2: 5,76%
200 / 100 / Baixo	C2: 6,5	C6: 2,5	C2: 2285,5	C6: 1732,75	C3: 26%	C4: 2%	C2: 19,04%	C4: 12,26%
200 / 100 / Médio	C2: 10,4	C4: 4	C5: 3434	C6: 2265,2	C6: 29,5%	C3: 3%	C6: 18,04%	C2: 12,79%
200 / 200 / Baixo	C2: 6,5	C6: 2,5	C1: 4679,75	C6: 4030	C2: 51%	SEM: 7,5%	C2: 35,16%	C3: 25,53%
200 / 200 / Alto	C2: 13	C6: 5	C5: 7673	C6: 6585	C6: 49,5%	C5: 5%	C6: 33,57%	C5: 21,64%

**Fonte: Autoria Própria (2021)**

Os resultados das maiores e menores taxas de ocupação apresentados na Tabela 12 não indicam, superficialmente, uma correlação entre a configuração usada no cenário as taxas extremas de ocupação. Além do intervalo de ocupação predominante em cada cenário, a maior e menor taxa de ocupação durante uma simulação depende mais da aleatoriedade do comportamento do *driver*. As duas características do

estacionamento que influenciam a taxa de ocupação são o número aleatório de *drivers* desejando uma vaga e o preço dessa vaga. Quanto menor o preço da vaga, maior a chance de ser aceito e quanto maior a quantidade de *drivers* buscando vaga, maior a chance de um aceitar preço. Nas simulações destes cenários, cada *driver* determina um tempo de esperar até usar o estacionamento de 0 até 7 dias e um período de permanência variando de 1 hora até 3 dias. Considerando essas propriedades, aqueles cenários em que um número maior de *drivers* frequentemente escolhe esperar mais dias até usar e reutilizar as vagas combinado com períodos menores de uso dessas vagas resulta em menores taxas de ocupação durante sua simulação. Essas circunstâncias infortunas combinadas com preço de vagas altos diminuem mais a ocupação. Entretanto, este valor sozinho não foi mostrado afetar drasticamente a ocupação do estacionamento.

As configuração 1 adquiriu o maior resultado em 4 ocasiões, 1 em total de crédito recebido, 1 em média de ocupação e 2 em taxa de ocupação. Essa configuração conteve o menor resultado em 2 ocasiões, 1 em média de ocupação e a outra em taxa de ocupação. O seu cenário com o maior valor de créditos recebidos foi o de 200 vagas, 200 *drivers* e conjunto baixo. Este cenário possui, por definição, taxas regularmente no intervalo de 20 a 40% de ocupação devido ao mesmo número de vagas e *drivers*. Essa característica implica que, usando a configuração 1, o preço manteve-se em 5 créditos na maioria do mês e aumenta em 15% para 5,75 momento de demanda maior. Ao aumentar durante maiores ocupações, o resultado total é maior que o do cenário sem módulo e devido as taxas de ocupações não serem muito baixas ou altas, outros cenários não alcançam resultados melhores devido descontos que não ocorrem nesta configuração. O cenário em que adquiriu a maior média de ocupação foi o de 200 vagas, 50 *drivers* e conjunto baixo cuja taxas de ocupação circulam os 10% nunca atingindo 20%. Durante a simulação deste estacionamento, o preço da vaga durante todo o mês foi de 4,5 créditos por causa da baixa demanda. Devido a este valor, 0,5 menor que o valor do cenário sem módulo, a chance de um *driver* aceita-lo e usar uma vaga foi maior e resultou em maiores taxas de ocupação ao decorrer do mês. A configuração 6 possui valores de vaga menores em seu cenário equivalente e pode ser mostrado como exemplo de como o valor da vaga não afetou sinaficamente a ocupação nestes cenários. Ocorreram 2 cenários utilizando o arquivo de configuração 1 que possuem o menor resultado, 1 em média de ocupação e o outro em taxa de ocupação. O cenário com a menor média foi de 100 vagas, 50 *drivers* e conjunto baixo, que implica taxas

de ocupações principalmente no intervalo de 10 a 20%. Nestas taxas, o preço das vagas usando a configuração 1 é de 4,5 créditos durante a maior parte do mês com breves subidas aos 5 créditos durante ocupações maiores que 20%. O preço de vaga constantemente alto, em comparação aos cenários equivalentes, pode ser usado como motivo das baixas taxas de ocupação durante o mês. O caráter aleatório dos *drivers* deve ser considerado devido o cenário sem módulo conter média de ocupação 0,05% maior porém possui preço de vaga superior ou equivalente durante toda simulação.

A configuração 2 adquiriu o maior resultado em 12 ocasiões, 3 em total de créditos recebido, 2 em média das ocupações, 6 em preço da vaga e 1 em taxa de ocupação. Os cenários com maior total de créditos recebidos ocorrem em estacionamento de 100 vagas, 100 *drivers* em conjunto médio e 200 vagas, 50 e 100 em conjunto baixo. A configuração 2 considera apenas o horário do dia e mudanças na ocupação não afetam o preço das vagas. Logo, a configuração 2 adquiriu mais créditos nestas simulações devido a uma combinação de baixa ocupação em outros cenários, principalmente nos de 200 vagas, e aleatoriedade dos *drivers*, principalmente no de 100 vagas. Os cenários de maior média foram em estacionamentos de 200 vagas, 100 e 200 *drivers* em conjunto baixo. Normalmente a configuração 2 é uma das piores no quesito de ocupação, possuindo a menor média em 4 cenários. Pode-se assumir que estes maiores resultados são uma combinação entre os preço de vagas comparativamente mais baixos e consistência no valor das taxas de ocupação ao decorrer do mês, sejam elas baixas ou altas. A configuração 2 possui o menor resultado em 9 cenários, 2 destes em créditos recebidos, 4 em média de ocupação e 3 em taxa de ocupação. Os cenários com os menores totais de créditos adquiridos ao fim de suas execuções foram em estacionamentos de 100 vagas, 100 e 200 *drivers* em conjunto baixo. Nesta características, que implicam intervalos de ocupação respectivamente em 25 a 50% e 40 a 80%, outras configurações que observam a demanda adquirem mais créditos devido a maior ocupação. Os cenários com as menores médias de taxas de ocupação ocorreram em estacionamentos de 100 vagas, 100 e 200 *drivers* respectivamente em conjuntos médio e alto e de 200 vagas, 50 e 100 *drivers* em conjuntos respectivamente baixo e médio. Como indicado nos maiores resultados, as taxas de ocupação em simulações da configuração 2 permanecem em uma faixa consistente a maioria do tempo. Uma possível explicação é equilíbrio entre o maior preço das vagas durante o dia e os descontos durante a noite. Essa consistência abriu, nestes cenários, caminho para a

variação em outras configurações resultado em taxas de ocupações maiores.

A configuração 3 adquiriu o maior resultado em 4 ocasiões, 1 em crédito total recebido, 2 em preço da vaga e a última em taxa de ocupação. Os cenários com maior quantidade de créditos recebidos foi no estacionamento de 100 vagas, 200 *drivers* e conjunto alto. Este é o único cenário em que a configuração 3 possui o maior resultado, porém ela consistentemente encontra-se entre nas 3 configurações com maior total, principalmente nos estacionamentos com 200 vagas. Especificamente de maior valor, este resultado é alcançado pela combinação troca de preço horário, eficiência mostrada pela configuração 2, e consideração de demanda neste estacionamento com faixa de ocupação de 40 a 60%. Essa configuração retem os piores resultados em 4 cenários, 2 em média de ocupação e 2 em taxa de ocupação. Os cenários com as menores médias ocorreram nos estacionamentos de 100 vagas, 100 *drivers* em conjunto baixo e de 200 vagas, 200 *drivers* também em conjunto baixo. A média das taxas de ocupação da configuração 3 estão frequentemente entre as 3 menores, especialmente nos cenários contendo 100 vagas. A configuração 3 possui características semelhantes à 2 porém se mostra pior quando a ocupação passa para a faixa de demanda média causando um aumento no preço e redução da probabilidade de um *driver* aceitá-lo.

A Tabela 12 mostra 4 ocasiões onde a configuração 4 alcançou o maior resultado, 2 desses em média de ocupação e os outros 2 em taxa de ocupação. Os cenários com as maiores médias ocorreram em estacionamentos de 100 vagas, 200 *drivers* de conjunto baixo e alto. Nestes estacionamentos a faixa comum de ocupação encontra-se entre 40 e 60%, porém o uso de regras de demanda e condição climática resultaram em taxas que alcançam a lotação máxima das vagas. Especificamente nestes cenários, a configuração 4 possui uma mudança de preço diferenciada em que passa dias com preços altos seguido por dias de preço baixos e vice-versa. Essa característica ocorre de forma amenizada em todos seus cenários e existe devido a particularidades das regras de mudança de preço por condição climática sem influência do horário do dia. As regras de demanda, neste caso, servem como um meio de aumentar a amplitude da diferença, mais evidente nestes cenários com maiores taxas de ocupação. A configuração 4 obteve a melhor média em apenas 2 dos cenários, porém está na maioria deles com segundo melhor resultado. Esta configuração possui os menores resultados em 6 ocasiões, 2 em total de créditos recebidos, 1 em média de ocupação, 2 em preço da vagas e 2 em taxa de ocupação. Os cenários com os menores valores de créditos recebidos ocorrem em

estacionamentos de 100 vagas, 100 e 200 *drivers* respectivamente em conjunto médio e alto. Devido as características explicadas previamente, essa configuração encontra-se entre os 3 menores totais de créditos em todos seus cenários com exceção de um. A pouca quantidade de créditos adquiridos ocorre principalmente por causa do desconto de preço durante demanda baixa. O cenário com a menor média de ocupação ocorre em um estacionamento de 200 vagas, 100 *drivers* em conjunto baixo. A exceção entre as taxas de ocupações encontradas em outros cenários desta configuração. Pode-se afirmar que as características randômicas dos *drivers* tiveram grande influência na baixa ocupação apesar do baixos preços de vagas ao decorrer de toda simulação do cenário.

Ocorreram 8 momentos onde a configuração 5 possuiu os maiores resultados, 4 em créditos recebidos, 1 em média de ocupação, 2 em preço de vaga e 1 em taxa de ocupação. Os cenários com os maiores totais de créditos recebidos ocorreram em estacionamentos de 100 vagas, 100 e 200 *drivers* em conjunto baixo e de 200 vagas, 100 e 200 *drivers* respectivamente em conjunto médio e alto. Além de raramente possuir o menor resultado entre cenários semelhantes, com a exceção de 2 cenários, o total de créditos recebidos nessa configuração estão sempre entre os dois melhores. Por causa da falta de descontos, apenas 5% de redução durante céus limpos ou a noite, os preços das vagas são constantemente altos quando comparados com os de outras configurações no mesmo cenário. O estacionamento em que a configuração 5 adquiriu a maior média de ocupação possui 100 vagas, 100 *drivers* e conjunto baixo além de possuir ocupações decentes em conjunto médio. Nestes dois cenários, essa configuração observa aumento em suas taxas de ocupações enquanto o comportamento em seus outros apresenta platitude em relação ao número de *drivers* usando as vagas. A sua maior média é alcançada devido a seu comportamento de manter-se em uma faixa constante de ocupação precedido por um fluxo maior de *drivers* desejando por vaga. A configuração 5 possui 2 cenários com a menor média de ocupação e 3 com a menor taxa de ocupação. Em geral, a configuração 5 contém as menores taxas de ocupação, mas especificamente nos estacionamentos de 100 e 200 vagas, 200 *drivers*, em respectivamente conjunto baixo e alto foram alcançadas as menores taxas. Devido seu atributo de estagnação, enquanto outras configurações possuem crescimento de suas taxas ao decorrer do mês, a configuração 5 permanece em ocupações baixas. Nestes menores casos ocorreram grandes quedas de número de *drivers* desejando vagas e a ocupação não recupera-se posteriormente.



A configuração 6 apresentou os maiores resultados em 7 simulações, 3 destes em média de ocupação e os 4 restantes em taxa de ocupação. Os estacionamentos dos cenários com maiores médias possuíram 100 vagas, com 50 e 100 *drivers* respectivamente em conjunto baixo e médio e 200 vagas, com 100 e 200 *drivers* respectivamente em conjunto médio e alto. Na maioria de seus cenários, a configuração 6 possui taxas de ocupações superiores devido a seus constantes preços de vaga baixos. Este atributo aumenta a chance de cada *driver* aceitar o preço e logo gera maiores taxas de ocupação e uma média maior ao fim do mês. Essencialmente, esta configuração possui resultados contrários aos da configuração 5, onde este possui maiores médias e aquela possui maiores créditos recebidos. A diferença entre as duas configurações encontra-se: no número de regras em cada categoria, a 5 possuindo 3 regras em cada e a 6 possuindo 5 de horário, 3 de demanda e 4 de condição climática; e na quantidade de descontos nestas regras, a 5 contendo leves descontos durante temporais ou durante a noite, enquanto a 6 possui pequenos descontos durante temporais ou grandes descontos durante baixa demanda ou durante noite e madrugada. Em 12 ocasiões a configuração 6 retem o menor resultado, 6 em total de créditos recebidos e 9 em preço de vaga. A simulação com a configuração 6 resultou no menor total de créditos recebidos em todos os estacionamentos de 200 vagas e no de 100 vagas com 50 *drivers* em conjunto baixo. A quantidade de créditos adquiridos é diretamente afetado pelo preço das vagas ao decorrer da simulação e na configuração 6 esses são constantemente os menores entre todos os cenários equivalentes. Até mesmo com a maior chance de *drivers* aceitarem o preço, e resultando em taxas altas de ocupação, a soma dos créditos ao fim do mês nos cenários da configuração 6 são frequentemente os mais baixos.

Os resultados dos cenários sem utilização do módulo encontram-se predominantemente no meio das colocações com as exceções de um resultado maior de total de créditos recebidos e dois resultados menores de taxa de ocupação. O estacionamento do cenário com maior total de créditos recebidos contém 100 vagas, 50 *drivers* criados e usou conjunto baixo. Em estacionamentos onde a ocupações encontram-se no intervalo de 10 a 20%, os cenários mais prováveis de adquirir mais créditos são aqueles que não consideram a demanda durante a alteração do valor das vagas. Essas condições favorecem tanto a não usar do módulo de precificação como a configuração 2 que observa apenas o horário do dia. As taxas de ocupações nos cenários sem troca de preço dependem completamente das características randômicas dos agentes *drivers*.

Os gráficos nestes cenários mostram o atributo desejado na criação das características dos *drivers*, ondas de ocupação que podem estar criando aumento ou queda ao decorrer do mês.

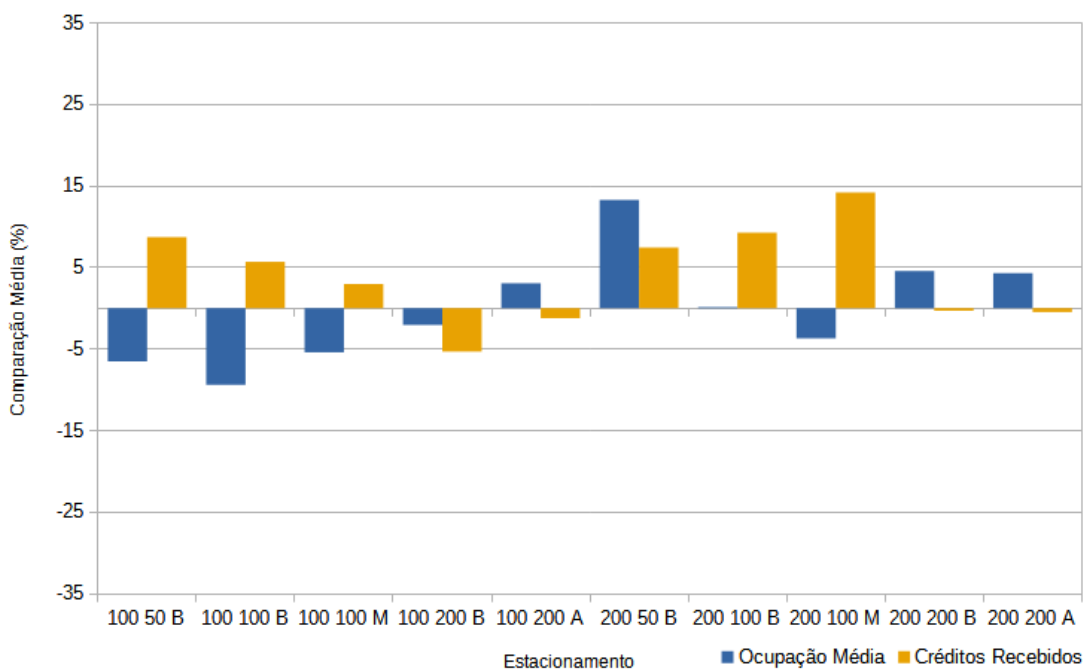
### 6.3.1 Vantagens e Desvantagens

As simulação dos cenários sem utilização do módulo de precificação servem como controle e identificação das decisões de agentes sem a influência da alteração do preço das vagas. Observando as taxas de ocupação nestes cenários, a decisões dos *drivers* contém um padrão de crescimento da ocupação durante o mês atrelado a quedas repentinas em momentos randômicos onde muitos agentes recusam o preço ou decidem usar o estacionamento em outro dia. As figuras nesta subseção mostram gráficos contendo as porcentagens médias de quanto os resultados dos cenários sem uso do módulo e de cada configuração se comparam entre si nos mesmos estacionamentos. Cada elemento do eixo  $x$  destes gráficos representa o estacionamento contendo em sequência a quantidade de vagas, número de *drives* e conjunto de preços. O eixo  $y$  representa a média das diferenças em porcentagem do cenário do gráfico com os semelhantes de outras simulações. Estes dados foram observados usando os resultados da Tabela 11.

O Gráfico 11 mostra que não usar o módulo resulta em ocupações médias maiores em estacionamentos com número alto de vagas e que adquire maior total de créditos em menores quantidades de *drivers*. Em comparação com as configurações, um uso ideal para estacionamento sem o módulo em questão de crédito recebido deve conter baixa demanda base ou número de *drivers* pequenos em relação ao total de vagas disponíveis. Se o foco é maior ocupação média, um estacionamento sem módulo não proporciona nenhuma vantagem inata.

Sabendo que nestas simulações a ocupação das vagas tem característica volátil com tendência a crescimento, a seguir são descritas as vantagens e desvantagens de cada configuração.

A configuração 1 mostrou-se bastante balanceada em suas taxas de ocupações em comparação com seus créditos adquiridos ao decorrer de suas simulações e o Gráfico 12 mostra este aspecto. Apesar de uma grande exceção na ocupação média no estacionamento de 200 vagas, 50 *drivers* e conjunto baixo os resultados de seus cenários

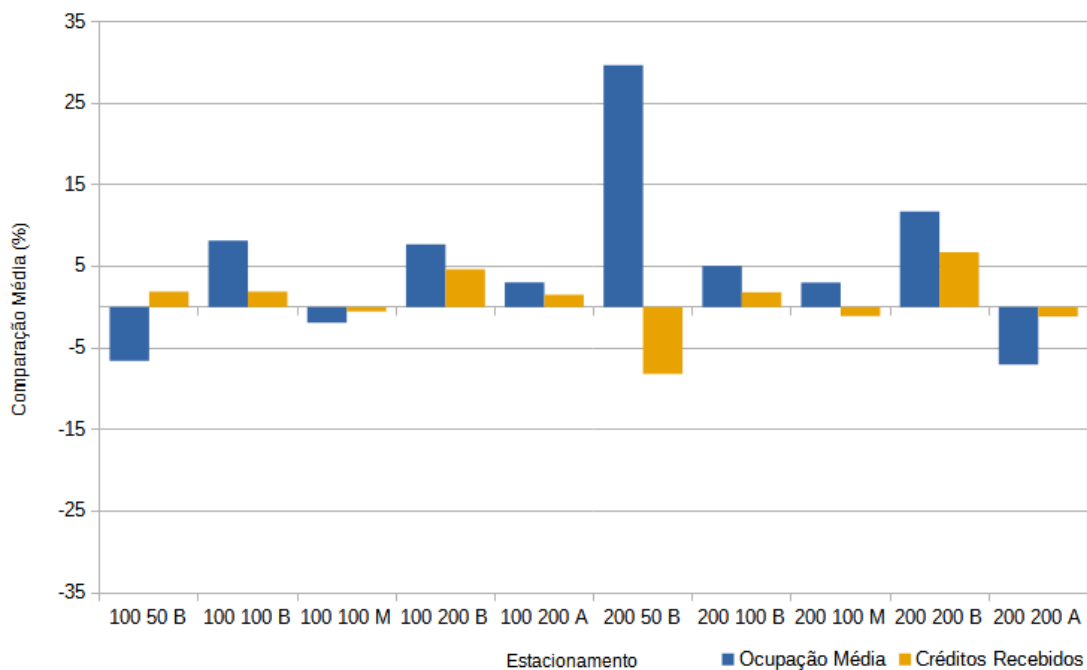
**Gráfico 11 – Comparação dos resultados dos cenários sem uso do módulo**

**Fonte: Autoria Própria (2021).**

estiveram em geral na faixa de -5 a 5% de diferença comparado as outras configurações. As vantagens em seu funcionamento encontram-se em ocupações das vagas maiores sem muita redução de créditos ao fim de suas simulações. A desvantagem é que, em geral, o aumento de ocupação é desprezível em ocasiões de baixa demanda. O uso ideal desta configuração está em estacionamentos com histórico de alto uso de suas vagas ou aqueles que previamente não utilizavam de nenhum método de precificação dinâmica.

A configuração 2 tem foco em ganhar mais créditos em horários específicos. Os preços das vagas nesta configuração constantemente são os maiores devido ao alto incremento durante o período definido na sua regra do meio-dia. Um resultado desta característica é uma aparente alta rotatividade de *drivers* usando as vagas, indicado pela relativa estabilidade das taxas de ocupação. As desvantagens encontram-se em seu uso em locais contendo quantidade baixa de vagas e em sua volatilidade entre simulações onde ela pode estabilizar em taxas altas ou baixas de ocupação. O Gráfico 13 expõe esta instabilidade visualmente. Os resultados mostram que um maior número de vagas aumenta o ganho de créditos porém desestabiliza ainda mais as taxas de ocupação entre simulações. O uso ideal desta configuração, quando o objetivo é adquirir mais

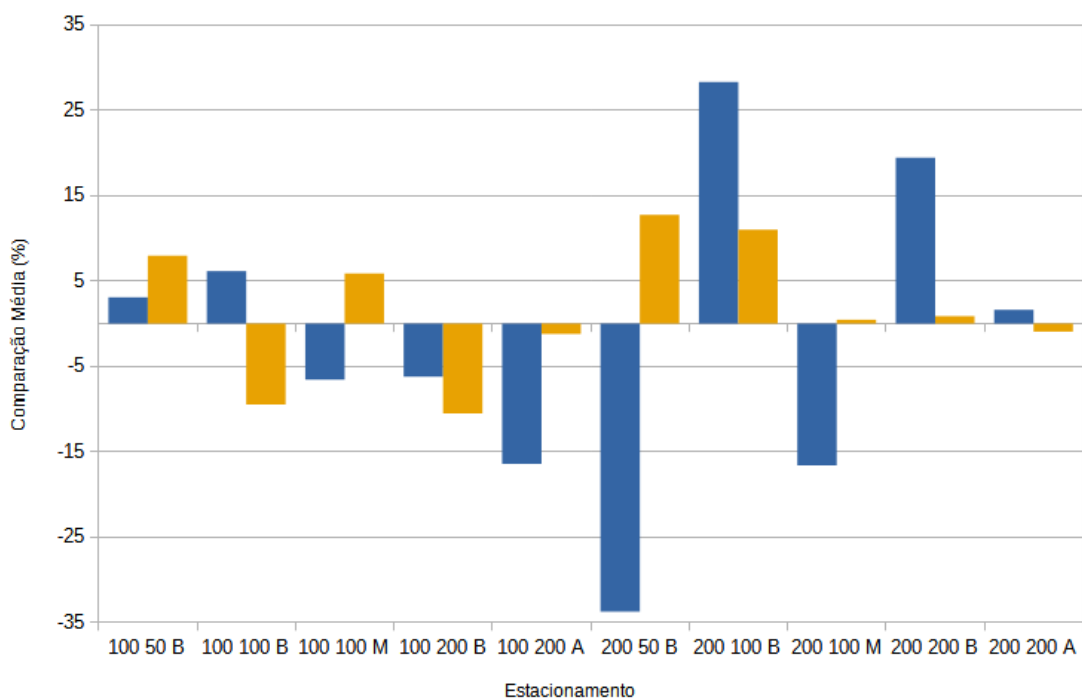
Gráfico 12 – Comparação dos resultados da configuração 1



Fonte: Autoria Própria (2021).

créditos, está em estacionamentos com histórico de maiores demandas em horários específicos do dia e que contenham um número alto de vagas disponíveis.

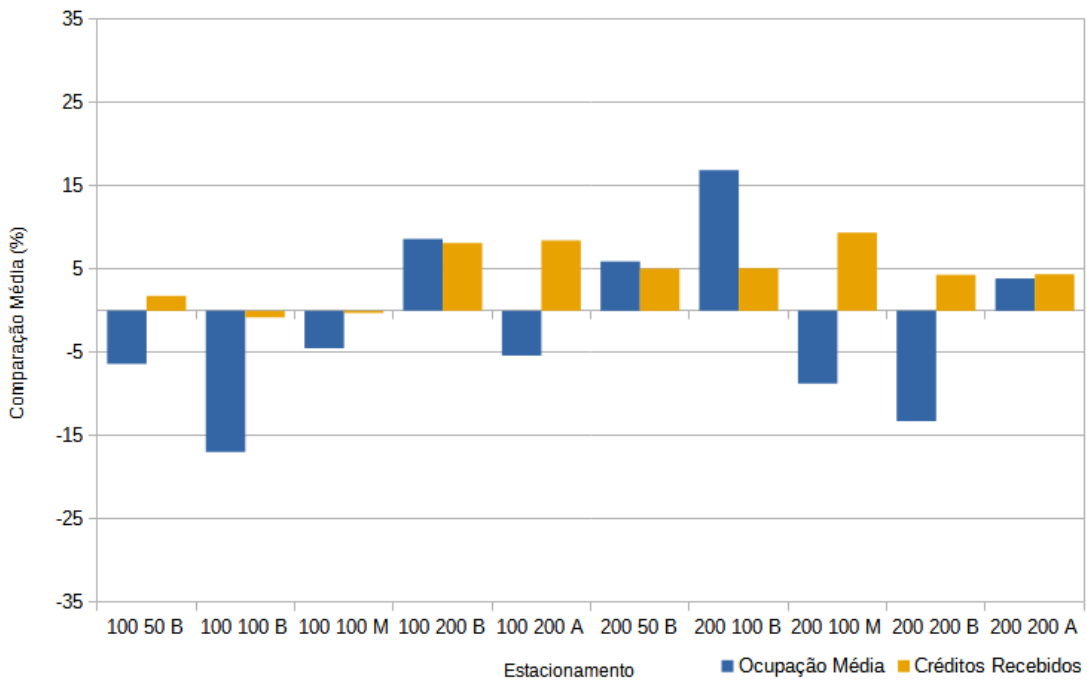
Gráfico 13 – Comparação dos resultados da configuração 2



Fonte: Autoria Própria (2021).

A configuração 3 aparenta usufruir diretamente de maiores demandas base em questão de créditos recebidos. A combinação de regras de demanda sem descontos e de horário, com aumento do preço ao meio dia, efetivamente retornam mais créditos ao fim do mês e nos piores casos, perdem menos que outras configurações. Está configuração é essencialmente um versão menos volátil entre cada simulação, porém com maior instabilidade em suas taxas de ocupação comparada com a configuração 2. O Gráfico 14 contém comparações dos resultados entre as configurações e mostra a abrangência de mais créditos recebidos em estacionamentos contendo maior demanda base. O melhor uso desta configuração é em vagas exclusivas onde o objetivo é cobrar caro pelo seu uso, sabendo que resultará em menores ocupações.

**Gráfico 14 – Comparação dos resultados da configuração 3**

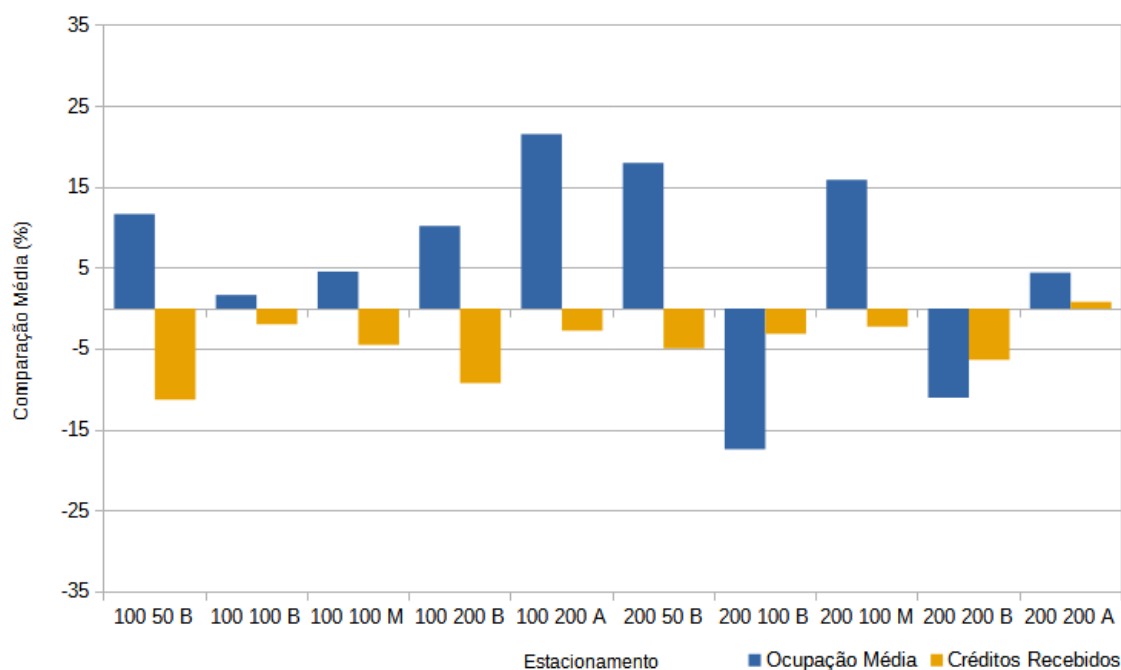


**Fonte: Autoria Própria (2021).**

A configuração 4 reduz seus ganhos de créditos e aumenta a probabilidade de maior ocupação. A estabilização dos preço de vagas em valores baixos durante múltiplos dias aumenta a chance de *drivers* utilizarem o estacionamento. Essa propriedade de permanência, devido a característica pseudorrandômica das regras de condição climática, traz maiores ocupações durante baixo preço e recupera um pouco de créditos durante período de maior preço. O Gráfico 15 contém os seus resultados comparativos e mostra que em locais contendo número maior de vagas, a configuração 4 apresenta

maior instabilidade de ocupação quando comparado com a versão mais extrema da 6. O uso ideal desta configuração ocorre quando o objetivo é atrair clientes para o local do estacionamento e ocupando o máximo das vagas o tempo todo.

**Gráfico 15 – Comparação dos resultados da configuração 4**

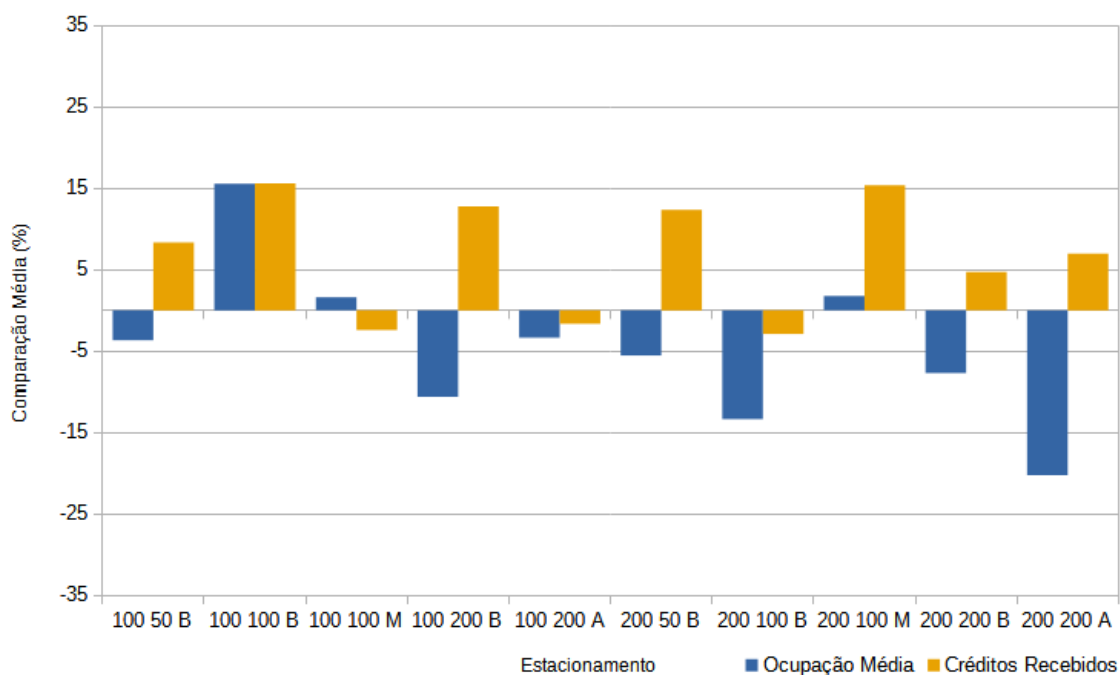


**Fonte: Autoria Própria (2021).**

A configuração 5 funciona por cobrar valores altos em cada uso das vagas resultando em ocupações menores. A vantagem desta configuração é total maior de créditos recebidos não importando o tamanho do estacionamento onde é usado. Seus poucos descontos e a propriedade estabilizadora das regras de condição climática mantém o preço sempre altos. O Gráfico 16 mostra que em seus piores cenários, o total de crédito recebido ficou em torno de apenas -3,5% menor que outras configurações. A desvantagem é sua utilização com o intuito de aumentar a ocupação. O alto valor das vagas diminui drasticamente a probabilidade de *drivers* aceitarem o preço. Um estacionamento ideal para esta configuração são aqueles de eventos em lugares exclusivos e ou isolados que cobram mais caro por essa vaga especial.

A configuração 6 é uma versão mais extrema da 4 ao possuir preços de vagas constantemente menores que os de qualquer outra configuração. Nos cenários com as menores ocupações, a sua média nunca ficou -5% menor que as de outras configurações, como visto no Gráfico 17. Utilizando descontos em todas suas categorias de

Gráfico 16 – Comparação dos resultados da configuração 5



Fonte: Autoria Própria (2021).

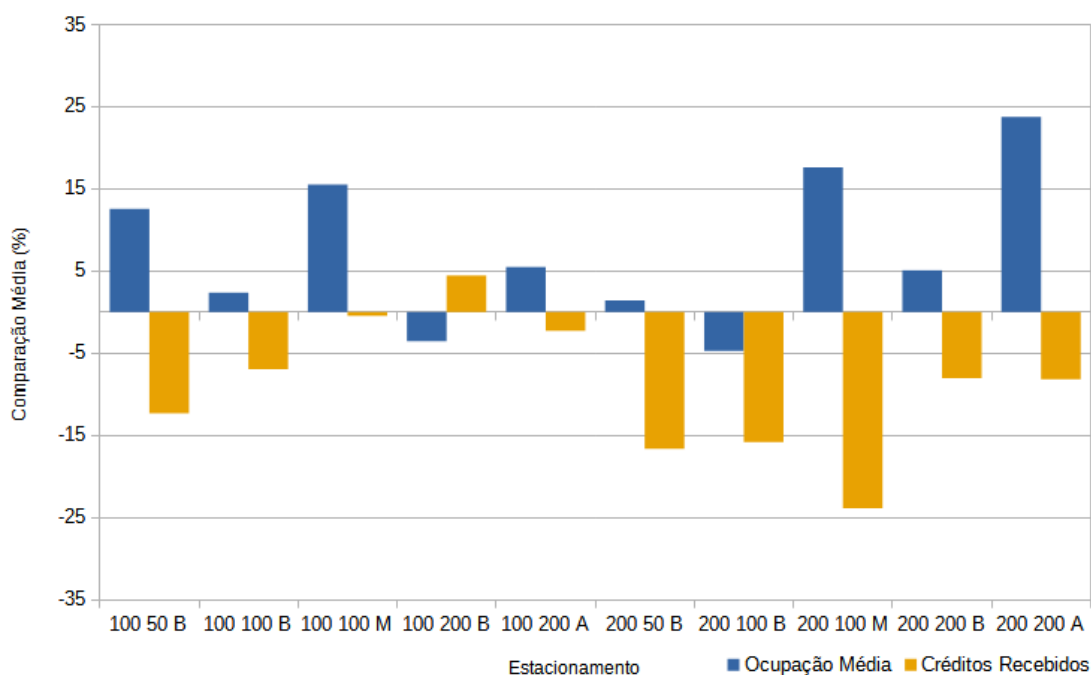
regras, essa configuração mantém sua ocupação constantemente em crescimento na maioria de suas simulações. A clara desvantagem é usa-la com o objetivo de adquirir créditos com uso do estacionamento. O gráfico também mostra como em estacionamentos com demanda base baixa, o total de créditos adquiridos é ainda menor. Esta configuração é melhor usada quando o objetivo principal é manter preço de vagas baixos com o intuito de atrair um maior número de clientes ao local do estacionamento.

### 6.3.2 Melhores Casos de Uso

Observando os resultados descritos anteriormente e considerando apenas as configurações apresentadas, a seguir são propostos exemplos dos melhores casos de uso para proprietários, clientes e a cidade onde encontra o estacionamento. Para decidir qual ocasião é melhor para cada objetivo foi determinado que: o proprietário deseja o maior total de lucro ou créditos recebidos; o cliente deseja os menores preços de vaga; e a cidade almeja maiores ocupações das vagas.

O melhor cenário para o proprietário do estacionamento contém 100 vagas, foi usado por 200 *drivers* recorrentes durante o mês e possui preço base das vagas de 10 créditos e disponibilidade máxima dos *drivers* de 20 créditos. Nesta ocasião o

Gráfico 17 – Comparação dos resultados da configuração 6



Fonte: Autoria Própria (2021).

estacionamento adquiriu 8070 créditos ao fim do mês, o maior entre todos os cenários. Porém, em estacionamentos de 100 vagas, a melhor configuração é a 5 que rende em média 7,76% mais créditos ao fim de um mês nesta quantidade de vagas. Já em estacionamentos com 200 vagas disponíveis, a configuração 2 alcança o maior total de créditos recebidos, porém apenas 0,6% e 1,5% a mais que os das configurações 3 e 5 respectivamente.

Para o cliente, a melhor situação ocorre em qualquer uso do arquivo de configuração 6, principalmente quando o preço base da vaga é de 5 créditos. Os mais baixos preços das vagas durante os meses executados nesta configuração são 22,14% menores que o de outras. Especialmente em estacionamentos contendo muitas vagas e baixo volume de motoristas, os preços determinados pelas regras deste arquivo estão constantemente abaixo dos outros e seus maiores valores costumam ser os menores em outras configurações.

Considerando as necessidades da cidade, a situação com a maiores taxas de ocupação ao decorrer do mês e conseqüentemente maior média usou a configuração 4 em um estacionamento de 100 vagas, com 200 *drivers* recorrentes, preço base de 10 créditos e máximo disponível de 20 créditos para cada *driver*. Sabendo que o objetivo



nestes casos é diminuir o tempo de circulação do motorista, para um volume de 50 *drivers*, a configuração 4 é a melhor escolha, porém contém médias apenas 4,74% maiores que da configuração 6. Para volumes de 100 *drivers*, a configuração 6 passa a ser a melhor escolha ao conter médias 5,80% maiores que da configuração 4. Nas ocasiões de 200 *drivers*, a configuração 4 volta a ser melhor, dessa vez com médias somente 4,06% maiores.

Procurando por um apuramento balanceado, encontra-se que os resultados das simulações da configuração 1, em média, são os mais equilibrados entre todas as execuções. Os valores da configuração 3 são os mais moderados especificamente apenas os cenários de 50 *drivers* e ao considerar apenas os cenários de 200 vagas, as simulações sem uso do módulo de precificação são mais balanceadas. Ambos casos devem se a grande exceção do cenário no estacionamento de 200 vagas e 50 *drivers* da configuração 1 que abriu a oportunidade de outros resultados serem mais balanceados.

## 7 CONCLUSÃO

O desenvolvimento de estacionamentos inteligentes e tecnologias análogas têm foco em reduzir os problemas de trânsito e mobilidade dos habitantes de grandes centros urbanos. O método de minimizar estes problemas envolve a automatização das tarefas de um estacionamento que englobam o gerenciamento das vagas. Adjacente ao controle das vagas, a modificação de seu preço também possui influência sobre o funcionamento do estacionamento. Este trabalho apresentou o desenvolvimento e avaliação de um algoritmo modular de precificação dinâmica do preço de vagas.

O projeto MAPS é uma colaboração de várias soluções para estacionamentos utilizando de Sistemas Multiagentes. O método de alteração de preços deste trabalho é uma extensão de outros trabalhos que abrangem desde a capacidade de alocação de vagas usando critério de confiança de cada agente até um sistema de gerenciamento aberto e descentralizado. Os Estacionamentos Inteligentes desenvolvidos e aquele que foi usado como alvo das simulações do módulo foram criados com a utilização do *framework* JaCaMo devido sua intercalação das linguagens JASON, Cartago e Moise que promove maior facilidade na comunicação e interação entre agentes e os componentes do estacionamento.

Com o objetivo de avaliar a influência do preço e sua alteração na operação de um estacionamento, foi desenvolvido um módulo que recebe características do ambiente deste estacionamento e usando regras pré-estabelecidas, modifica o preço de todas as vagas. Foram desenvolvidos e executados 60 cenários usando 6 arquivos de configurações contendo as regras de mudança de preço e 10 cenários de simulações sem o uso do módulo de precificação. O foco da comparação dos resultados foram nos totais de créditos recebidos e ocupações médias com intuito de realizar um contraste entre cada configuração e o estacionamento sem troca de preço.

Os resultados mostraram que o módulo de precificação possui a capacidade de modificar o funcionamento do estacionamento de forma significativa. As diferenças dos resultados entre o uso das configurações 5, que resulta em maior valor de créditos e a 6, que proporciona maior ocupação, mostra como pequenas diferenças na definição das regras alteram o desempenho do estacionamento. Foi observado que não usar módulo pode resultar em melhores soluções, porém cada configuração possui uma característica que realiza melhor com o uso da precificação dinâmica. Em sua maioria, os

resultados das configurações apresentam valores oscilantes devido suas características pseudo aleatórias, e seu uso é melhor em ambiente específicos e pré-definidos, com a exceção da configuração 1 que possui caráter bastante balanceado.

Além disso, como o algoritmo do módulo de precificação funciona como uma árvore de decisão programável, este trabalho contém abrangência em outras áreas. Com o devido ampliamto do código e definição de diferentes regras, o módulo pode ser usado com um banco de tarefas decididas externamente do sistema.

## 7.1 Trabalhos Futuros

Este projeto de modificação dinâmica de preço abre caminho para testes mais extensivos e novas tecnologias sobre a manipulação do preço. Além da sua utilização em outras áreas fora de estacionamentos inteligentes, destacam-se as seguintes propostas:

- Desenvolvimento de um sistema multiagente com uma arquitetura diferente. Seja negociações de vagas descentralizado como em (DUCHEIKO; BORGES; ALVES, G. V., 2018) ou um modelo holônico reconfigurável como em (CASTRO, L. F. S. d. *et al.*, 2018)).
- Acréscimo e alteração de atributos aos cenários de simulação ao otimizar o código para aceitar volume maior de motoristas e vagas.
- Ampliação da generalização das regras do módulo, abrindo espaço para criação de novas categorias para uso em tipos mais específicos de estacionamentos e em sistemas fora desse foco.
- Aplicação de um método de escolha dinâmica e automática do arquivo de configuração adequado a situação do estacionamento.
- Implementação dos modelos teóricos de precificação dinâmica apresentados.
- Desenvolvimento de um sistema de previsão de preços das vagas, usando o histórico do estacionamento. Os preços previstos podem então serem utilizados em um método de reserva de vagas (MENDES, 2021).
- Implementação de um algoritmo de *machine learning* para previsão de demanda de acordo com condição climática, tempo, histórico, etc...
- Aplicação dos métodos desenvolvidos em um estacionamento embarcado com fim de encontrar resultados mais próximos da realidade.

## 7.2 Limitações do Trabalho

Devido à implementação de um relógio interno para controle do tempo da simulação do estacionamento, inúmeros problemas surgiram durante a maior aceleração necessária de cada operação do sistema. O relógio foi criado com o intuito de controlar a velocidade da passagem do tempo do estacionamento e executar suas funções durante grandes intervalos de forma acelerada. Para os testes realizados, cada segundo em tempo real representa cerca de uma hora em tempo de simulação. Logo para simular 30 dias no sistema, são necessários aproximadamente 12 minutos de execução.

Essa aceleração do tempo na simulação cria a necessidade de funcionalidades do estacionamento serem executadas indevidamente mais rápidas. Algumas dessas operações, como a entrada e saída de veículos e a reorganização das vagas ficaram propensas a erros graves que resultam na perda da simulação e necessidade de nova tentativa da execução. Em tempo real essas operações possivelmente teriam vários segundos livres para sua execução, porém no sistema elas devem ser completadas em milésimos de segundo. Isto não foi um problema em cenários com poucos *drivers* simultâneos, porém o incremento destes aumenta drasticamente a possibilidade de um erro grave durante a execução.

Esses erros graves são de forma geral problemas de sincronização em relação à utilização da vaga, que é um recurso comum entre todos os agentes. Em circunstâncias onde um agente não terminou sua operação sobre a crença da vaga, essa pode ter sido perdida ou criada indevidamente, resultando em um número total de vagas diferente do criado no começo da simulação.

Foi previsto durante o desenvolvimento do sistema que 300 *drivers* simultâneos é demais para o código e as execuções provaram essa hipótese. As simulações com 300 ou mais *drivers* constantemente deram erros antes de chegar a 30 dias. Nestes cenários os erros ocorreram principalmente durante a entrada e saída de veículos. Os erros causaram a criação ou remoção inadequada de crenças de vagas, ou de setores ao vários agentes passarem paralelamente pelos mesmos recursos. Foi desenvolvido um método que mitiga este problema em cenários com menos agentes, porém não foi possível criar um procedimento que protege o recurso consistentemente.

O número alto de *drivers* também causa erros durante o plano de reorganização de vagas. Por mais que eles sejam impedidos de atuar no estacionamento durante a

execução da reorganização, operações internas dos *drivers*, como verificar o relógio, causam um gargalho em todo o sistema ao decorrer dos dias de simulação o que pode impedir o plano de reorganização de terminar em seu intervalo de 2 segundos. Não terminando neste espaço de tempo, as crenças de vagas e setores que não foram finalizadas podem ser usadas por *drivers* e causar conflito com o algoritmo de reorganização ao criar crenças repetidas.

Fora limitações no desenvolvimento do sistema descritas previamente, faltou também execuções de um maior número combinações das características de ambiente. Tipos evidentes faltantes são um arquivo de configuração que observa apenas as condições climáticas, um com apenas regras de demanda contendo mais descontos no preço e uma combinação de condição climática durante os horários do dia. Devido problemas de execução dos cenários previamente determinados, não foi viável incrementar estes e outros arquivos de configuração.

## REFERÊNCIAS

- ALIEDANI, A.; LOKE, S. W. Cooperative car parking using vehicle-to-vehicle communication: An agent-based analysis. **Computers, Environment and Urban Systems**, Elsevier, 2018.
- AMINI, M. H.; MOGHADDAM, M. P.; KARABASOGLU, O. Simultaneous allocation of electric vehicles' parking lots and distributed renewable resources in smart power distribution networks. **Sustainable Cities and Society**, Elsevier, v. 28, p. 332–342, 2017.
- ASADZADEH, L. A local search genetic algorithm for the job shop scheduling problem with intelligent agents. **Computers & Industrial Engineering**, Elsevier, v. 85, p. 376–383, 2015.
- BARILE, F. *et al.* Evaluating the Social Benefit of a Negotiation–Based Parking Allocation. In: SPRINGER. INTERNATIONAL Conference on Practical Applications of Agents and Multi-Agent Systems. [S. l.: s. n.], 2015. P. 15–26.
- BENENSON, I.; MARTENS, K.; BIRFIR, S. PARKAGENT: An agent-based model of parking in the city. **Computers, Environment and Urban Systems**, Elsevier, v. 32, n. 6, p. 431–439, 2008.
- BOER, A. V. den. Dynamic pricing and learning: historical origins, current research, and new directions. **Surveys in operations research and management science**, Elsevier, v. 20, n. 1, p. 1–18, 2015.
- BORDINI, R. H.; HÜBNER, J. F.; WOOLDRIDGE, M. **Programming Multi-Agent Systems in AgentSpeak Using Jason (Wiley Series in Agent Technology)**. [S. l.]: John Wiley & Sons, 2007. ISBN 0470029005.
- BOTELHO, P. W.; BORGES, A. P.; ALVES, G. V. Proposta de implantação de um sistema ciber-físico para um Smart Parking baseado em agentes inteligentes, 2019.
- BOTELHO, P. W.; ALVES, G. V.; BORGES, A. P. Desenvolvimento de Aplicação para um Estacionamento Inteligente via Computação em Nuvem baseada em Agentes e Sistema Ciber-Físico. In: PROCEEDINGS of the 14th Workshop-School on Agents, Environments, and Applications (WESAAC 2020). [S. l.]: UTFPR, jul. 2020. v. 1, p. 185–194. ISBN 2177-2096. DOI: 10.5281/zenodo.4037413. Disponível em: <https://doi.org/10.5281/zenodo.4037413>.
- CAO, B. *et al.* Dynamic Pricing for Resource Consumption in Cloud Service. **Wireless Communications and Mobile Computing**, Hindawi, v. 2018, 2018.
- CARTAGO. **CARtAgO**. 2006. Disponível em: <http://cartago.sourceforge.net/>. Acesso em: 5 jul. 2019.

CASTRO, L. F. S. *et al.* Integrando Sistemas Multi-Agentes Embarcados, Simulação Urbana e Aplicações de IoT. In: PROCEEDINGS of the 14th Workshop-School on Agents, Environments, and Applications (WESAAC 2020). [S. l.]: UTFPR, jul. 2020. v. 1, p. 165–176. ISBN 2177-2096. DOI: 10.5281/zenodo.4037413. Disponível em: <https://doi.org/10.5281/zenodo.4037413>.

CASTRO, L. F. S. d. *et al.* **Desenvolvimento de uma arquitetura multiagente holônica reconfigurável aplicada ao cenário de smart parking**. 2018. Diss. (Mestrado) – Universidade Tecnológica Federal do Paraná.

CASTRO, L. F. S. d. **Modelagem e implementação de um sistema multiagente utilizando a plataforma Jacamo para alocação de vagas em um estacionamento inteligente**. 2015. B.S. thesis – Universidade Tecnológica Federal do Paraná.

CASTRO, L. F. S. D.; ALVES, G. V.; BORGES, A. P. Using trust degree for agents in order to assign spots in a Smart Parking. en. **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, v. 6, n. 2, p. 45–55, jun. 2017. ISSN 2255-2863. DOI: 10.14201/ADCAIJ207624555. Disponível em: <http://revistas.usal.es/index.php/2255-2863/article/view/ADCAIJ2017624555>. Acesso em: 18 jul. 2017.

CHEN, M. K.; SHELDON, M. Dynamic pricing in a labor market: Surge pricing and flexible work on the Uber platform. **UCLA Anderson**. URL: <https://www.anderson.ucla.edu>, 2015.

CHINRUNGRUENG, J.; SUNANTACHAIKUL, U.; TRIAMLUMLERD, S. Smart parking: An application of optical wireless sensor network. In: IEEE. 2007 International Symposium on Applications and the Internet Workshops. [S. l.: s. n.], 2007. P. 66–66.

DI NAPOLI, C.; DI NOCERA, D.; ROSSI, S. Negotiating parking spaces in smart cities. In: PROCEEDING of the 8th International Workshop on Agents in Traffic and Transportation, in conjunction with AAMAS. [S. l.: s. n.], 2014.

DI NAPOLI, C.; ROSSI, S. A City-aware Car Parks Marketplace for Smart Parking. In: ICAART (1). [S. l.: s. n.], 2021. P. 242–249.

DUCHEIKO, F. F.; BORGES, A. P.; ALVES, G. V. Implementação de Modelo de Raciocínio e Protocolo de Negociação para um Estacionamento Inteligente com Mecanismo de Negociação Descentralizado. Portuguese. **Revista Junior de Iniciação Científica em Ciências Exatas e Engenharia**, v. 1, n. 19, p. 25–32, 2018. ISSN 2236-0093. Disponível em: [http://www.icceeg.c3.furg.br/index.php?Itemid=837&option=bloco\\_texto&id\\_site\\_componente=1241](http://www.icceeg.c3.furg.br/index.php?Itemid=837&option=bloco_texto&id_site_componente=1241).

FERBER, J. **Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence**. [S. l.]: Addison-Wesley Longman Publishing Co., Inc., 1999.

FREE SOFTWARE FOUNDATION. **GNU Lesser General Public License**. Versão inglesa. [S. l.: s. n.], 29 jun. 2007. Disponível em: <https://www.gnu.org/licenses/lgpl-3.0.html>. Acesso em: 5 jul. 2021.

- JACAMO. **The JaCaMo approach**. 2011. Disponível em: [http://jacamo.sourceforge.net/?page\\_id=40](http://jacamo.sourceforge.net/?page_id=40). Acesso em: 5 jul. 2019.
- JASON. **Jason, a Java-based interpreter for an extended version of AgentSpeak**. 2005. Disponível em: <http://jason.sourceforge.net/wp/>. Acesso em: 5 jul. 2019.
- LI, H. *et al.* A dynamic price model based on levelized cost for district heating. **Energy, Ecology and Environment**, Springer, v. 4, n. 1, p. 15–25, 2019.
- LIAO, Y. *et al.* Intelligent dynamic service pricing strategy for multi-user vehicle-aided MEC networks. **Future Generation Computer Systems**, Elsevier, v. 114, p. 15–22, 2021.
- LIN, T.; RIVANO, H.; LE MOUËL, F. A survey of smart parking solutions. **IEEE Transactions on Intelligent Transportation Systems**, IEEE, v. 18, n. 12, p. 3229–3253, 2017.
- LITMAN, T. **Parking Management**. 2019. Disponível em: [https://www.vtspi.org/park\\_man\\_comp.pdf](https://www.vtspi.org/park_man_comp.pdf). Acesso em: 10 jul. 2019.
- MELLADO, A. *et al.* Um Módulo de Precificação Dinâmica em Sistema Multiagente de um Estacionamento Inteligente. **Revista Eletrônica de Iniciação Científica em Computação**, v. 19, n. 1, set. 2021. Disponível em: <https://sol.sbc.org.br/journals/index.php/reic/article/view/1779>.
- MELLADO, A.; ALVES, G.; BORGES, A. P. Uma comparação entre soluções de smart parkings baseados em agentes inteligentes. In: 13TH Workshop-School on Agents, Environments, and Applications (WESAAC 2019). Florianópolis: Editora UFSC, mai. 2019. P. 1–6. ISBN 2177-2096. Disponível em: <https://gsigma.ufsc.br/wesaac2019/index.php/proceedings/>.
- MELLADO, A.; ALVES, G. V. *et al.* Implantação de Precificação Dinâmica em um Estacionamento Inteligente utilizando Agentes. In: PROCEEDINGS of the 14th Workshop-School on Agents, Environments, and Applications (WESAAC 2020). [S. l.]: UTFPR, jul. 2020. v. 1, p. 152–164. ISBN 2177-2096. DOI: 10.5281/zenodo.4037413. Disponível em: <https://doi.org/10.5281/zenodo.4037413>.
- MENDES, L. R. **Utilização de data mining e deep learning para business intelligence em estrutura integrada de sistema smart parking**. 2021. Tese (Doutorado).
- MOISE. **Moise**. 2006. Disponível em: <http://moise.sourceforge.net/>. Acesso em: 5 jul. 2019.
- MONZONI MARIO. NICOLLETTI, M. A cidade para os cidadãos: mobilidade, energia e agricultura urbana. In: CADERNOS FGV Projetos – Cidades Inteligentes e Mobilidade Urbana. n.24. [S. l.: s. n.], 2014. P. 90–107.



NOBRE, L. Desafios e soluções para a mobilidade urbana. In: CADERNOS FGV Projetos – Cidades Inteligentes e Mobilidade Urbana. n.24. [S. l.: s. n.], 2014. P. 198–215.

PARKWHIZ. **BestParking**. 2021. Disponível em: <https://www.bestparking.com/>. Acesso em: 5 jun. 2021.

PIERCE, G.; SHOUP, D. Getting the prices right: an evaluation of pricing parking by demand in San Francisco. **Journal of the American Planning Association**, Taylor & Francis, v. 79, n. 1, p. 67–81, 2013.

PIERCE, G.; WILLSON, H.; SHOUP, D. Optimizing the use of public garages: Pricing parking by demand. **Transport Policy**, Elsevier, v. 44, p. 89–95, 2015.

RIZVI, S. R.; ZEHRA, S.; OLARIU, S. ASPIRE: An Agent-Oriented Smart Parking Recommendation System for Smart Cities. **IEEE Intelligent Transportation Systems Magazine**, IEEE, 2018.

RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. [S. l.]: CUMINCAD, 1995.

SAID, A. M.; KAMAL, A. E.; AFIFI, H. An intelligent parking sharing system for green and smart cities based IoT. **Computer Communications**, Elsevier, v. 172, p. 10–18, 2021.

SAKURADA, L.; BARBOSA, J.; LEITÃO, P. Deployment of Industrial Agents in a Smart Parking System. In: 2019 IEEE 17th International Conference on Industrial Informatics (INDIN). [S. l.: s. n.], 2019. v. 1, p. 1552–1557. DOI: 10.1109/INDIN41052.2019.8972190.

SFPARK. **SFpark**. 2015. Disponível em: <http://www.sfpark.org>. Acesso em: 5 jul. 2019.

SHIN, J.-H.; JUN, H.-B. A study on smart parking guidance algorithm. **Transportation Research Part C: Emerging Technologies**, Elsevier, v. 44, p. 299–317, 2014.

TIAN, Q. *et al.* Dynamic pricing for reservation-based parking system: A revenue management method. **Transport Policy**, Elsevier, v. 71, p. 36–44, 2018.

WILLIAMS, K. Dynamic airline pricing and seat availability. **Yale University - Cowles Foundation**, Cowles Foundation discussion paper, 2018.

WOOLDRIDGE, M. **An introduction to multiagent systems**. [S. l.]: John Wiley & Sons, 2009.

WOOLDRIDGE, M.; JENNINGS, N. R. Intelligent agents: Theory and practice. **The knowledge engineering review**, Cambridge University Press, v. 10, n. 2, p. 115–152, 1995.