UNIVERSIDADE FEDERAL DO MARANHÃO

Programa de Pós-Graduação em Ciência da Computação

D'jefferson Smith Santos Maranhão

Análise de um modelo para a tomada de decisões pedagógicas em ambientes voltados ao aprendizado de algoritmos.

São Luís
2021

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Análise de um modelo para a tomada de decisões pedagógicas em um ambiente voltado ao aprendizado de algoritmos

D'jefferson Smith Santos Maranhão

São Luís - MA
2021

D'jefferson Smith Santos Maranhão

Análise de um modelo para a tomada de decisões pedagógicas em um ambiente voltado ao aprendizado de algoritmos

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA, como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

Universidade Federal do Maranhão - UFMA

Centro de Ciências Exatas e Tecnologia - CCET

Programa de Pós-Graduação em Ciência da Computação - PPGCC

Orientador: Prof. Dr. Carlos de Salles Soares Neto

São Luís - MA

2021

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Diretoria Integrada de Bibliotecas/UFMA

Maranhão, D'jefferson Smith Santos.
Análise de um modelo para a tomada de decisões pedagógicas em um ambiente voltado ao aprendizado de algoritmos / D'jefferson Smith Santos Maranhão. - 2021.
120 f.

Orientador(a): Carlos de Salles Soares Neto.
Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, 2021.

1. Algoritmos de planejamento. 2. Processos de Decisão de Markov. 3. Tomada de decisões. 4. Tutoria inteligente. I. Soares Neto, Carlos de Salles. II. Título.

D'jefferson Smith Santos Maranhão

Análise de um modelo para a tomada de decisões pedagógicas em um ambiente voltado ao aprendizado de algoritmos

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA, como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

Prof. Dr. Carlos de Salles Soares Neto
Orientador

Prof. Dr. Alexandre César Muniz de Oliveira
Universidade Federal do Maranhão

Prof. Dr. Eduardo Barrére
Universidade Federal de Juiz de Fora

São Luís - MA

2021

Aos meus pais, à minha família, aos meus amigos e a todos que me acompanharam nesta jornada.

Agradecimentos

Primeiramente gostaria de agradecer a Deus, por ter me permitido alcançar mais esta vitória.

Agradeço também ao meu orientador, Prof. Carlos de Salles, por conduzir este trabalho com muita dedicação e maestria, sempre me fazendo enxergar um caminho mais simples para trilhar. Seus ensinamentos são sempre muito proveitosos e me fazem querer ser melhor. Por isso, meu respeito e minha admiração são enormes.

Aos meus pais, João Maranhão e Goretti Maranhão, pelo apoio incondicional ao longo desta jornada.

Aos meus irmãos, pelo amparo nos momentos de angústia e aflição.

A minha namorada, pelo carinho e compreensão, mesmo durante as minhas ausências.

Aos colegas do laboratório Telemídia-MA, Antônio Raposo, Christyellen e Elydillse, com os quais tive o prazer de trabalhar e colaborar, seja na elaboração de artigos, no desenvolvimento de sistemas, ou na configuração de novos serviços. Nada disso seria possível sem o auxílio de vocês.

Aos meus colegas de curso, pelo companheirismo e pela troca de conhecimentos e experiências.

A todos os professores e funcionários do Curso de Pós-Graduação em Ciência da Computação, pela paciência e apoio ao longo dos últimos anos.

A todos que, direta ou indiretamente, colaboraram para a realização deste trabalho, registro o meu muito obrigado.

"Eu estou entre aqueles que pensam que a ciência tem uma grande beleza"
(Marie Curie)

Resumo

Os professores de algoritmos são responsáveis por apresentar problemas computacionais do mundo real aos seus alunos, incentivando a resolução de problemas similares. Naturalmente, isso demanda um tempo precioso dos professores, pois requer a seleção e a ordenação dos problemas. Porém, o esforço exigido por essa tarefa de organização dos problemas acaba, muitas vezes, por torná-la impraticável. Nesse cenário, encara-se essa etapa de planejamento como um processo de tomada de decisão sequencial, em que o professor é um agente responsável por monitorar as soluções submetidas pelos alunos e apresentar novos problemas. Em específico, tem-se que os professores conseguem apenas fazer suposições quanto ao ganho cognitivo proporcionado pela resolução de um determinado problema, mas não conseguem mensurá-lo com exatidão. Em virtude disso, o presente trabalho tem o objetivo de avaliar o emprego de modelos POMDP para o sequenciamento de problemas computacionais, propiciando uma forma mais ágil, dinâmica e individualizada de ensino. Como forma de avaliar o modelo proposto, são comparadas políticas de sequenciamento geradas de forma aleatória contra políticas produzidas por algoritmos específicos, tais como QMDP, FIB, SARSOP, POMCP e POMCPOW. Os resultados mostram que os algoritmos de planejamento específicos convergem para valores muito próximos em termos de recompensa descontada média. Também indicam que uma política aleatória sempre converge para um valor inferior ao dos algoritmos específicos com relação à recompensa descontada média. No que concerne ao tempo de execução, o algoritmo FIB apresenta os melhores resultados. As principais contribuições deste trabalho são o desevolvimento de um modelo capaz de representar o processo de tomada de decisões pedagógicas que pode ser empregado em Sistemas Tutores Inteligentes (STI) voltados ao aprendizado de programação; assim como a demonstração, por meio de simulação, de que algoritmos de planejamento podem ser aplicados de forma satisfatória ao sequenciamento de problemas computacionais, superando os resultados de políticas aleatórias.

Palavras-chaves: Tutoria inteligente, Tomada de decisões, Processos de Decisão de Markov, Algoritmos de planejamento.

Abstract

Typically, algorithm teachers are responsible for presenting real-world computational problems to their students, encouraging the resolution of similar problems. This, of course, requires precious time from teachers, as it requires the selection and ordering of problems. However, the effort required by this task of organizing problems often ends up making it impractical. In this scenario, this planning stage is seen as a sequential decision-making process, in which the teacher is an agent responsible for monitoring the solutions submitted by students and presenting new problems. In particular, we have that teachers can only make assumptions about the cognitive gain provided by solving a certain problem, but they cannot measure it accurately. As a result, the present work aims to evaluate the use of POMDP models for the sequencing of computational problems, providing a more agile, dynamic and individualized form of teaching. As a way of evaluating the proposed model, randomly generated sequencing policies are compared against policies produced by specific algorithms, such as QMDP, FIB, SARSOP, POMCP and POMCPOW. The results show that the specific planning algorithms converge to very close values in terms of average discounted reward. They also indicate that a random policy always converges to a value lower than that of specific algorithms with respect to the average discounted reward. Regarding the execution time, the FIB algorithm presents the best results. The main contributions of this work are the development of a model capable of representing the pedagogical decision-making process that can be used in Intelligent Tutoring Systems (STI) aimed at learning programming; as well as the demonstration, through simulation, that planning algorithms can be applied satisfactorily to the sequencing of computational problems, surpassing the results of random policies.

Keywords: Intelligent tutoring, Decision-making, Markov Decision Processes, Planning algorithms.

Lista de ilustrações

Figura 1 – Processo de Decisão Sequencial 24
Figura 2 – Critérios de otimização: a) horizonte finito, b) descontado, c) recompensa média. 26
Figura 3 – Estrutura em árvore de um modelo POMDP. 33
Figura 4 – Representação do funcionamento de um modelo POMDP. 34
Figura 5 – Aplicação de uma política em uma árvore de crenças. 37
Figura 6 – Função de valor ótima para um modelo POMDP de dois estados. . . . 38
Figura 7 – Metodologia do Trabalho 41
Figura 8 – Atividades da Etapa de Levantamento Bibliográfico. 42
Figura 9 – Atividades da Etapa de Detalhamento da Abordagem. 44
Figura 10 – Atividades da Etapa de Simulação. 45
Figura 11 – Atividades da Etapa de Análise e Discussão de Resultados. 46
Figura 12 – *String* de busca. 48
Figura 13 – Funcionamento do Modelo Pedagógico. 58
Figura 14 – Estados de conhecimento e suas transições (para apenas dois problemas). 59
Figura 15 – Distribuição de probabilidade da ação de recomendar o problema P_1. . 60
Figura 16 – Distribuição de probabilidade das observações para a ação de recomendar o problema P_1. 60
Figura 17 – Árvore de Crenças $T_{\mathcal{R}}$. 67
Figura 18 – Árvore de Crenças do POMCPOW 69
Figura 19 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo POMCP. 73
Figura 20 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo POMCPOW. 74
Figura 21 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo QMDP. 74
Figura 22 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo QMDP. 74
Figura 23 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo FIB. 75
Figura 24 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Política Aleatória. 75
Figura 25 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 1. 76
Figura 26 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 2. 76

Figura 27 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 3. 77
Figura 28 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 4. 77
Figura 29 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 5. 77
Figura 30 – Gráfico da distribuição t de Student para 9 Graus de Liberdade. 79
Figura A1 – Definição da distribuição de probabilidade do estado inicial. 91
Figura A2 – Definição do espaço de estados. 92
Figura A3 – Definição do espaço de ações. 92
Figura A4 – Definição do espaço de observações. 92
Figura A5 – Definição da função de transição. 93
Figura A6 – Definição da função de observação. 93
Figura A7 – Definição da função de recompensa. 94
Figura A8 – Definição do fator de desconto. 94
Figura A9 – Instanciação do modelo. 94

Lista de tabelas

Tabela 1 – Critérios de seleção. 49
Tabela 2 – Formulário para extração de dados. 50
Tabela 3 – Quadro comparativo. 52
Tabela 4 – Resumo das Instâncias. 64
Tabela 5 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo QMDP. 70
Tabela 6 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo FIB. 70
Tabela 7 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo SARSOP. 71
Tabela 8 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo POMCP. 71
Tabela 9 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo POMCPOW. 72
Tabela 10 – Recompensa Descontada Média e Tempo Médio de Execução da Política Aleatória. 72
Tabela 11 – Testes de Hipóteses para a Instância 1. 80
Tabela 12 – Testes de Hipóteses para a Instância 2. 80
Tabela 13 – Testes de Hipóteses para a Instância 3. 81
Tabela 14 – Testes de Hipóteses para a Instância 4. 81
Tabela 15 – Testes de Hipóteses para a Instância 5. 82
Tabela 16 – Resumo dos Testes Hipóteses. 82
Tabela B1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 1. 95
Tabela B2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 1. 95
Tabela B3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 1, por Horizonte e Amostra, para a instância 1. 96
Tabela B4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 1. 96
Tabela B5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 1. 96
Tabela B6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 1. 97
Tabela B7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 1. 97

Tabela B8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 1. 97
Tabela B9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 1. 98
Tabela B10–Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 1. 98
Tabela B11–Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 1. 98
Tabela B12–Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 1. 99
Tabela C1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 2. 100
Tabela C2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 2. 100
Tabela C3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 2. 101
Tabela C4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 2. 101
Tabela C5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 2. 101
Tabela C6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 2. 102
Tabela C7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 2. 102
Tabela C8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 2. 102
Tabela C9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 2. 103
Tabela C10–Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 2. 103
Tabela C11–Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 2. 103
Tabela C12–Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 2. 104
Tabela D1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 3. 105
Tabela D2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 3. 105

Tabela D3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 3. 106
Tabela D4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 3. 106
Tabela D5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 3. 106
Tabela D6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 3. 107
Tabela D7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 3. 107
Tabela D8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 3. 107
Tabela D9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 3. 108
Tabela D10–Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 3. 108
Tabela D11–Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 3. 108
Tabela D12–Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 3. 109
Tabela E1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 4. 110
Tabela E2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 4. 110
Tabela E3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 4. 111
Tabela E4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 4. 111
Tabela E5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 4. 111
Tabela E6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 4. 112
Tabela E7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 4. 112
Tabela E8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 4. 112
Tabela E9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 4. 113

Tabela E10–Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 4. 113
Tabela E11–Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 4. 113
Tabela E12–Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 4. 114
Tabela F1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 5. 115
Tabela F2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 5. 115
Tabela F3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 5. 116
Tabela F4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 5. 116
Tabela F5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 5. 116
Tabela F6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 5. 117
Tabela F7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 5. 117
Tabela F8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 5. 117
Tabela F9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 5. 118
Tabela F10–Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 5. 118
Tabela F11–Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 5. 118
Tabela F12–Tempo de Execução da Política Aleatória por Horizonte e Amostra. . . 119
Tabela G1 – Teste de Normalidade de Amostra do Algoritmo QMDP. 120
Tabela G2 – Teste de Normalidade de Amostra do Algoritmo POMCP. 120

Lista de abreviaturas e siglas

AVA - Ambiente Virtual de Aprendizagem

FIB - Fast Informed Bound

MCTS - Monte-Carlo Tree Search

MDP - Processo de Decisão de Markov

POMCP - Partially Observable Monte Carlo Planning

POMCPOW - Partially Observable Monte Carlo Planning with Observation Widening

POMDP - Processo de Decisão de Markov Parcialmente Observável

QMDP - Q-Value Markov Decision Process

RSL - Revisão Sistemática da Literatura

SARSOP - Successive Approximations of the Reachable Space under Optimal Policies

STI - Sistema Tutor Inteligente

UCT - Upper Confidence bounds applied to Trees

UFMA - Universidade Federal do Maranhão

Sumário

1 INTRODUÇÃO . . . 19
1.1 Objetivos . . . 20
1.2 Objetivos específicos . . . 20
1.3 Contribuições . . . 21
1.4 Organização do Trabalho . . . 21

2 FUNDAMENTAÇÃO TEÓRICA . . . 23
2.1 Processos de Decisão Sequencial . . . 23
2.2 Processos de Decisão de Markov . . . 24
2.2.1 Definição formal . . . 24
2.2.2 Políticas . . . 25
2.2.3 Critérios de desempenho . . . 25
2.2.4 Funções de valor . . . 26
2.2.5 Abordagens para a resolução de MDPs . . . 28
2.2.6 Métodos Baseados em Modelo - Programação Dinâmica . . . 28
2.2.6.1 Iteração de política . . . 28
2.2.6.2 Iteração de valor . . . 31
2.2.6.3 Outros métodos . . . 32
2.3 Processo de Decisão de Markov Parcialmente Observável . . . 33
2.3.1 Definição formal . . . 34
2.3.2 Critérios de Desempenho . . . 35
2.3.3 Estados de Crença . . . 35
2.3.4 Política . . . 36
2.3.5 Função de valor . . . 37
2.3.6 Algoritmos . . . 39
2.4 Considerações Finais . . . 39

3 METODOLOGIA . . . 41
3.1 Etapa de Levantamento bibliográfico . . . 41
3.2 Etapa de Modelagem do Problema . . . 43
3.3 Etapa de Simulação . . . 45
3.4 Etapa de Análise e Discussão de Resultados . . . 46

4 LEVANTAMENTO BIBLIOGRÁFICO . . . 47
4.1 Planejamento . . . 47
4.1.1 Definição das questões de pesquisa . . . 47

4.1.2 Definição da *string* de busca . . . 48
4.1.3 Definição das fontes de pesquisa . . . 48
4.1.4 Definição dos critérios de seleção . . . 49
4.1.5 Definição da estratégia para extração de dados . . . 50
4.2 Execução . . . 50
4.3 Sumarização dos resultados . . . 51
4.3.1 Quais as linguagens de programação ensinadas? . . . 51
4.3.2 Quais as estratégias de ensino adotadas? . . . 53
4.3.3 Quais as técnicas de inteligência artificial empregadas? . . . 54
4.3.4 Para quais finalidades as técnicas são aplicadas? . . . 55
4.4 Considerações Finais . . . 56

5 MODELAGEM DO PROBLEMA . . . 57
5.1 Definição do Cenário de Ensino e Aprendizagem . . . 57
5.2 Definição da Estrutura Interna do Modelo Pedagógico . . . 57
5.2.1 Espaço de Estados . . . 58
5.2.2 Espaço de Ações . . . 59
5.2.3 Espaço de Observações . . . 60
5.2.4 Função de transição . . . 61
5.2.5 Função de Observação . . . 61
5.2.6 Função de Recompensa . . . 61
5.2.7 Fator de desconto . . . 62
5.2.8 Estado de Crença Inicial . . . 62
5.3 Criação de Instâncias do Modelo Pedagógico . . . 62

6 SIMULAÇÃO . . . 65
6.1 Detalhamento dos Algoritmos de Planejamento . . . 65
6.1.1 *Q-Value Markov Decision Process* (QMDP) . . . 65
6.1.2 *Fast Informed Bound* (FIB) . . . 66
6.1.3 *Successive Approximations of the Reachable Space under Optimal Policies* (SARSOP) . . . 66
6.1.4 *Partially Observable Monte-Carlo Planning* (POMCP) . . . 68
6.1.5 *Partially Observable Monte Carlo Planning with Observation Widening* (POMCPOW) . . . 69
6.2 Execução das Instâncias . . . 69

7 ANÁLISE E DISCUSSÃO DOS RESULTADOS . . . 73
7.1 Comparação dos Resultados por Instância . . . 73
7.2 Comparação dos Resultados por Algoritmo . . . 75
7.3 Análise Estatística dos Resultados . . . 78

8 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS 83

REFERÊNCIAS 85

APÊNDICES 90

APÊNDICE A – CRIAÇÃO DE INSTÂNCIAS NA LINGUAGEM JULIA 91

APÊNDICE B – DADOS DA INSTÂNCIA 1 95

APÊNDICE C – DADOS DA INSTÂNCIA 2 100

APÊNDICE D – DADOS DA INSTÂNCIA 3 105

APÊNDICE E – DADOS DA INSTÂNCIA 4 110

APÊNDICE F – DADOS DA INSTÂNCIA 5 115

APÊNDICE G – TESTE DE NORMALIDADE PARA AMOSTRAS 120

1 Introdução

O aprendizado de algoritmos é um tema multifacetado e complexo. Em geral, os alunos demonstram pouca habilidade em interpretar a descrição de um problema e decompô-lo em instâncias menores (MCCRACKEN et al., 2001). Também apresentam dificuldades em rastrear, ler e entender trechos de código e não são capazes de compreender princípios e rotinas básicas de programação (LISTER et al., 2004).

Adicionalmente, o aprendizado de algoritmos possui etapas bem definidas, que são pré-requisitos umas das outras. De forma ilustrativa, primeiro aprende-se o conceito de declaração e inicialização de variáveis. Depois, passa-se ao estudo de operadores aritméticos, de comparação e lógicos. Por fim, compreende-se o conceito de estruturas de decisão, o qual é necessário para a compreensão de laços de repetição.

Uma prática comum no aprendizado de algoritmos é a resolução de problemas computacionais do mundo real. Para isso, o professor precisa selecionar um conjunto de atividades e organizar de acordo com o conceito abordado e o nível de dificuldade das questões. Há uma ampla diversidade de ferramentas que apoiam o professor nesse processo de escolha das atividades, tais como o URI Online (Bez; Tonin; Rodegheri, 2014) e o UVA Online Judge (REVILLA; MANZOOR; LIU, 2008).

Embora sejam muito úteis enquanto repositórios de problemas, essas ferramentas não oferecem um suporte inteligente à navegação. Assim, cabe ao professor a definição da ordem apropriada para a resolução das atividades, para que os alunos não se sintam desmotivados. Todavia, o esforço exigido por essa tarefa de organização pode se mostrar impraticável, pois cada aluno aprende em um ritmo particular e possui dificuldades específicas.

Ademais, o trabalho de organização das atividades de forma personalizada pode ser encarado como um processo de tomada de decisão sequencial, em que o professor é um agente/tutor inteligente responsável por monitorar as soluções submetidas pelos alunos e por propor novas ações de ensino de forma sequencial, levando em consideração as respostas observadas anteriormente.

Nesse ambiente, o tutor inteligente não tem capacidade de aferir com exatidão o estado de conhecimento do aluno, mas é capaz de fazer uma série de suposições com base nas respostas submetidas pelos alunos. Esse processo de ensino, no qual a tomada de decisões ocorre em meio a incertezas quanto ao estado de conhecimento do aluno, pode ser modelado por meio de Processos de Decisão de Markov Parcialmente Observáveis (WOOLF, 2010) (WANG, 2018).

Os estudos acerca da aplicação dessa técnica à tutoria inteligente começaram ainda na década de 90. Nos primeiros anos, ela foi usada para modelar estados mentais e para encontrar as melhores maneiras de se ensinar certos conceitos. Os trabalhos mais atuais, todavia, caracterizam-se pela aplicação da técnica para otimizar e personalizar o ensino, mas variam conforme as definições de estados, ações, observações e com relação às estratégias de resolução do modelo (RAFFERTY et al., 2011)(THEOCHAROUS et al., 2009).

Ante todo o exposto, mostra-se de enorme relevância a análise da tomada de decisões pedagógicas por Sistemas Tutores Inteligentes (STI), em especial, no que diz respeito ao sequenciamento de problemas computacionais em ambientes voltados ao aprendizado de algoritmos. Nesse sentido, a presente pesquisa visa responder ao seguinte questionamento: o emprego de estratégias basedas em Processos de Decisão de Markov Parcialmente Observáveis (POMDP, do inglês Partially Observable Markov Decision Processes) é capaz de promover uma melhoria na recomendação de problemas computacionais se comparado a uma abordagem que recomenda questões de forma aleatória?

1.1 Objetivos

Este trabalho tem por objetivo geral avaliar o emprego de um modelo POMDP para a tomada de decisões quanto ao sequenciamento de problemas computacionais em ambientes voltados ao ensino de algoritmos, comparando políticas de sequenciamento geradas de forma aleatória contra políticas produzidas por algoritmos específicos.

1.2 Objetivos específicos

Mais especificamente, o presente trabalho busca alcançar os seguintes objetivos relacionados ao emprego de estratégias de ensino baseadas em modelos POMDP para o sequenciamento de problemas computacionais:

- apresentar os principais aspectos presentes na literatura a respeito de Processos de Decisão de Markov Parcialmente Observáveis (POMDP);
- modelar o processo de tomada de decisões pedagógicas em ambientes de tutoria inteligente voltados ao aprendizado de programação, criando um conjunto de instâncias;
- selecionar um conjunto de algoritmos de planejamento e utilizá-los para resolver as instâncias do modelo POMDP.

1.3 Contribuições

Dentre as principais contribuições deste trabalho, destacam-se:

- a elaboração de uma Revisão Sistemática da Literatura (RSL) com o objetivo de descobrir quais técnicas de inteligência artificial (IA) podem ser empregadas na personalização de STIs voltados ao ensino de programação;
- o desevolvimento de um modelo capaz de representar o processo de tomada de decisões pedagógicas que pode ser empregado em STIs voltados ao aprendizado de programação.
- a demonstração, por meio de simulação, de que algoritmos de planejamento podem ser aplicados de forma satisfatória ao sequenciamento de problemas computacionais, superando os resultados de políticas aleatórias.

1.4 Organização do Trabalho

O trabalho está organizado da seguinte forma:

- O Capítulo 2 apresenta os conceitos de Processos de Decisão de Markov parcialmente observáveis utilizados na elaboração da proposta deste trabalho.
- O Capítulo 3 apresenta a metodologia utilizada para atingir o objetivo proposto por este trabalho.
- O Capítulo 4 descreve uma Revisão Sistemática da Literatura que foi feito para atingir a etapa de levantamento bibliográfico, além de descrever os critérios para seleção de técnicas de modelagem de conhecimento, datasets e análise para este trabalho.
- O Capítulo 5 apresenta uma visão do cenário pedagógico e a definição da estrutura interna do modelo responsável por selecionar as estratégias pedagógicas mais apropriadas para orientar o processo de ensino. Ela também é responsável por construir um conjunto de instâncias do modelo pedagógico para diferentes números de problemas computacionais.
- O Capítulo 6 detalha os algoritmos de planejamento escolhidos para resolução das instâncias do modelo pedagógico (QMDP, FIB, SARSOP, POMCP e POMCPOW). Também são apresentados os resultados da simulação utilizando esses algoritmos e uma política aleatória.

- O Capítulo 7 analisa e representa graficamente dos dados obtidos na etapa de simulação, comparando os resultados de algoritmos de planejamento específicos contra os resultados de uma política aleatória. Também é avaliada a hipótese de que uma política gerada por um algoritmo específico é capaz de superar o resultado alcançado por uma política aleatória.

- o Capítulo 8 apresenta as conclusões do trabalho e o direcionamento para trabalhos futuros.

2 Fundamentação Teórica

As seções a seguir apresentam conceitos-chaves para este trabalho, como Processos de Decisão Sequencial; Processos de Decisão de Markov; e Processos de Decisão de Markov Parcialmente Observáveis, que serão empregados na criação de um modelo para a tomada de decisões pedagógicas em ambientes voltados ao aprendizado de algoritmos.

2.1 Processos de Decisão Sequencial

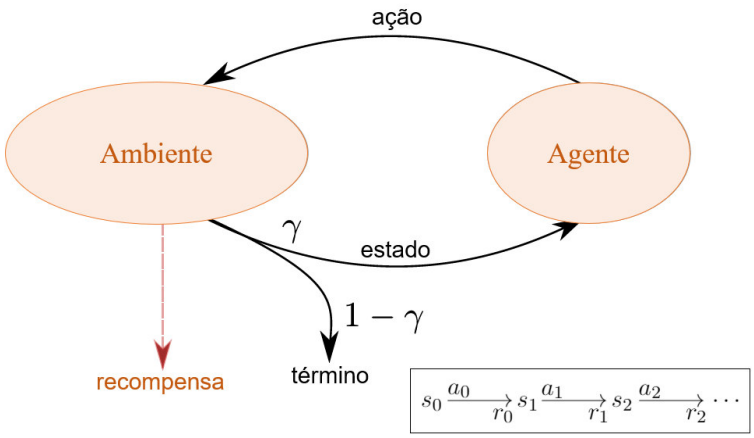
Um Processo de Decisão Sequencial é um modelo que representa a interação síncrona de um agente com um ambiente externo. O objetivo deste agente é maximizar sua recompensa por meio da escolha de ações apropriadas. Essas ações e o histórico de estados anteriores determinam a distribuição de probabilidade sobre os próximos estados possíveis. Assim sendo, a sequência de estados do sistema pode ser modelada como um processo estocástico (BRAZIUNAS, 2003), o que significa que o efeito das ações não é certo nem determinado.

Para Russel, Norvig et al. (2016), os problemas de busca e planejamento são casos especiais dos problemas de decisão sequencial que lidam com questões que geralmente envolvem os seguintes aspectos: utilidades, incerteza e percepção. Um problema de decisão sequencial normalmente exige a repetição das seguintes etapas: 1. observação do estado do ambiente; 2. escolha e realização de uma ação; 3. cômputo da recompensa imediata; e 4. transição para um novo estado.

Em decorrência das ações serem essencialmente estocásticas, o ambiente nem sempre responde conforme o esperado. Por isso, o agente deve buscar atingir uma meta (estado) considerando a imprevisibilidade de suas ações. Esse tipo de problema pode ser modelado por meio de Processos de Decisão de Markov (MDP), em que um agente toma suas decisões sequenciais com base em uma percepção do ambiente. Outra forma de abordar esse problema é mediante o emprego de Processos de Decisão de Markov Parcialmente Observáveis (POMDP), que são capazes de lidar com decisões sequenciais em ambientes de incerteza.

A Figura 1 mostra como é realizada a interação de um agente com um ambiente em um processo de decisão sequencial. Em cada espaço de tempo, o ambiente encontra-se em um estado s_k. O agente então escolhe uma ação a_k e o ambiente evolui para um novo estado s_{k+1}. Nesse instante, o agente também obtém uma recompensa r_k pela execução da ação a_k no estado s_k, que representa um estímulo quantitativo decorrente da decisão tomada.

Figura 1 – Processo de Decisão Sequencial



Fonte – Junior, Freitas et al. (2018).

2.2 Processos de Decisão de Markov

Os Processos de Decisão de Markov (MDP, do inglês Markov Decision Processes) são modelos comumente utilizados para representar problemas de decisão sequencial. Eles consistem em processos estocásticos discretos no tempo, em que um agente atua em um ambiente que lhe é visível, buscando alcançar recompensas cada vez mais promissoras.

2.2.1 Definição formal

O modelo é composto por estados, ações, transições entre estados e funções de recompensa, sendo formalmente caracterizado pela tupla $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$, como definido a seguir. A representação adotada neste trabalho está fundamentada nas pesquisas de (OTTERLO; WIERING, 2012), (COMANICI, 2016), (LITTMAN, 1996) e (KOLOBOV, 2013).

O espaço de estados $\mathcal{S}$ é definido como um conjunto finito $\{s^1, ..., s^N\}$, em que o tamanho do espaço é N, isto é, $|\mathcal{S}| = N$. Ele consiste em todas as possíveis descrições do estado do ambiente. Um elemento de $\mathcal{S}$ deve conter toda a informação necessária para que o agente decida acerca dos possíveis caminhos pelos quais o ambiente passará.

O espaço de ações $\mathcal{A}$ é definido como um conjunto finito $\{a^1, ..., a^K\}$ em que o tamanho do espaço é K, isto é, $|\mathcal{A}| = K$. Ele consiste em todas as possibilidades de escolha que um agente pode fazer. Como o sistema é probabilístico, o agente não é capaz de escolher com exatidão as sequências de transições de estados pelas quais o ambiente passará, mas suas escolhas podem influenciar a distribuição de probabilidade dos possíveis caminhos.

A função de transição $\mathcal{T}$ é definida como $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$, isto é, a probabilidade

de se terminar no estado s' após realizar a ação a no estado s é representada $\mathcal{T}(s, a, s')$. Ou seja, aplicando a ação $a \in \mathcal{A}$ no estado $s \in \mathcal{S}$, o sistema faz a transição de um estado s para um novo estado $s' \in \mathcal{S}$, com base na distribuição de probabilidades do conjunto de possíveis transições.

Para todas as ações a e todos os estados s e s', é necessário que $\mathcal{T}(s, a, s') \geq 0$ e $\mathcal{T}(s, a, s') \leq 1$. Além disso, para todos os estados s e para todas as ações a, $\sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') = 1$, ou seja, $\mathcal{T}$ define uma distribuição de probabilidade adequada dos próximos estados possíveis. Ainda, pode-se estabelecer que $\mathcal{T}(s, a, s') = 0$ para todos os estados $s' \in \mathcal{S}$ em que a não for aplicável a s.

Por fim, a coleção de funções de recompensa ($\mathcal{R}$) é responsável por especificar as recompensas relacionadas ao fato de se estar em um determinado estado, ou de se realizar uma certa ação em um determinado estado. Associada a cada ação $a \in \mathcal{A}$, há uma função de recompensa imediata $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, que provê a recompensa esperada $\mathcal{R}(s, a)$ por executar a ação $a \in \mathcal{A}$ no estado $s \in \mathcal{S}$.

2.2.2 Políticas

Dado um Processo de Decisão de Markov $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$, uma política é uma função computável que gera uma ação $a \in \mathcal{A}$ como saída para cada estado $s \in \mathcal{S}$. Formalmente, uma política determinística π é uma função definida como $\pi : \mathcal{S} \to \mathcal{A}$. Também é possível definir uma política estocástica como $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ tal que, para cada estado $s \in \mathcal{S}$, assegure-se que $\pi(s, a) \geq 0$ e $\sum_{a \in \mathcal{A}} \pi(s, a) = 1$.

Em um MDP, a aplicação de uma política ocorre da seguinte forma: primeiro, gera-se uma distribuição inicial para o estado s_0. Em seguida, a política π sugere a ação $a_0 = \pi(s_0)$ e essa ação é executada. Com base na função de transição $\mathcal{T}$ e na função de recompensa $\mathcal{R}$, é feita uma transição para o estado s_1, com probabilidade $\mathcal{T}(s_0, a, s_1)$ e uma recompensa $r_0 = \mathcal{R}(s_0, a_0, s_1)$ é recebida. O processo continua até o agente alcançar um determinado objetivo, produzindo $s_0, a_0, r_0, s_1, a_1, r_1, s_2, a_2, \cdots, s_n, a_n, r_n$.

2.2.3 Critérios de desempenho

O objetivo do aprendizado em um MDP é coletar recompensas. Se o agente estivesse preocupado apenas com a recompensa imediata, o critério de desempenho mais simples seria a maximização de $\mathbb{E}[r_t]$. No entanto, existem outros critérios que buscam antecipar um comportamento futuro. Os três critérios de desempenho principais são: modelo de horizonte finito; modelo de horizonte infinito e modelo de recompensa média.

No modelo de horizonte finito, o agente deve otimizar a recompensa esperada em uma sequência de etapas de comprimento fixo, h. Pode-se encarar isso de duas maneiras.

Figura 2 – Critérios de otimização: a) horizonte finito, b) descontado, c) recompensa média.

$$\mathbb{E}\left[\sum_{t=0}^{h} r_t\right] \qquad \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t \times r_t\right] \qquad \lim_{h\to\infty} \mathbb{E}\left[\frac{1}{h}\sum_{t=0}^{h} r_t\right]$$

Fonte: Otterlo e Wiering (2012)

O agente poderia, em um primeiro momento, executar a ação ótima da etapa h; depois, a ação ótima da etapa $(h-1)$ e assim por diante. Outra maneira é o agente sempre executar a ação ótima da etapa h. O problema é que o valor ideal para o comprimento do horizonte h nem sempre pode ser estabelecido.

No modelo de horizonte infinito, a recompensa de longo prazo é levada em consideração, mas as recompensas recebidas no futuro são descontadas de acordo com o espaço de tempo em que são recebidas. Um fator de desconto γ, com $0 \leq \gamma < 1$, é usado para isso. Nesse caso, as recompensas obtidas no futuro são mais descontadas que as recompensas recebidas no presente. O fator de desconto garante que, mesmo com horizonte infinito, a soma das recompensas obtidas seja finita.

No critério da recompensa média, a recompensa média de longo prazo é maximizada. Um problema com esse critério é que não se consegue distinguir entre duas políticas em que uma recebe muita recompensa nas fases iniciais e outra que não. A diferença inicial acaba sendo ocultada na média de longo prazo.

2.2.4 Funções de valor

Uma política permite que se avalie a recompensa esperada a longo prazo decorrente de sua execução. No caso do horizonte finito, seja $\delta = \langle \pi_k, ..., \pi_1 \rangle$ uma política estacionária de k-etapas e seja $V_t^{\delta_t}(s)$ a recompensa futura esperada ao se iniciar no estado s e se executar a política não-estacionária δ por t períodos de tempo. O valor da etapa final é a recompensa imediata $V_1^{\delta_t}(s) = \sum_{s' \in \mathcal{S}} \mathcal{T}(s, \pi_1(s), s') \times \mathcal{R}(s, \pi_1(s), s')$. Para $t > 1$, pode-se definir $V_t^{\delta_t}(s)$ como:

$$V_t^{\delta_t}(s) = \sum_{a \in \mathcal{A}} \pi(s,a) \times \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \left(\mathcal{R}(s,a,s') + \gamma \times V_{t-1}^{\delta_t}(s')\right) \tag{2.1}$$

O valor da etapa t de estar no estado s e executar a política não-estacionária δ é a recompensa imediata, $\sum_{s' \in \mathcal{S}} \mathcal{T}(s, \pi_t(s), s') \times \mathcal{R}(s, \pi_t(s), s')$, somada ao valor descontado esperado das $(t-1)$ etapas restantes. Para avaliar as etapas restantes, deve-se considerar todos os possíveis estados resultantes s', a probabilidade de sua ocorrência $\mathcal{T}(s, \pi_t(s), s')$, e o valor da etapa $(t-1)$ na política δ, $V_{t-1}^{\delta_t}(s')$.

Seja $V^{\pi}(s)$ a recompensa futura descontada esperada ao se iniciar no estado s e executar a política estacionária π indefinidamente. A função de valor para a política π é a única solução para um conjunto de equações lineares simultâneas, uma para cada estado s, definidas recursivamente por:

$$V^{\pi}(s) = \sum_{a \in \mathcal{A}} \pi(s,a) \times \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \left(\mathcal{R}(s,a,s') + \gamma \times V_{t-1}^{\delta_t}(s') \right) \tag{2.2}$$

Sabendo-se calcular a função de valor para uma determinada política, pode-se também definir uma política com base em uma função de valor. Dada uma função de valor V, uma política gulosa, π_V, pode ser obtida executando-se a ação em cada estado com o melhor valor de cada etapa, sendo definida da seguinte forma:

$$\pi_V(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \left[\sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \left(\mathcal{R}(s,a,s') + \gamma(s') \right) \right] \tag{2.3}$$

Pode-se, ainda, encontrar a política ótima de horizonte-finito para um determinado MDP, $\delta^* = \langle \pi_k^*, ..., \pi_1^* \rangle$. Para $1 \leq t \leq k$, pode-se definir π_t^* como mostrado a seguir. Na etapa final, o agente deveria maximizar sua recompensa imediata:

$$\pi_1^*(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \left[\sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \mathcal{R}(s,a,s') \right] \tag{2.4}$$

Pode-se definir π_t^* em termos da função de valor para a política ótima de $(t-1)$ etapas, $V_{t-1}^{\delta_t}$:

$$\pi_t^*(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \left[\sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \left(\mathcal{R}(s,a,s') + \gamma \times V_{t-1}^{\delta_t}(s') \right) \right] \tag{2.5}$$

No caso de horizonte infinito descontado, dado um estado inicial s, procura-se executar uma política π que maximize $V^{\pi}(s)$. Pode-se mostrar que existe uma política estacionária π^* ótima para cada estado inicial. A função de valor para essa política, escrita V^*, é definida pelo conjunto de equações:

$$V^*(s) = \max_{a \in \mathcal{A}} \left[\sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \left(\mathcal{R}(s,a,s') + \gamma \times V^*(s') \right) \right] \tag{2.6}$$

A presença do operador *max* significa que o sistema é não-linear, isto é, o método de eliminação Gaussiana não é suficiente para resolvê-lo. Na próxima seção, serão explorados

algoritmos capazes de resolver processos de decisão de Markov, encontrando uma política ótima e uma função de valor, dada sua descrição em termos de $\mathcal{T}$, $\mathcal{R}$ e γ.

2.2.5 Abordagens para a resolução de MDPs

Agora que foram definidos políticas, critérios de otimização e funções de valor, é hora de considerar a questão de como calcular políticas ótimas. Resolver um determinado MDP consiste em calcular uma política ótima π^*. Os algoritmos desenvolvidos para esse fim classificam-se em: baseados em modelo e livres de modelo.

Os algoritmos baseados em modelo pressupõem que um modelo do MDP é conhecido antecipadamente e pode ser usado para calcular funções e políticas de valor usando a equação de Bellman. A maioria dos métodos tem como objetivo calcular funções de valor de estado que, na presença do modelo, podem ser usadas para a seleção da ação ótima. De modo geral, os algoritmos dessa categoria se baseiam na técnica de programação dinâmica.

Os algoritmos livres de modelo não dependem da disponibilidade de um modelo perfeito. Em vez disso, eles dependem da interação com o ambiente, ou seja, uma simulação da política, gerando amostras de transições e recompensas de estados. Essas amostras são usadas para estimar funções de valor de ação de estado. Como um modelo do MDP não é conhecido, o agente precisa explorar o MDP para obter informações. De modo geral, os algoritmos dessa categoria se baseiam na técnica de aprendizagem por reforço.

2.2.6 Métodos Baseados em Modelo - Programação Dinâmica

Os dois principais métodos de programação dinâmica são a iteração de políticas e a iteração de valores. No primeiro, há uma clara separação entre as etapas de avaliação e de aprimoramento da política. No último, há uma forte integração dessas etapas. Nas subseções seguintes serão apresentados os detalhes desses algoritmos.

2.2.6.1 Iteração de política

É um algoritmo de programação dinâmica usado para calcular uma política ótima que se baseia em duas etapas fundamentais: avaliação e aprimoramento. O algoritmo começa avaliando uma determinada política e, em seguida, usa a função de valor dessa política para encontrar outras melhores. Isso é feito considerando-se uma variação da política atual com relação aos demais estados, para saber se devemos ou não alterar a política para escolher deterministicamente uma ação diferente daquela, de acordo com o estado s.

A etapa de avaliação consiste em encontrar uma função de valor V^π de uma determinada política π. Na seção anterior, foi estabelecido que, para todo estado $s \in S$, a

seguinte relação é válida:

$$V^{\pi}(s) = \sum_{a \in \mathcal{A}} \pi(s,a) \times \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \Big(\mathcal{R}(s,a,s') + \gamma \times V^{\pi}(s') \Big) \tag{2.7}$$

Esse sistema apresenta o mesmo número de equações e incógnitas, isto é, $|\mathcal{S}|$. Por conta disso, pode ser resolvido por meio de algoritmos de programação linear. No entanto, um procedimento iterativo também é possível. Nesse caso, a equação de Bellman é transformada em uma regra que atualiza a função de valor atual V_k^{π} em V_{k+1}^{π}, ampliando o horizonte de planejamento em uma etapa:

$$\begin{aligned} V_{k+1}^{\pi}(s) &= \mathbb{E}_{\pi} \Big\{ r_t + \gamma \times V^{\pi}(s_{t+1}) \mid s_t = s \Big\} \\ &= \sum_{a \in \mathcal{A}} \pi(s,a) \times \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \Big(\mathcal{R}(s,a,s') + \gamma \times V_k^{\pi}(s') \Big) \end{aligned} \tag{2.8}$$

A sequência de aproximações de V_{π}^{k} com k tendendo ao infinito tem comportamento convergente. Para isso, é necessário que a regra de atualização seja aplicada a todos os estados $s \in \mathcal{S}$ a cada iteração. Isso faz com que o antigo valor daquele estado seja substituído por um novo, que é baseado no valor esperado dos possíveis estados sucessores e ponderado pelas probabilidades de transição.

Conhecendo-se a função de valor V_{π} de uma política π como resultado da etapa de avaliação, pode-se tentar melhorar essa política. Primeiro, identifica-se o valor de todas as ações em cada estado. Em seguida, selecionam-se as ações que produzem o maior retorno. Esse processo de criação de uma nova política, que aprimora uma anterior, consiste na etapa de aprimoramento. Isso é feito usando:

$$\begin{aligned} Q^{\pi}(s,a) &= \mathbb{E}_{\pi} \Big\{ r_t + \gamma \times V^{\pi}(s_{t+1}) \mid s_t = s,\ a_t = a \Big\} \\ &= \sum_{a \in \mathcal{A}} \pi(s,a) \times \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \Big(\mathcal{R}(s,a,s') + \gamma \times V^{\pi}(s') \Big) \end{aligned} \tag{2.9}$$

Se $Q^{\pi}(s,a)$ for maior que $V^{\pi}(s)$ para algum $a \in \mathcal{A}$, pode-se tentar melhorar a política atual selecionando uma ação diferente, possivelmente melhor, em um estado específico. De fato, pode-se avaliar todas as ações em todos os estados, escolhendo a melhor ação dentre todos os estados. Ou seja, pode-se calcular uma política gulosa π' selecionando

a melhor ação em cada estado, com base na função de valor atual V^{π}:

$$\begin{aligned}\pi'(s) &= \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, Q^{\pi}(s,a) \\ &= \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, \mathbb{E}_{\pi}\Big\{ r_t + \gamma \times V^{\pi}(s_{t+1}) \mid s_t = s,\ a_t = a \Big\} \\ &= \underset{a \in \mathcal{A}}{\operatorname{argmax}} \left[\sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \Big(\mathcal{R}(s,a,s') + \gamma \times V_{t-1}^{\delta_t}(s') \Big) \right]\end{aligned} \tag{2.10}$$

Por outro lado, se a política não puder ser aprimorada dessa maneira, significa que ela já é ótima e sua função de valor satisfaz a equação de Bellman para a função de valor ótimo. De maneira semelhante, também é possível executar essas etapas para políticas estocásticas misturando as probabilidades de ação no operador de expectância.

De forma resumida, a iteração de políticas começa com uma política inicial π_0 e realiza uma série de iterações nas quais a política atual é avaliada e, em seguida, aprimorada. A etapa de avaliação da política calcula o V_{π}^{k} de maneira iterativa. A etapa de aprimoramento calcula π_{k+1} usando V_{π}^{k}. Para cada estado, a ação ótima é determinada. Se para todos os estados s, $\pi_{k+1}(s) = \pi_k(s)$, a política é estável e o algoritmo de iteração da política pode parar. A iteração de política gera uma sequência de políticas e funções de valor alternadas:

$$\pi_0 \to V^{\pi_0} \to \pi_1 \to V^{\pi_1} \to \pi_2 \to V^{\pi_2} \to \pi_3 \to V^{\pi_3} \to ... \to \pi^*$$

Para MDPs finitos, isto é, espaços de estado e ação finitos, o algoritmo de iteração de política converge após um número finito de iterações. Cada política π_{k+1} é estritamente melhor do que a anterior π_k, a menos que $\pi_k = \pi^*$, caso em que o algoritmo termina. Embora seja capaz de calcular a política ideal para um determinado MDP em tempo finito, o algoritmo é relativamente ineficiente.

Em particular, a etapa de avaliação de políticas é computacionalmente cara. São calculadas funções de valor para cada política intermediária $\pi_0, \cdots, \pi_k, \cdots, \pi^*$, o que

envolve várias varreduras em todo o espaço de estados, a cada iteração.

Algoritmo 1: Iteração de Política

$\pi_0 \leftarrow \mathcal{D}$;
$n \leftarrow 0$;
repeat
  Resolver o sistema linear:
  $V_n(s) = \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_n(s'), \forall s$;
  foreach $s \in \mathcal{S}$ **do**
    $\pi(s) \in \text{argmax}_{a \in A} \left\{ \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_n(s') \right\}$;
  $n \leftarrow n + 1$;
until $\pi_n = \pi_{n+1}$;
return V_n, π_{n+1}

2.2.6.2 Iteração de valor

O algoritmo de iteração de valor adota uma abordagem diferente para obter a política ótima. Em vez de manipulá-la diretamente, o algoritmo a obtém por meio do cálculo de uma função de valor ótima. Isso é feito percorrendo o espaço de estados e atribuindo a cada estado o valor máximo estimado, com base no valor descontado de seus estados vizinhos.

Em essência, o algoritmo combina uma versão truncada da etapa de avaliação com a etapa de aprimoramento de políticas, que consiste na seguinte regra de atualização:

$$\begin{aligned} V_{t+1}(s) &= \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') \times \left(\mathcal{R}(s,a,s') + \gamma \times V_k^{\pi}(s') \right) \\ &= \max_{a \in \mathcal{A}} Q_{t+1}(s,a) \end{aligned} \tag{2.11}$$

De forma resumida, o algoritmo funciona da seguinte maneira: iniciando com uma função de valor V_0 em todos os estados, o algoritmo atualiza iterativamente o valor de cada estado de acordo com a equação anterior para obter a próxima função de valor V_t $(t = 1, 2, 3, ...)$, produzindo uma sequência de funções de valor:

$$V_0 \rightarrow V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4 \rightarrow V_5 \rightarrow V_6 \rightarrow V_7 \rightarrow \cdots \rightarrow V^*$$

O algoritmo também produz as funções de valor intermediárias Q, de modo que a sequência correta é:

$$V_0 \to Q_1 \to V_1 \to Q_2 \to V_2 \to Q_3 \to V_3 \to Q_4 \to V_4 \to \cdots \to V^*$$

Esse cálculo continua até que a alteração máxima no valor de todos os estados em cada varredura seja menor do que um número positivo predefinido, indicado como ϵ. Quanto menor esse valor, maior a precisão do algoritmo. O algoritmo exige que cada estado seja processado apenas uma vez em cada varredura do espaço de estados e, assim, elimina uma das desvantagens da iteração de política, que é o excesso de varreduras no espaço de estados da etapa de avaliação de políticas.

Por fim, é importante mencionar que esse algoritmo é flexível e não exige que o valor dos estados seja computado em nenhuma ordem estrita nem com igual frequência para convergir, desde que todos os estados sejam processados durante uma varredura (MOORE; ATKESON, 1993). Isso fornece a flexibilidade de que os valores dos estados possam ser calculados em qualquer ordem, usando quaisquer valores de outros estados que estejam disponíveis.

Algoritmo 2: Iteração de Valor

$V_0 \leftarrow \mathcal{V}$;
$n \leftarrow 0$;
repeat
  for $s \in \mathcal{S}$ **do**
    $V_{n+1}(s) \leftarrow \max_{a \in A} \left\{ \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_n(s') \right\}$;
  end
  $n \leftarrow n + 1$;
until $|V_{n+1} - V_n| < \epsilon$;
for $s \in \mathcal{S}$ **do**
  $\pi(s) \in \operatorname{argmax}_{a \in A} \left\{ \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s') V_n(s') \right\}$;
end
return V_n, π

2.2.6.3 Outros métodos

Existe uma variedade de outros métodos baseados em agentes que podem ser usados para resolver MDPs. Esses métodos se enquadram principalmente em duas classes: métodos de Monte Carlo e métodos de aprendizado de Diferenças Temporais.

Os métodos de Monte Carlo são baseados na média de retornos amostrais para a solução de problemas. Eles diferem da Programação Dinâmica, pois não exigem um modelo

completo do ambiente nem passam por um período de inicialização, mas a estimativa para cada estado é independente.

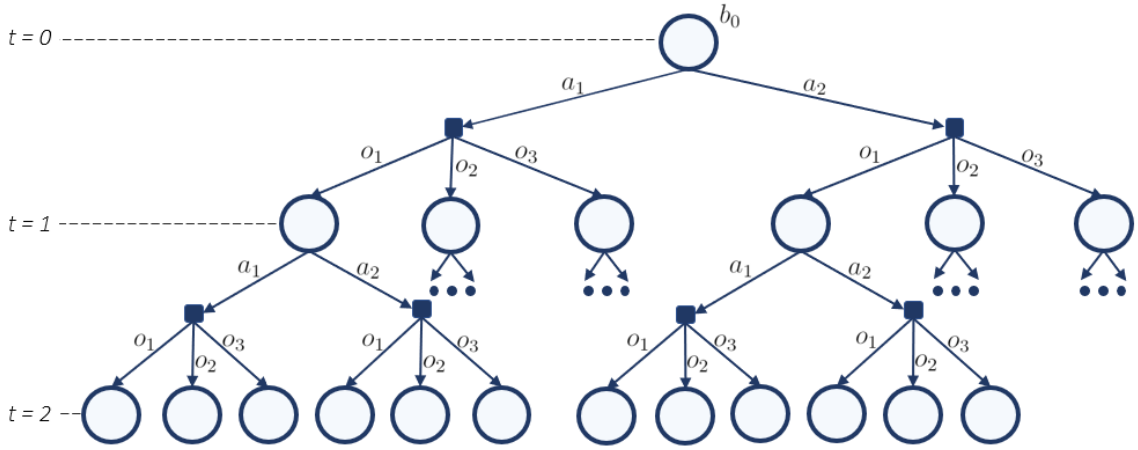
Os métodos de Diferenças Temporais, no entanto, combinam qualidades de ambas as classes de métodos, pois não requerem um modelo completo do ambiente e não passam pela fase de inicialização. Os métodos de DT são implementados de maneira on-line e totalmente incrementais. Dois métodos de DT bem conhecidos que merecem destaque são Q-Learning e Sarsa.

2.3 Processo de Decisão de Markov Parcialmente Observável

Os Processos de Decisão de Markov Parcialmente Observáveis (POMDPs) constituem um modelo probabilístico para problemas de planejamento que incluem estado oculto e incerteza nos efeitos da ações. Essa estrutura matemática é capaz de modelar um agente atuando sob incerteza. A cada etapa, o agente seleciona uma ação que tenha algum resultado estocástico e recebe uma observação com um certo nível de ruído.

Essa sequência de eventos pode ser vista como uma estrutura em árvore (Figura 3), cujos nós representam pontos em que o agente deve tomar uma decisão. Cada nó é rotulado com uma crença, b, que o agente teria se o alcançasse. O nó raiz é rotulado com uma crença inicial, b_0. Caso esteja em um nó b, escolha uma ação a e receba uma observação o, o agente prosseguirá para uma nova crença $\tau\Big(b, a, o\Big)$, correspondente a um dos filhos de b na estrutura da árvore (SMITH; SIMMONS, 2012a).

Figura 3 – Estrutura em árvore de um modelo POMDP.

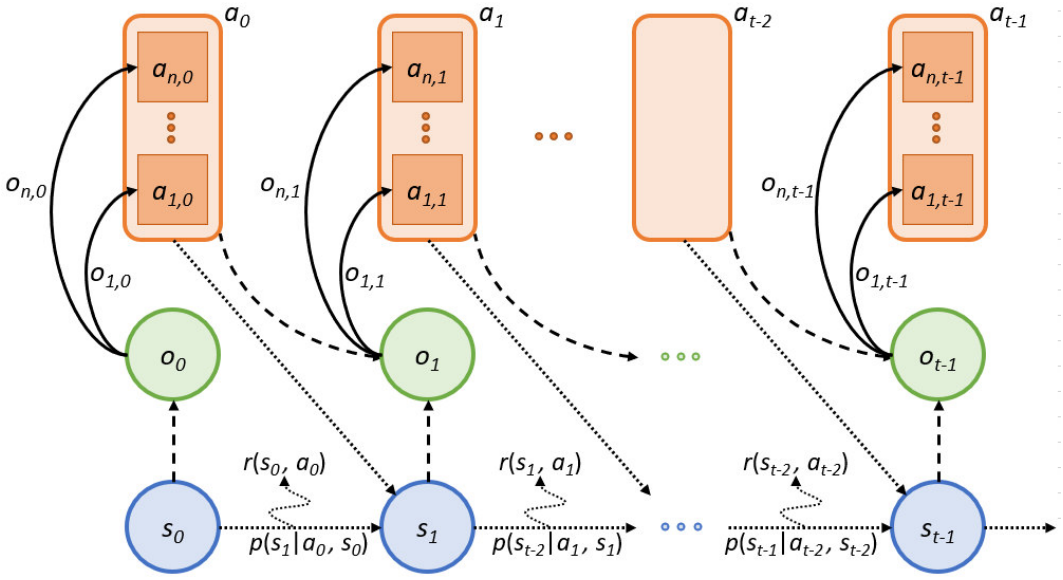


Fonte – Elaborada pelo Autor.

Assim, a cada instante t de T, o agente não sabe com exatidão em qual estado atual $s_t \in \mathcal{S}$ se encontra, mas possui apenas uma visão limitada por meio de uma observação $o_t \in \Omega$. Essa observação é dada pela função de observação $\mathcal{O}$. Quando o agente executa uma ação $a_t \in \mathcal{A}$, o estado do processo muda estocasticamente de acordo com $\mathcal{T}$, alcançando

o novo estado s_{t+1}. O agente recebe apenas a observação o_{t+1}, gerada de acordo com $\mathcal{O}$. Então, o agente recebe uma recompensa $r \in \mathcal{R}$, conforme ilustra a Figura 4.

Figura 4 – Representação do funcionamento de um modelo POMDP.



Fonte – Elaborada pelo Autor.

2.3.1 Definição formal

Formalmente, o modelo é representado por uma tupla $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O}, b_0 \rangle$, em que $\mathcal{S}$ é o espaço de estados, $\mathcal{A}$ é o espaço de ações, $\mathcal{T}$ é a função de transição estocástica, $\mathcal{R}$ é a função de recompensa, Ω é o espaço de observações, $\mathcal{O}$ é a função de observação estocástica e b_0 é o estado de crença inicial (PAS, 2012).

O espaço de estados $\mathcal{S}$, o espaço de ações $\mathcal{A}$, a função de transição $\mathcal{T}$ e a função de recompensa $\mathcal{R}$ são exatamente os mesmos mostrados no Processo de Decisão de Markov, razão pela qual não serão abordados neste tópico.

O espaço de observações Ω é um conjunto finito $\{o_1, o_2, ..., o_M\}$ em que o tamanho do espaço de observações é M, ou seja, $|\Omega| = M$. Ele representa as observações parciais que o agente é capaz de experimentar em seu ambiente (RENS; FERREIN; POEL, 2008).

A função de observação $\mathcal{O}$ é definida como $\mathcal{O} : \mathcal{S} \times \mathcal{A} \rightarrow \Pr\left(\Omega\right)$. Essa função é responsável por especificar a probabilidade de se perceber a observação o no estado s' depois de se executar a ação a, fornecendo, para cada ação do agente e estado resultante, uma distribuição de probabilidade sobre as observações (RENS; FERREIN; POEL, 2008).

Finalmente, o estado de crença inicial b_0 define uma distribuição de probabilidade inicial com relação a todos os estados do ambiente presentes em $\mathcal{S}$ (RENS; FERREIN; POEL, 2008).

2.3.2 Critérios de Desempenho

O critério de desempenho usualmente adotado em POMDPs é o modelo de horizonte infinito. Segundo esse critério, a recompensa de longo prazo é levada em consideração, mas as recompensas recebidas no futuro são descontadas de acordo com a distância que serão recebidas no tempo. Um fator de desconto γ, com $0 \leq \gamma < 1$, é usado para isso. Nesse caso, as recompensas obtidas no futuro são descontadas mais do que as obtidas no presente. Esse fator de desconto garante que, mesmo com horizonte infinito, a soma das recompensas obtidas seja finita (OTTERLO; WIERING, 2012). O fator de desconto é calculado pela seguinte expressão:

$$\mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t \times r_t\right] \tag{2.12}$$

2.3.3 Estados de Crença

De acordo com (PAS, 2012), em ambientes em que o estado não é diretamente observável, o agente precisa derivar seu comportamento a partir do histórico completo de ações e observações realizadas até o momento atual t, definido como segue:

$$h_t = \left\{a_0, o_1, a_1, o_2, \cdots, a_{t-1}, o_t\right\} \tag{2.13}$$

Representar o histórico de ações e observações (h_t) seria muito dispendioso, pois, com o passar do tempo, demandaria muita memória. Assim, o modelo utiliza apenas um conjunto $o \in \Omega$ de observações para determinar o provável estado atual, representando apenas as elementos relevantes por meio de uma distribuição de probabilidade sobre o espaço de estados $\mathcal{S}$, chamado de estado crença (JUNIOR; FREITAS et al., 2018). Um estado crença b_t no tempo t é uma estatística suficiente para o histórico h_t, sendo definido pela Equação 2.14:

$$b_t\Big(s\Big) = \Pr\Big(s_t = s | h_t, b_0\Big) \tag{2.14}$$

Desse modo, o agente pode tomar uma decisão com base no seu estado de crença ao invés de consultar todos as ações e observações passadas. A qualquer tempo t, o estado de crença b_t pode ser recalculado a partir do estado anterior b_{t-1} utilizando a ação prévia a_{t-1} e a observação atual o_t (JUNIOR; FREITAS et al., 2018). Isso é feito mediante uma função de atualização do estado de crença $\tau\Big(b, a, o\Big)$, em que $b_t = \tau\Big(b_{t-1}, a_{t-1}, o_t\Big)$ é definida pela

Equação 6.3:

$$\begin{aligned} b_t\Big(s'\Big) &= \tau\Big(b_{t-1}, a_{t-1}, o_t\Big)\Big(s'\Big) \\ &= \frac{1}{\Pr\Big(o_t|b_{t-1}, a_{t-1}\Big)}\mathcal{O}\Big(s', a_{t-1}, o_t\Big)\sum_{s\in S}\mathcal{T}\Big(s, a_{t-1}, s'\Big)b_{t-1}\Big(s\Big) \end{aligned} \tag{2.15}$$

Por fim, o estado de crença é inicializado de acordo com uma distribuição de probabilidade específica, $b_{t=0} = b_0$, que representa o grau inicial de incerteza do agente com relação ao estado real do ambiente. A escolha da distribuição b_0 depende do domínio. Se o agente não souber qual é o verdadeiro estado, a crença inicial é geralmente uniforme no espaço de estado $\mathcal{S}$ (PAS, 2012).

2.3.4 Política

Uma política $\pi : \mathcal{B} \to \mathcal{A}$ é um mapeamento do espaço de crenças $\mathcal{B}$ para o espaço de ação $\mathcal{A}$. Ele prescreve uma ação $\pi\Big(b\Big) \in \mathcal{A}$ na crença $b \in \mathcal{B}$ (SOMANI et al., 2013). Para um horizonte infinito, o valor de uma política π para uma crença b é a recompensa total com desconto esperada que o agente receberá ao executar π:

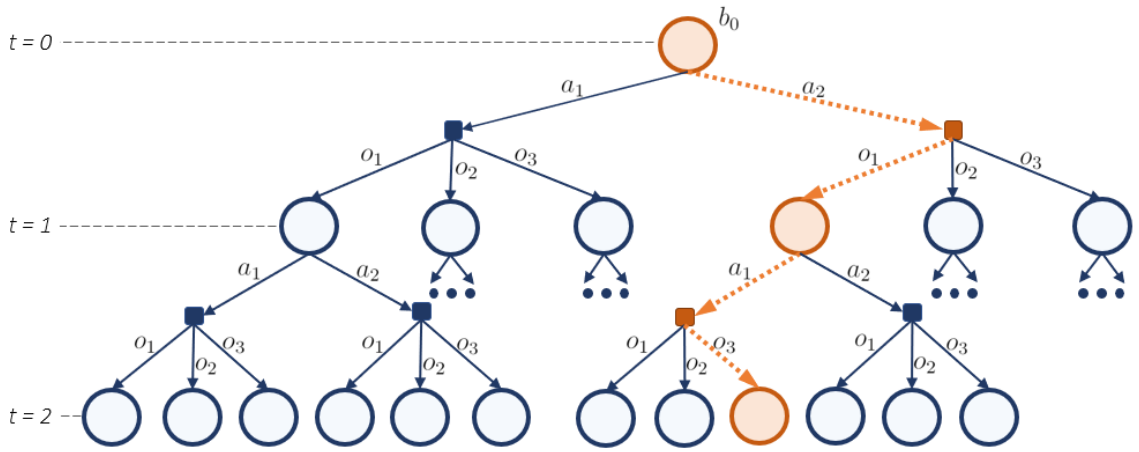
$$V_\pi\Big(b\Big) = \mathbb{E}\Bigg(\sum_{t=0}^{\infty}\gamma^t\mathcal{R}\Big(s_t, \pi\Big(b_t\Big)\Big) \mid b_0 = b\Bigg) \tag{2.16}$$

A aplicação de uma política a um POMDP é feita da seguinte forma: o agente começa com uma crença inicial. A cada passo, ele procura uma ação ótima a^* na crença atual b. O agente executa a ação a^* e recebe uma nova observação o. Então, o agente atualiza a crença usando a Equação 6.3. Em seguida, o processo se repete.

De acordo com (SOMANI et al., 2013), para procurar uma ação ideal a^*, pode-se construir uma árvore de crenças (Figura 5), sendo a crença inicial b_0 representada como a raiz da árvore. O agente busca nessa árvore uma política π que maximize o valor $V_\pi\Big(b_0\Big)$ em b_0, e define $a^* = \pi\Big(b_0\Big)$. Cada nó da árvore representa uma crença. Um nó divide-se em um conjunto de ações de tamanho $|\mathcal{A}|$ e cada ação ramifica-se ainda mais em um conjunto de observações de tamanho $|\Omega|$.

Por fim, (SOMANI et al., 2013) afirma que, para obter uma política aproximadamente ótima, pode-se truncar a árvore em uma profundidade máxima e buscar nela uma política ótima. Nesse caso, em cada nó folha, simula-se uma política padrão para obter uma estimativa de limite inferior do valor ótimo. Essa política padrão pode ser uma política aleatória ou heurística.

Figura 5 – Aplicação de uma política em uma árvore de crenças.



Fonte – Elaborada pelo Autor.

2.3.5 Função de valor

Cada árvore de políticas p está associada a uma função V_p. Conforme (KAELBLING; LITTMAN; CASSANDRA, 1998), se p é uma árvore que possui uma única etapa (uma única ação), o valor da execução da ação a no estado s é dado pela Equação 2.17, em que $a\big(p\big)$ é a ação especificada no nó superior da árvore p.

$$V_p\big(s\big) = \mathcal{R}\big(s, a\big(p\big)\big) = \sum_{s' \in \mathcal{S}} \mathcal{T}\big(s, a\big(p\big), s'\big)\mathcal{R}\big(s, a\big(p\big), s'\big) \tag{2.17}$$

Ainda de acordo com (KAELBLING; LITTMAN; CASSANDRA, 1998), se p é uma árvore de t-etapas, o valor da execução da ação a no estado s pode ser calculado pela Equação 2.18, em que $p(o)$ é uma subárvore de $\big(t-1\big)$ etapas associada à observação o. O valor esperado é calculado assumindo-se uma expectativa sobre os próximos estados possíveis, s', e depois considerando o valor de cada um desses estados. O valor também depende de qual subárvore será executada e da observação que será feita. Por isso, deve-se assumir outra expectativa, com relação às possíveis observações, do valor de executar a subárvore, $p(o)$, começando no estado s'.

$$V_p\big(s\big) = \mathcal{R}\big(s, a\big(p\big)\big) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}\big(s, a\big(p\big), s'\big) \sum_{o \in \Omega} \mathcal{O}\big(s', a\big(p\big), o\big) V_{p(o)}\big(s'\big) \tag{2.18}$$

Como o agente nunca saberá o estado exato do ambiente, deve poder determinar o valor da execução de uma árvore de políticas, p, a partir de algum estado de crença b. Conforme (KAELBLING; LITTMAN; CASSANDRA, 1998), isso pode ser calculado como

uma expectativa sobre os possíveis estados do ambiente de executar p em cada estado s:

$$V_p\Big(b\Big) = \sum_{s \in S} b\Big(s\Big) V_p\Big(s\Big) \tag{2.19}$$

Se a função de valor for definida como um vetor, $V_p = \langle V_p\Big(s_1\Big), \cdots, V_p\Big(s_n\Big)\rangle$, então a equação 2.19 pode ser representada como um produto interno entre b e V_p:

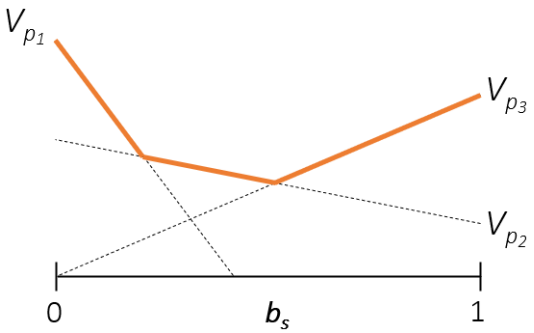
$$V_p\Big(b\Big) = b \cdot V_p \tag{2.20}$$

Agora, pode-se calcular o valor esperado resultante da execução de uma árvore de políticas p para todos os estados de crença possíveis. No entanto, para se construir uma política ótima de t etapas, faz-se necessário executar diferentes árvores de políticas a partir de diferentes estados de crença inicial. Seja $\mathcal{P}$ o conjunto finito de todas as ávores de crença de tamanho t. Então,

$$V_t(b) = \max_{p \in \mathcal{P}} V_p\Big(b\Big) = \max_{p \in \mathcal{P}} b \cdot V_p \tag{2.21}$$

Essa definição da função de valor leva a propriedades geométricas importantes. Cada árvore de política, p, induz uma função de valor V_p que é linear em b e V_t é a superfície superior dessa coleção de funções. Portanto, V_t é uma função linear por partes e convexa (KAELBLING; LITTMAN; CASSANDRA, 1998). Considerando um ambiente com apenas dois estados, V_t é o valor máximo de todos os vetores V_{p_i} em cada ponto do espaço de crenças, conforme ilustra a Figura 6.

Figura 6 – Função de valor ótima para um modelo POMDP de dois estados.



Fonte – Elaborada pelo Autor

Por fim, a política $\pi\Big(b\Big)$, aproximada por uma árvore de políticas, é uma função do estado de crença b que retorna uma árvore que maximiza o valor de $V\Big(b\Big)$.

$$\pi(b) = \operatorname*{argmax}_{p \in \mathcal{P}} V_p\Big(b\Big) \tag{2.22}$$

2.3.6 Algoritmos

No modelo POMDP, um agente toma as decisões sequenciais de forma a maximizar sua utilidade, dadas as ações e os estados atuais. Os algoritmos para resolver um modelo POMDP usam vários métodos, tais como: iteração de valor (SAWAKI; ICHIKAWA, 1978) (CASSANDRA; KAELBLING; LITTMAN, 1994), iteração de política (SONDIK, 1978), iteração de valor acelerada (III; SCHERER, 1989), representação estruturada (BOUTILIER; POOLE, 1996) e aproximação (ZHANG; LIU, 1996).

Para cada um desses métodos, existem técnicas diferentes. Por exemplo, métodos baseados em pontos, como a iteração de valor baseada em pontos (PINEAU et al., 2003) e a iteração de valor de busca heurística (SMITH; SIMMONS, 2012b), receberam atenção recente por sua capacidade de resolver problemas relativamente grandes. Outro exemplo é o QMDP, um método de aproximação que usa uma função de valor de estado-ação $Q\big(s, a\big)$ para aproximar os vetores V_p (LITTMAN; CASSANDRA; KAELBLING, 1995).

Por último, há também abordagens baseadas em métodos de Monte-Carlo, como o algoritmo POMCP (*Partially Observable Monte-Carlo Planning*) (SILVER; VENESS, 2010). Esses métodos avaliam os estados por meio amostragem, a partir de simulações quase aleatórias. A abordagem considera que não há muito a ser aprendido com uma única simulação aleatória, mas que, com muitas simulações, uma estratégia bem-sucedida pode ser derivada.

2.4 Considerações Finais

Com frequência, os professores de algoritmos são responsáveis por apresentar problemas computacionais do mundo real aos alunos, explorando-os em sala de aula e incentivando a resolução de problemas similares. Naturalmente, isso demanda um tempo precioso dos professores, pois requer a seleção e a ordenação dos problemas, assim como o acompanhamento individualizado dos alunos, como forma de tirar eventuais dúvidas e evitar que eles fiquem entediados ou desmotivados durante o aprendizado.

Porém, o esforço exigido por essa tarefa de organização dos problemas acaba, muitas vezes, tornando-a impraticável, ainda mais se considerarmos que cada aluno possui um ritmo particular de aprendizado. Nesse cenário, pode-se encarar essa etapa de planejamento como um processo de tomada de decisão sequencial, em que o professor é um agente responsável por monitorar as soluções submetidas pelos alunos e por apresentar novos problemas de forma sequencial, com base nas respostas obtidas anteriormente.

Em específico, deve-se ter em mente que os professores conseguem apenas fazer suposições quanto ao ganho cognitivo proporcionado pela resolução de um determinado problema, mas não são capazes de mensurá-lo com exatidão, justamente porque essa

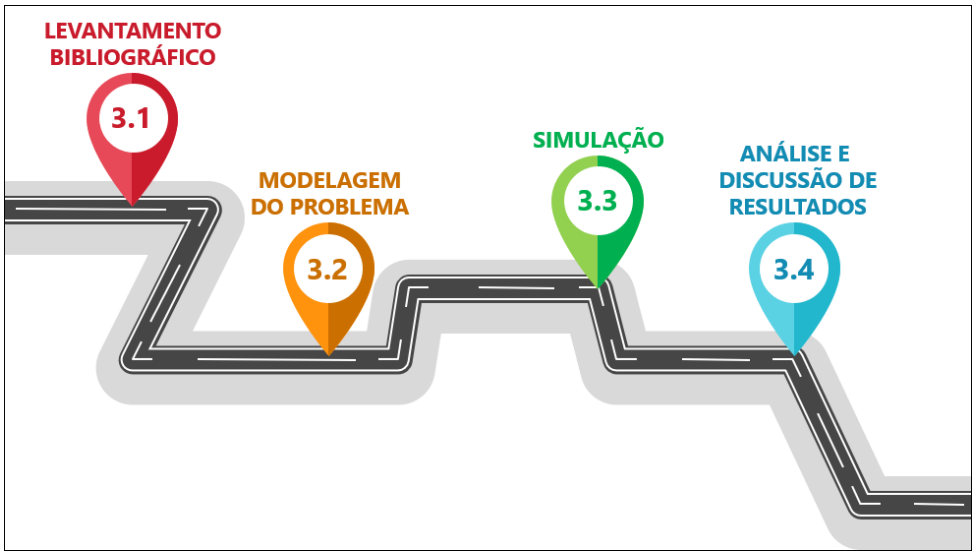
característica é intrínseca ao aluno. Enquanto alguns alunos conseguirão dominar um assunto com a resolução de apenas um problema, outros não. Esse processo de tomada de decisões que ocorre em meio a incertezas quanto ao conhecimento do aluno pode ser modelado por meio de Processos de Decisão de Markov Parcialmente Observáveis (WOOLF, 2010) (WANG, 2018).

Em virtude das incertezas concernentes ao processo de ensino de algoritmos, verificou-se uma oportunidade de se utilizar Processos de Decisão de Markov Parcialmente Observáveis para modelar o sequenciamento de problemas computacionais, propiciando uma forma mais ágil, dinâmica e individualizada de ensino. Como forma de avaliar o modelo proposto, serão comparadas políticas de sequenciamento geradas de forma aleatória versus políticas produzidas por algoritmos específicos, tais como POMCP e QMDP.

3 Metodologia

Este capítulo tem o propósito de apresentar a metodologia utilizada para alcançar os objetivos deste trabalho, a qual foi dividida em quatro etapas principais: levantamento bibliográfico, apresentado na seção 3.1; modelagem do problema, definida na seção 3.2; simulação, cujo escopo é delineado na seção 3.3; e análise e discussão dos resultados, descrita na seção 3.4. A Figura 7 apresenta uma visão geral dessas etapas.

Figura 7 – Metodologia do Trabalho



Fonte – Elaborada pelo Autor.

3.1 Etapa de Levantamento bibliográfico

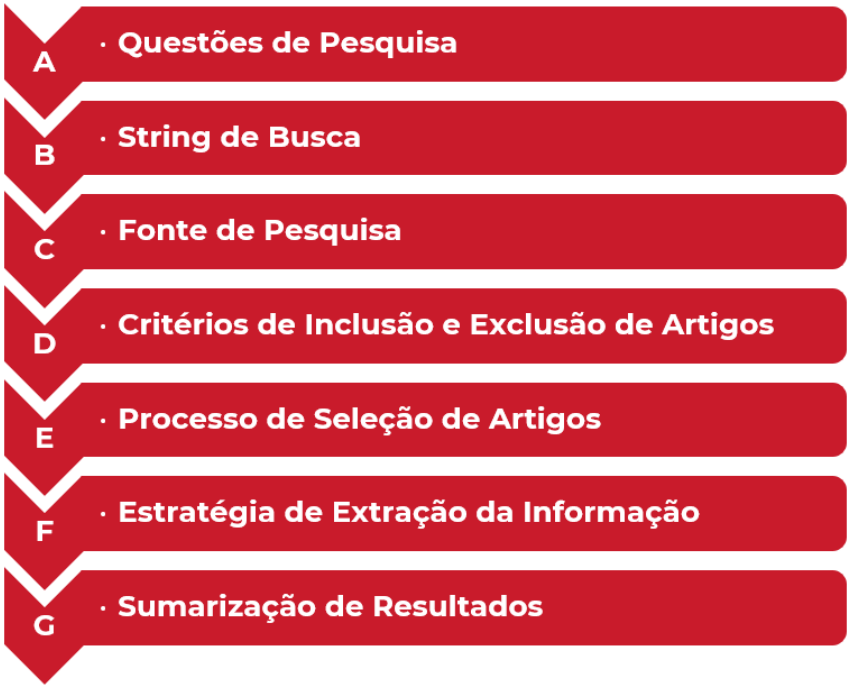
A primeira etapa do trabalho consiste na elaboração de uma Revisão Sistemática da Literatura (RSL), cujo objetivo principal é descobrir quais técnicas de inteligência artificial (IA) podem ser empregadas na personalização de Sistemas Tutores Inteligentes (STI) voltados ao ensino de programação.

De acordo com (KITCHENHAM; CHARTERS, 2007), a RSL é um instrumento para identificar, avaliar e interpretar os estudos relevantes a uma questão de pesquisa em particular, um determinado tópico, ou um fenômeno de interesse. Por meio desse procedimento, consegue-se identificar publicações relacionadas a um tema específico; definir critérios para seleção de estudos; formalizar a extração de dados; sumarizar as evidências encontradas; reproduzir uma outra revisão sistemática.

As atividades realizadas nesta etapa estão descritas na Figura 8. Cada atividade

possui um objetivo específico bem definido, conforme mostrado na sequência:

Figura 8 – Atividades da Etapa de Levantamento Bibliográfico.



Fonte – Elaborada pelo Autor.

- **Questões de Pesquisa**: define-se o que deve ser respondido quanto ao tópico de pesquisa abordado. As questões de pesquisa podem ser classificadas como primárias ou secundárias. A questão de pesquisa primária (ou principal) define o tópico de pesquisa que está sendo investigado. Já as questões secundárias fornecem subsídios para entender as especificidades do tópico de pesquisa estudado.

- ***String* de Busca**: define-se uma cadeia de caracteres com base em um conjunto de palavras-chave que compõem o tema de pesquisa. Esses termos devem ser relacionados por meio de expressões lógicas (AND e OR). A *string* de busca é utilizada como entrada para os mecanismos de busca.

- **Fonte de Pesquisa**: define-se o repositório em que as publicações são procuradas. Ou seja, onde podem ser encontrados os artigos, periódicos e outros elementos de base acadêmica utilizados no trabalho.

- **Critérios de Inclusão e Exclusão de Artigos**: definem-se os critérios para inclusão ou exclusão de uma publicação no processo de consulta às fontes de pesquisa. A depender da *string* de busca, a consulta de elementos nas fontes de pesquisa pode se tornar cansativa. Devido a isso, é necessário que existam critérios claros de inclusão e exclusão de elementos com base nas questões de pesquisa.

- **Processo de Seleção de Artigos**: define-se o processo de filtragem das publicações. Geralmente, realiza-se um primeiro filtro com base na leitura do título e do resumo dos artigos, visando identificar os que atendem aos critérios de seleção. Depois, as publicações passam por um segundo crivo, que consiste na leitura completa do documento, para avaliar se os critérios de seleção são realmente atendidos. O processo de seleção pode ser executado de forma manual ou automática, baseado nos critérios de inclusão e exclusão.

- **Estratégia de Extração de Informação**: define-se a estratégia para a extração de informação dos elementos obtidos nas etapas anteriores. Normalmente, desenvolve-se um formulário de extração que permite responder às questões secundárias. Além disso, os dados coletados com o formulário de extração são utilizados para agregar os resultados de estudos que possuem contextos, hipóteses e métricas similares.

- **Sumarização de Resultados**: consolidam-se os resultados da revisão sistemática, apresentando todos os elementos levantados em resposta às questões de pesquisa;

Os resultados desta etapa são apresentados no Capítulo 4 e servem de base para a etapa de modelagem, descrita na seção 3.2.

3.2 Etapa de Modelagem do Problema

Como resultado da etapa anterior, tem-se que diversas técnicas de Inteligência Artificial podem ser empregadas no contexto da personalização de STIs, em especial, *knowledge tracing*, redes bayesianas, redes neurais, árvores de decisão e Processos de Decisão de Markov (MDP).

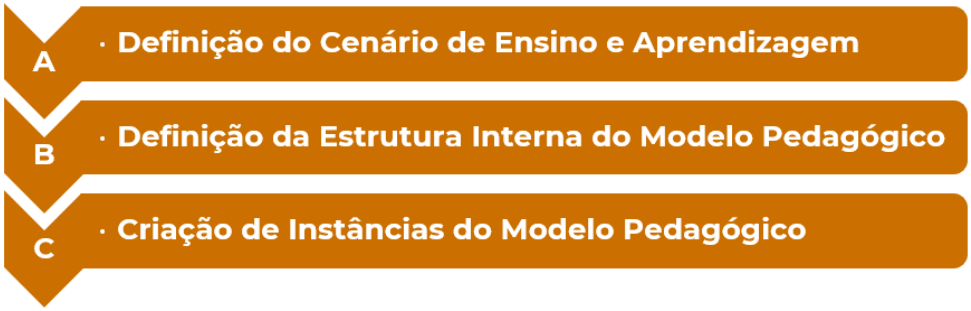
De forma particular, quando a finalidade do trabalho é o sequenciamento de exercícios, a abordagem normalmente utilizada fundamenta-se em Processos de Decisão de Markov (MDP) e Processos de Decisão de Markov Parcialmente Observáveis (MDP), o que justifica a realização da presente pesquisa.

O cerne dessa etapa consiste na definição da estrutura interna do modelo responsável por selecionar as estratégias pedagógicas mais apropriadas para orientar o processo de ensino. Esse componente é responsável por processar um conjunto de observações e tomar decisões quanto às ações de ensino.

Essa etapa também é responsável por construir um conjunto de instâncias do modelo pedagógico para diferentes números de problemas computacionais. Essa atividade exige que alguns desafios sejam enfrentados e superados, tais como a parametrização das probabilidades de transição e observação; e o crescimento exponencial do modelo.

Dessa forma, essa etapa consiste na análise e modelagem do problema. Para tanto, realizam-se as atividades descritas na Figura 9, as quais são detalhadas na sequência.

Figura 9 – Atividades da Etapa de Detalhamento da Abordagem.



Fonte – Elaborada pelo Autor.

- **Definição do Cenário de Ensino e Aprendizagem**: faz-se uma breve descrição do cenário de ensino e aprendizagem que se pretende modelar, detalhando o domínio de conhecimento; o contexto em que ocorre o processo de ensino; a forma de exposição do conteúdo ao aluno; o mecanismo de avaliação adotado; a necessidade de presença de um tutor humano; etc;
- **Definição da Estrutura do Modelo Pedagógico**: define-se a estrutura interna do modelo responsável por selecionar as estratégias pedagógicas mais apropriadas para orientar o processo de ensino. Esse componente é responsável por processar um conjunto de observações e tomar decisões quanto às ações de ensino. Para isso, o modelo deve realizar os seguintes passos: estimar o estado de conhecimento do estudante; atualizar as probabilidades de transição; calcular as recompensas; calcular e aplicar as novas políticas;
- **Criação de Instâncias do Modelo Pedagógico**: criam-se instâncias do modelo definido na etapa anterior. Para isso, as probabilidades de transição e observação são padronizadas para todas as instâncias do modelo pedagógico. Além disso, o problema do crescimento exponencial faz com que o tempo necessário para codificar o modelo no pacote JuliaPOMDP também cresça na mesma proporção. Por isso, faz-se necessária a criação de uma ferramenta capaz de gerar instâncias do modelo pedagógico codificadas na linguagem Julia, de forma que seja compatível com o pacote JuliaPOMDP.

Os resultados desta etapa são apresentados no Capítulo 5 e servem de base para a etapa de simulação, descrita na seção 3.3.

3.3 Etapa de Simulação

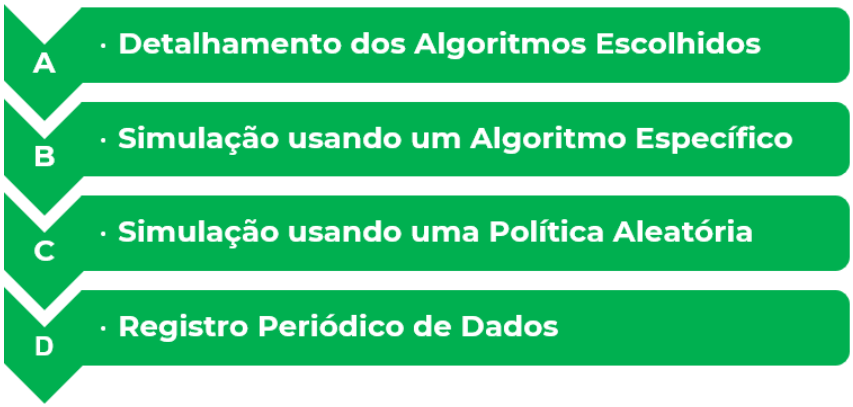
Os algoritmos utilizados para resolver as instâncias do modelo pedagógico durante a simulação são apresentados nesta etapa, dentre os disponíveis no pacote JuliaPOMDP. Essa ferramenta oferece suporte a um conjunto expressivo de algoritmos para a resolução de Processos de Decisão de Markov Parcialmente Observáveis (POMDP), tais como QMDP, SARSOP, FIB, POMCP, dentre outros.

Nesse estágio ocorre a execução das instâncias criadas na etapa de modelagem. Assim, para cada instância, os algoritmos previamente definidos são executados repetidas vezes para diferentes tamanhos de horizontes. Também é executada uma política aleatória seguindo os mesmos parâmetros adotados para os algoritmos de planejamento específicos.

Como resultado dessa etapa, obtém-se um conjunto de dados significativos: recompensa descontada média, tempo de execução dos algoritmos e as melhores políticas identificadas. O registro desses dados viabiliza a comparação de um algoritmo específico contra uma política aleatória para uma mesma instância do modelo.

As atividades realizadas nesta etapa estão descritas na Figura 10, as quais são detalhadas na sequência.

Figura 10 – Atividades da Etapa de Simulação.



Fonte – Elaborada pelo Autor.

- **Detalhamento dos Algoritmos Escolhidos**: definem-se os algoritmos de planejamento empregados na simulação e detalham-se suas principais características;
- **Simulação usando um Algoritmo Específico**: os algoritmos previamente definidos são executados para uma determinada instância, até ocorrer sua convergência, e os resultados são registrados;
- **Simulação usando uma Política Aleatória**: uma política aleatória é executada para uma instância e os resultados são registrados;

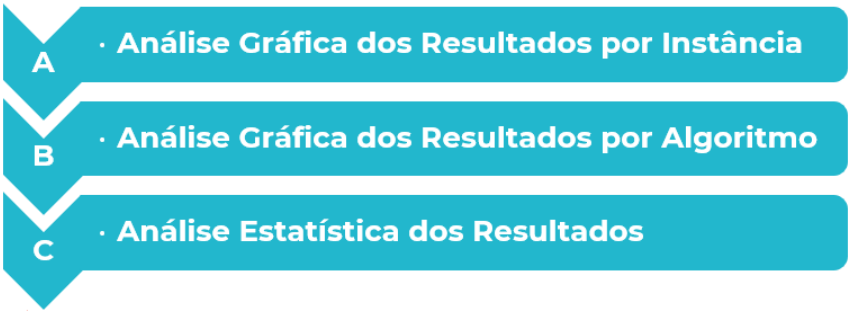
- **Registro Periódico de Dados**: ao longo das execuções, são periodicamente registrados, tanto para um algoritmo específico quanto para uma política aleatória: o número de simulações; o tempo total de execução; a melhor política identificada; e a recompensa descontada média.

Os resultados desta etapa são apresentados no Capítulo 6 e servem de base para a etapa de análise e discussão de resultados, descrita na seção 3.4.

3.4 Etapa de Análise e Discussão de Resultados

Nessa etapa são analisados os dados obtidos na etapa de simulação. Além disso, os resultados provenientes da execução de um algoritmo específico são comparados com os resultados de uma política aleatória. Essas atividades estão descritas na Figura 11, as quais são detalhadas na sequência.

Figura 11 – Atividades da Etapa de Análise e Discussão de Resultados.



Fonte – Elaborada pelo Autor.

- **Análise Gráfica dos Resultados por Instância**: elaboram-se gráficos comparativos da recompensa descontada média e do tempo médio de execução por instância;
- **Análise Gráfica dos Resultados por Altoritmo**: apresentam-se gráficos comparativos da recompensa descontada média e do tempo médio de execução por algoritmo;
- **Análise Estatística dos Resultados**: comparam-se os resultados dos algoritmos específicos e da política aleatória, por meio da avaliação de hipóteses. Para tanto, supõe-se que, se o modelo é capaz de representar adequadamente o sequenciamento das questões, deve haver uma política gerada por um algoritmo específico capaz de superar o resultado alcançado por uma política aleatória.

Os resultados desta etapa são apresentados no Capítulo 7.

4 Levantamento Bibliográfico

De acordo com (KITCHENHAM; CHARTERS, 2007), a revisão sistemática da literatura é um instrumento para identificar, avaliar e interpretar os estudos relevantes a uma questão de pesquisa em particular, a um determinado tópico, ou a um fenômeno de interesse.

Eesse procedimento simplifica a identificação de publicações relacionadas a um tema específico; a definição de critérios claros para seleção de estudos; a formalização da extração de dados; a sumarização das evidências encontradas; bem como possibilita a reprodução da revisão da literatura, com a inclusão de novas publicações.

4.1 Planejamento

As atividades de planejamento foram realizadas com base nas diretrizes propostas por (KITCHENHAM; CHARTERS, 2007). Nesse sentido, fez-se a definição das questões de pesquisa, da estratégia de busca; da *string* de busca; das fontes de pesquisa; dos critérios de seleção; e da estratégia para extração de dados, conforme apresentado nas seções seguintes.

4.1.1 Definição das questões de pesquisa

De acordo com (KITCHENHAM; CHARTERS, 2007), as questões de pesquisa definem o que deve ser respondido quanto ao tópico de pesquisa abordado, podendo ser classificadas como primárias ou secundárias. A questão de pesquisa primária (ou principal) define o tópico de pesquisa que está sendo investigado. Já as questões secundárias fornecem subsídios para entender as especificidades do tópico de pesquisa estudado.

A questão de pesquisa primária foi definida com base nos critérios propostos por (KITCHENHAM; CHARTERS, 2007): (P) população, o tipo de sistema a ser observado; (I) intervenção, o método ou procedimento a ser investigado; (C) comparação, o método ou procedimento com que será feito o comparativo; e (O) resultados, o efeito a ser investigado. Desse modo, a questão principal (QP) foi estruturada da seguinte forma: quais técnicas de IA (I) podem ser aplicadas a fim de se personalizar Sistemas Tutores Inteligentes (STI) voltados ao ensino de cursos introdutórios de programação (P)?

As questões de pesquisa secundárias guiam a coleta de dados para a identificação de detalhes relacionados à questão principal. Assim, tendo como base a questão formulada, ficaram estabelecidas as seguintes questões secundárias: (QS01) qual a linguagem de

programação ensinada; (QS02) qual a estratégia de ensino adotada? (QS03) qual a técnica de inteligência artificial empregada? (QS04) qual a finalidade da técnica empregada?

4.1.2 Definição da *string* de busca

A *string* de busca foi construída com base nos vários termos correlacionados ao ensino introdutório de programação e aos Sistemas Tutores Inteligentes (STI). Esses termos foram relacionados por meio de expressões booleanas (AND e OR), conforme mostrado na Figura 12:

Figura 12 – *String* de busca.

```
(
        ("computer science education" OR "introductory programming" OR
         "teach * programming" OR "learn * programming" OR "novice programming" OR
         "cs1" OR "introductory computer science" OR "ensino de programação" OR
         "ensino de algoritmos" OR "introdução à programação" OR
         "programação introdutória" OR "iniciação à ciência da computação")
    AND
        ("intelligent" OR "inteligente" OR "adapt*" OR "cognit*" OR "smart" OR
         "personaliz*" OR "customiz*" OR "individualiz*" OR "sequenc*" OR
         "estratég*" OR "strateg*" OR "policy" OR "política" OR "decision" OR
         "decisão")
    AND
        ("tutor*" OR "mentor*" OR "educa*" OR "instru*")
    AND
        ("system" OR "software" OR "application" OR "sistema"  OR "aplicativo")
)
```

Fonte – Elaborada pelo Autor

Adicionalmente, foi averiguada a necessidade de se incluir outros termos à *string* de busca, para aumentar a quantidade de artigos retornados. Dessa forma, a busca foi repetida para inclusão de sinônimos encontrados nas publicações mais relevantes.

4.1.3 Definição das fontes de pesquisa

A revisão levou em consideração as publicações dos anos de 2013 a 2019. As buscas ocorreram nas bases de dados das bibliotecas digitais da Springer, da ACM e do IEEE. Essas bibliotecas foram escolhidas porque seus motores de busca oferecem um comportamento adequado e permitem o uso de expressões lógicas.

Além disso, os anais do Simpósio Brasileiro de Informática na Educação (SBIE) foram consultados de forma manual, com base apenas no título da publicação, por serem considerados de alta relevância para o tema pesquisado e por não haver um mecanismo de busca que possibilite o uso de expressões lógicas.

4.1.4 Definição dos critérios de seleção

Ao selecionar as publicações que servem de base para responder as questões de pesquisa, deve-se fazer a aplicação dos critérios de inclusão e exclusão em duas etapas, analisando-se dados distintos das publicações.

Inicialmente, realiza-se um primeiro filtro com base na leitura do título e do resumo dos artigos, visando identificar os que atendem aos critérios de seleção. Depois, as publicações passam por um segundo crivo, que consiste na leitura completa do documento, para avaliar se os critérios de seleção são realmente atendidos.

A Tabela 1 apresenta os critérios de inclusão e exclusão adotados para selecionar os artigos retornados após a aplicação da *string* de busca. Os critérios de inclusão são definidos pela sigla "CI-NN", em que NN indica o número do critério. De forma análoga, critérios de exclusão são definidos pela sigla "CE-NN".

Tabela 1 – Critérios de seleção.

ID	DESCRIÇÃO
CI01	A publicação descreve um STI voltado ao aprendizado de linguagens de programação estruturadas.
CI02	A publicação descreve um STI que requer a digitação de linhas de código como forma de avaliação do aprendizado dos alunos.
CI03	A publicação descreve um STI que usa técnicas de Inteligência Artificial (IA) para adaptar/personalizar o processo de tutoria.
CE01	A publicação descreve um STI voltado ao aprendizado de outros temas da Ciência da Computação, tais como linguagens de consulta (SQL), estruturas de dados, orientação a objetos
CE02	A publicação descreve um STI voltado ao aprendizado de linguagens de programação não digitadas (baseadas em blocos)

Fonte – Elaborada pelo Autor

Nesses termos, os critérios de inclusão guiam a busca por estudos que descrevam STIs voltados ao aprendizado de linguagens de programação estruturadas (CI01); que exijam a digitação de linhas de código (CI02), como forma de avaliação do aprendizado dos alunos; e que façam uso de técnicas de inteligência artificial como mecanismo para adaptar/personalizar o processo de tutoria (CI03).

Já os critérios de exclusão ajudam a eliminar os estudos que tenham como foco principal o ensino de tópicos especiais em programação, tais como linguagens de consulta (SQL), estruturas de dados e orientação a objetos (CE01); também auxiliam a filtrar os sistemas voltados ao aprendizado de programação baseada em blocos (CE02), posto que são projetados de forma a se evitar problemas relacionados a erros sintáticos e semânticos.

4.1.5 Definição da estratégia para extração de dados

Com relação à estratégia para extração de dados, elaborou-se um formulário de extração que permite responder às questões secundárias definidas na seção 4.1.1, conforme mostra a Tabela 2. Os dados coletados com o formulário de extração podem ser utilizados para agregar os resultados de estudos que possuem contextos, hipóteses e métricas similares.

Tabela 2 – Formulário para extração de dados.

FORMULÁRIO DE EXTRAÇÃO		
Id da publicação		
Referência completa		
Nome do sistema		
ID	**PERGUNTA**	**POSSÍVEL RESPOSTA**
QS01	Qual é a linguagem de programação ensinada?	[ ] Python [ ] C [ ] C++ [ ] Java [ ] Outra
QS02	Qual a estratégia de ensino adotada?	[ ] Baseada em exemplos [ ] Baseada em simulação [ ] Baseada em diálogos [ ] Baseada em análise de programas [ ] Baseada em *feedback* [ ] Baseada em colaboração
QS03	Qual a técnica de inteligência artificial empregada?	[ ] Model Tracing [ ] Knowledge Tracing [ ] Bayesian Network Model [ ] Neural Network Model [ ] Hidden Markov Model [ ] K-Means [ ] KNN [ ] Decision Trees [ ] Markov Decision Process [ ] Outra
QS04	Qual a finalidade da técnica aplicada?	[ ] Feedback [ ] Dicas [ ] Seleção de exercícios [ ] Sequenciamento de exercícios [ ] Sumarização de diálogos [ ] Detecção de expressões faciais [ ] Agrupamento de códigos [ ] Outra

Fonte – Elaborada pelo Autor

4.2 Execução

A etapa de execução consistiu na aplicação prática do protocolo de revisão descrito nas seções anteriores. Nesse contexto, as publicações foram identificadas e filtradas. Após

isso, os dados das publicações foram extraídos e analisados, buscando-se responder às questões de pesquisa estabelecidas.

A *string* de busca definida na seção 4.1.2 serviu de base para a atividade de identificação automática de publicações relevantes nas bases de dados das bibliotecas digitais da Springer, ACM e IEEE, bem como para a descoberta de publicações relevantes nos anais do SBIE. O principal objetivo dessa atividade era recuperar o maior número de publicações relevantes para responder às questões de pesquisa, diminuindo o total de falsos positivos (estudos que não se relacionam com o tema pesquisado).

Após isso, os critérios de inclusão e exclusão foram aplicados no processo de filtragem da publicações retornadas. Em uma primeira etapa, foram lidos o título e resumo dos artigos, visando identificar os que atendiam aos critérios de seleção. Em seguida, foram lidas as publicações completas, visando avaliar se os resultados apresentados eram relevantes para responder à questão de pesquisa principal.

Por fim, as informações extraídas foram tabuladas a fim de apresentar as respostas para as questões de pesquisa definidas na subseção 4.1.1. Assim, para cada publicação analisada, buscou-se apresentar o nome do STI, a linguagem de programação ensinada, a abordagem de ensino adotada, a técnica de IA empregada e a sua finalidade.

4.3 Sumarização dos resultados

A Tabela 3 mostra os resultados do procedimento de pesquisa. Embora o processo de busca tenha identificado 50 artigos, um total de 12 (doze) artigos foram eliminados porque eram resumos ou estavam duplicados. Outros 16 (dezesseis) estudos potencialmente relevantes foram excluídos como resultado da aplicação dos critérios de inclusão e exclusão detalhados na subseção 4.1.4. Assim, a amostra resultante do processo de seleção possui 22 publicações.

4.3.1 Quais as linguagens de programação ensinadas?

A revisão mostrou que várias linguagens de programação estão sendo ensinadas por meio de tutoria inteligente. Notou-se a predominância das linguagens de programação Java e Python, presentes em 68% das pesquisas analisadas, muito provavelmente por conta da popularidade alcançada por essas linguagens nos últimos anos. As publicações mencionadas no restante desta seção retratam exemplos das linguagens identificadas.

Em (ALBRECHT; GRABOWSKI, 2016) foi desenvolvido um *framework* para coletar e processar soluções para tarefas de programação escritas na linguagem C. A ferramenta é uma plataforma de mineração de dados que fornece *feedback* imediato sobre o processo da aprendizado dos alunos. Os dados coletados podem fornecer conclusões acerca

Tabela 3 – Quadro comparativo.

ID	NOME DO SISTEMA	REFERÊNCIA	QS01					QS02							QS03										QS04							
			LINGUAGEM DE PROGRAMAÇÃO					ESTRATÉGIA DE ENSINO							TÉCNICA DE INTELIGÊNCIA ARTIFICIAL										FINALIDADE							
			Python	C	C++	Java	Outra	Baseada em exemplos	Baseada em simulação	Baseada em diálogos	Baseada em análise de programas	Baseada em feedback	Baseada em colaboração	Outra	Model tracing	Knowledge tracing	Bayesian Network Model	Neural Network Model	Hidden Markov Model	Kmeans	KNN	Decision Trees/Random Forests	Markov Decision Process	Statistical Models	Outra	Feedback	Dicas	Seleção de exercícios	Sequenciamento de exercícios	Detecção de Expressões Faciais	Detecção de Sentimentos	Agrupamento de códigos	Outra
1	--	[Albrecht e Grabowski 2016]		x								x								x	x					x					x	
2	--	[Barbosa et al. 2018]	x									x								x											x	x
3	--	[Choudhury 2016]	x									x									x			x	x	x					x	
4	--	[Fwa 2019]				x						x						x				x		x							x		
5	--	[Hamid et al. 2015]																	x				x				x	x					
6	ChiQAT	[Harsley et al. 2016]			x			x				x			x								x			x							
7	FITS	[Hooshyar et al. 2015]								x				x		x	x										x	x					
8	TIPOO	[Jiménez et al. 2016]				x				x																				x			x
9	--	[Jin et al. 2014]				x						x												x		x	x						
10	--	[Kumar 2016a]			x	x	x	x				x												x		x							
11	--	[Kumar 2016b]			x	x	x	x				x												x		x							
12	--	[Leong 2015]				x						x			x										x					x		
13	--	[Meliana e Nurjanah 2018]																	x				x				x	x					
14	--	[Swamy et al. 2018]	x									x				x		x							x	x							
15	--	[Tiam-Lee and Sumi 2019]				x						x						x				x			x					x	x		
16	JavaTutor	[Vail et al. 2016]				x				x															x					x	x		
17	Protus 2.1	[Vesin et al. 2016]				x							x		x												x						
18	--	[Wang 2014]										x											x				x	x					
19	--	[Wang 2018]										x											x				x	x					
20	--	[Wang 2019]										x											x				x	x					
21	TiPs	[Xhakaj e Aleven 2018]	x					x	x		x	x														x							
22	--	[Zhang 2013]										x											x				x	x					

Fonte – Elaborada pelo Autor

do comportamento, do estilo de programação, das habilidades de programação e dos erros mais comuns dos alunos.

Em (KUMAR, 2016a) foi analisado o efeito da funcionalidade de ignorar *feedback* em um sistema de tutoria projetado para ensinar instruções de seleção em C++/Java/C#. O sistema usa um pré-teste para preparar o modelo de aluno e adapta os exercícios subsequentes com base nessa avaliação inicial. Toda vez que a resposta do aluno está incorreta, o tutor apresenta uma explicação passo-a-passo da resposta correta.

Em (VAIL et al., 2016) foi sugerido que a observação de traços comportamentais multimodais pode fornecer informações valiosas acerca do aprendizado dos alunos. Em particular, o trabalho examinou a expressão facial do aluno, a atividade eletrodérmica, a postura e o gesto imediatamente após as perguntas feitas por tutores humanos. O JavaTutor é um ambiente para ensinar conceitos introdutórios na linguagem de programaçao Java, em que os alunos avançam nas atividades por meio de diálogos com o tutor.

Em (VESIN; KLAŠNJA-MILIĆEVIĆ; IVANOVIĆ, 2016) foi apresentada uma abordagem para fornecer recomendações personalizadas usando marcação (*tagging*) colabo-

rativa. O Protus 2.1 é um sistema de tutoria preparado para realizar cursos personalizados em vários domínios, mas sua primeira versão é voltada para o aprendizado da linguagem de programação Java. A arquitetura do sistema é composta por dois componentes principais, uma interface de recursos de aprendizado baseada em *tags* e uma ferramenta de recomendação baseada em *tags*.

Em (SWAMY et al., 2018) foram examinados códigos escritos na linguagem Python v3 de alunos do curso introdutório em ciência de dados da Universidade da Califórnia (Berkeley). Para cada aluno, o modelo criado usando a técnica *Deep Knowledge Tracing* (DKT) consegue prever o número de tentativas restantes para a conclusão das perguntas de cada habilidade, assim o tutor pode identificar em menor tempo a necessidade de intervir no processo de aprendizado.

4.3.2 Quais as estratégias de ensino adotadas?

Observou-se a prevalência das abordagens de ensino baseadas em feedback (59%), embora também tenham sido identificados STIs baseados em exemplos (15%), simulação (4%), diálogos (11%), análise de programas (4%) e colaboração (4%). As publicações mencionadas no restante desta seção apresentam algumas das abordagens identificadas.

Em (LEONG, 2015) foi investigada a importância de se modelar a frustração dos alunos, por meio de uma abordagem baseada em *feedback*. Para o autor, modelar a frustração é útil para saber quando intervir no processo de aprendizado do aluno, que, de outra forma, perderia confiança e interesse no assunto. Nesse cenário, registros de comportamentos e acionamentos de teclas são usados para detectar a frustração dos alunos em sessões de programação em Java.

Em (HARSLEY et al., 2016) foi avaliada a eficácia de uma estratégia de ensino baseada em exemplos e analogias por meio da incorporação desses conceitos no sistema ChiQat-Tutor. O estudo mostrou que os alunos que usaram exemplos trabalhados como padrão obtiveram maiores ganhos de aprendizado do que os alunos que não usaram. O estudo mostrou que o avanço por meio das etapas do exemplo não está diretamente relacionado aos ganhos de aprendizagem, mas que a conclusão do exemplo sim.

Em (JIMÉNEZ et al., 2016) foi apresentado um protótipo que usa a abordagem de ensino baseada em diálogos, denominado de TIPOO. Esse sistema possui um mecanismo de bate-papo que possibilita a interação entre o aluno e o tutor. No momento da publicação, haviam sido implementados 5 (cinco) atos de fala: saudação, para desejar boas-vindas aos novos alunos; resposta factual, para responder a uma pergunta técnica dos alunos e outros três tipos de respostas parabenizando, incentivando ou elogiando.

Em (XHAKAJ; ALEVEN, 2018) foi proposta uma abordagem baseada em simulação.

O TiPs suporta dois tipos de atividades orientadas a conceitos: rastreamento de código e compreensão de código. O TiPs tem como alvo tópicos considerados desafiadores no que diz respeito ao ensino de programação, incluindo variáveis, operações de atribuição, instruções condicionais, loops, etc. Esse sistema é capaz de simular a execução do código, permitindo que o aluno acompanhe a evolução do estado interno de um programa linha por linha.

Em (FWA, 2019) foi examinada a possibilidade de se prever a não conclusão de exercícios de programação por meio das teclas digitadas e das ações dos alunos capturadas em tempo real em um tutor inteligente de programação em Java. A abordagem adotada nesse estudo também é baseada em *feedback*. Conhecendo antecipadamente o aluno que não é capaz de concluir um exercício específico, o tutor pode intervir no processo de aprendizado do aluno, fornecendo o auxílio necessário para conclusão do exercício.

4.3.3 Quais as técnicas de inteligência artificial empregadas?

A revisão mostrou que várias técnicas de inteligência artificial podem ser empregadas em ambientes de tutoria para programação de computadores. As mais frequentes foram as abordagens baseadas em redes neurais (9%), as que usam processos de decisão de Markov (22%) e as que usam modelos estatísticos (9%) . As publicações mencionadas no restante desta seção retratam exemplos das técnicas encontradas.

Em (HAMID; ALAIWY; HUSSIEN, 2015) foi abordado o problema de personalização da sequência de aprendizagem utilizando cadeias de Markov em um *framework* de modelos ocultos de Markov, visando explorar o relacionamento de dependência observado entre os conceitos de um curso de programação Java. Os autores descobriram que fatores ocultos, como idade e sexo dos alunos, afetam seu caminho de aprendizagem.

Em (HOOSHYAR et al., 2015) foi proposto um tutor inteligente baseado em fluxogramas, denominado de FITS, para os estágios iniciais do aprendizado de programação. A fim de apoiar programadores iniciantes, a abordagem de rede bayesiana foi aplicada principalmente para a tomada de decisões e para lidar com incertezas quanto ao nível de conhecimento dos alunos. O ambiente usa várias técnicas de IA para apoiar os estudantes, como processamento de linguagem natural, bases de conhecimento e sistema multiagentes.

Em (CHOUDHURY; YIN; FOX, 2016) foi apresentada uma abordagem capaz de fornecer orientação automática aos estudantes mediante a identificação de semelhanças entre submissões de código, análise de semelhanças usando técnicas de clustering (KNN) e comparação de árvores sintáticas. As orientações fornecidas em uma determinada submissão são baseadas nas propriedades dos códigos de outros alunos que possuem estruturas similares, mas são estitilistamente superiores.

Em (BARBOSA; COSTA; BRITO, 2018) foi investigada uma abordagem adaptativa para agrupamento de códigos, a fim de minimizar o esforço despendido na avaliação de códigos em cursos introdutórios de programação. O algoritmo de agrupamento *K-means* foi utilizado para agrupar os conjuntos de códigos. Os resultados variaram de concordância razoável a perfeita, considerando as avaliações semi-automáticas obtidas com o agrupamento e as avaliações de especialistas.

Em (MELIANA; NURJANAH, 2018) foram analisados os exercícios realizados pelos melhores alunos para prever uma sequência apropriada de perguntas para os alunos seguirem. Nesse intuito, aplicaram modelos ocultos de Markov para representar as habilidades dos alunos e Processos de Decisão de Markov para modelar a experiência de exercício de bons alunos. O modelo resultante foi capaz de recomendar sequências dinâmicas de perguntas.

4.3.4 Para quais finalidades as técnicas são aplicadas?

A revisão mostrou que as técnicas de inteligência artificial têm sido empregadas mais frequentemente para fornecimento de sugestões e dicas (29%), reconhecimento de espressões faciais e detecção de sentimentos (18%), seleção e sequenciamento de exercícios (39%) e agrupamento de códigos (8%). As publicações mencionadas no restante desta seção retratam exemplos desses empregos.

Em (JIN et al., 2014) foi descrito um ambiente de tutoria que consiste em dois componentes configuráveis, um de planejamento guiado e um de codificação assistida que oferece dicas geradas automaticamente por tarefa, sob demanda, para os alunos. Os alunos que trabalharam com esse tipo de ambiente obtiveram maiores ganhos de aprendizado pré-teste e pós-teste do que os que trabalharam em ambientes somente de planejamento ou somente de codificação.

Em (KUMAR, 2016b) buscou-se aumentar o engajamento dos estudantes mediante a incorporação de questões que exigem o preenchimento de espaços vazios em sugestões fornecidas por sistemas de tutoria. O autor descobriu que, quando as perguntas eram incorporadas às sugestões, os alunos passavam muito mais tempo por atividade e aprendiam conceitos com significativamente menos problemas de prática.

Em (TIAM-LEE; SUMI, 2019) foram analisadas as emoções dos alunos enquanto escreviam programas de computador sem qualquer interferência humana, para identificar demonstrações de afeto. Para isso, os autores combinaram recursos derivados de registros de digitação, registros de compilação e vídeos do rosto dos alunos durante a resolução de exercícios de codificação.

4.4 Considerações Finais

Este capítulo apresentou uma Revisão Sistemática da Literatura acerca de STIs aplicados ao contexto do ensino de programação, abrangendo estudos publicados nos anos de 2013 a 2019, presentes na base de dados das bibliotecas digitais da Springer, da ACM, do IEEE e nos anais do SBIE.

De início, foram analisadas as 50 publicações resultantes da primeira etapa de filtragem, que teve como base os critérios de inclusão (CI) e exclusão (CE). Após esse processo, uma segunda etapa de filtragem foi realizada, o que culminou na eliminação de 28 publicações, resultando em um total de 22 artigos.

A revisão mostrou que várias linguagens de programação estão sendo ensinadas por meio de tutoria inteligente. Contudo, notou-se uma predominância das linguagens de programação Java e Python, presentes em 68% das pesquisas analisadas, muito provavelmente por conta da popularidade alcançada por essas linguagens nos últimos anos.

Além disso, ainda se observou a prevalência das abordagens de ensino baseadas em feedback (59%), embora também tenham sido identificados TIPs baseados em exemplos (15%), simulação (4%), diálogos (11%), análise de programas (4%) e colaboração (4%).

Também se identificou que várias técnicas de inteligência artificial podem ser empregadas em ambientes de tutoria para programação de computadores. As mais frequentes foram as abordagens baseadas em redes neurais (9%), as que usam processos de decisão de Markov (22%) e as que usam modelos estatísticos (9%) .

Por fim, no que diz respeito à finalidade das técnicas empregadas, a maior concentração dos estudos foca no fornecimento de sugestões e dicas (29%), reconhecimento de espressões faciais e detecção de sentimentos (18%), seleção e sequenciamento de exercícios (39%) e agrupamento de códigos (8%).

5 Modelagem do Problema

Comumente, no ensino introdutório de algoritmos, os professores incentivam a resolução de problemas computacionais do mundo real. Para tanto, um conjunto de atividades precisa ser selecionado e ordenado de forma apropriada, levando em consideração o conceito abordado e o nível de complexidade das questões. O esforço exigido por essa tarefa, todavia, pode torná-la impraticável, não sobrando tempo ao professor para realizar um acompanhamento individualizado dos alunos. Neste capítulo, apresenta-se um modelo baseado em Processos de Decisão de Markov Parcialmente Observáveis (POMDP) capaz de simplificar a tarefa de ordenação das questões e discutem-se os resultados da etapa metodológica de modelagem do problema, descrita na seção 3.2,

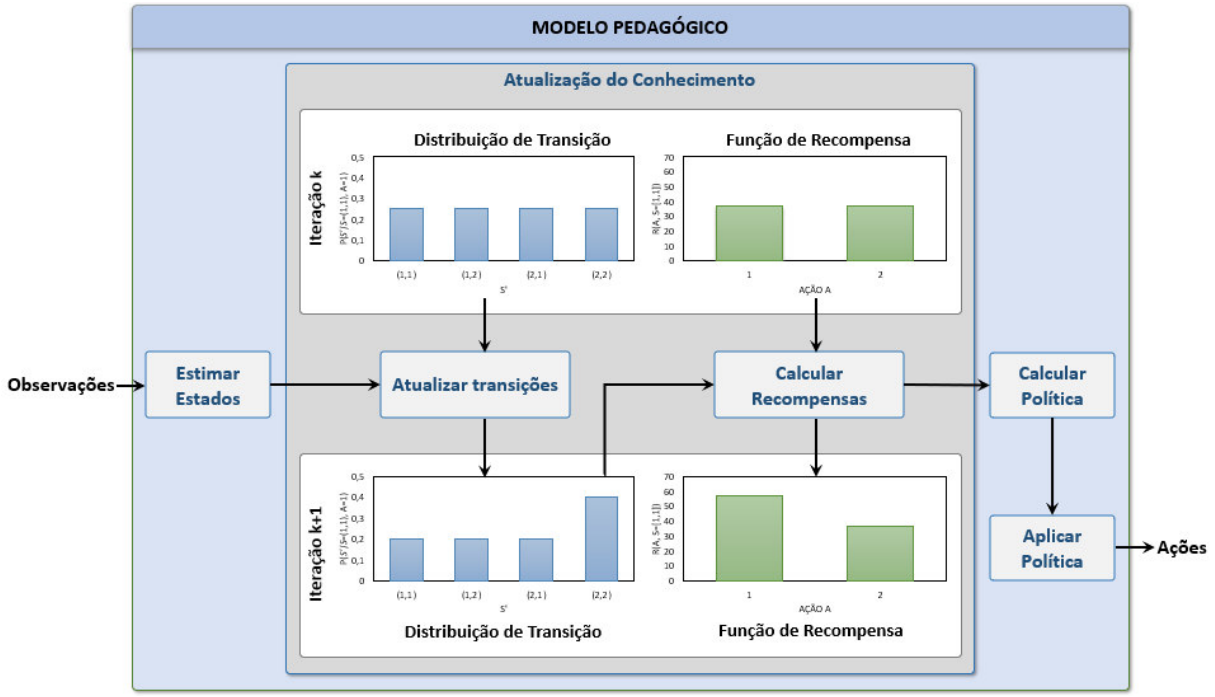
5.1 Definição do Cenário de Ensino e Aprendizagem

O cenário pedagógico que se deseja modelar situa-se no domínio de conhecimento de introdução aos algoritmos. O contexto em que ocorre o processo de ensino e aprendizagem envolve um professor (tutor humano), que é responsável por preparar e expor um conteúdo em sala de aula, para que os estudantes possam assimilar. A cada encontro presencial, o educador incentiva a resolução de problemas computacionais por meio de um ambiente virtual de aprendizagem. Nesse ambiente, a ordem dos problemas é adaptada com base em estratégias de ensino voltadas a promover o desenvolvimento dos estudantes de maneira individualizada.

5.2 Definição da Estrutura Interna do Modelo Pedagógico

Em um Sistema Tutor Inteligente (STI), o modelo pedagógico é o componente responsável por determinar as estratégias pedagógicas mais apropriadas para orientar o processo de aprendizagem de um determinado aluno. Basicamente, esse modelo é responsável por processar um conjunto de observações e tomar decisões quanto às ações de ensino. Para tanto, os seguintes passos devem ser realizados: estimar o estado de conhecimento do estudante; atualizar as probabilidades de transição; calcular as recompensas; calcular e aplicar as novas políticas, conforme descrito na (Figura 13).

Figura 13 – Funcionamento do Modelo Pedagógico.



Fonte – Elaborada pelo Autor.

No presente tópico, detalham-se os elementos que compõem o modelo para tomada de decisões pedagógicas proposto neste trabalho. Trata-se de um Processo de Decisão de Markov Parcialmente Observável (POMDP), $\mathcal{P} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O}, \gamma, b_0 \rangle$, focado especificamente no processo de ensino de algoritmos, em que um tutor inteligente precisa sequenciar um conjunto de problemas computacionais, $P = \{P_1, P_2, P_3, \cdots, P_n\}$, de acordo com as interações dos alunos.

5.2.1 Espaço de Estados

Para que possa escolher a estratégia de ensino mais adequada, o tutor deve ser capaz de estimar o nível de conhecimento acumulado pelo aluno até uma determinada interação. O estado de conhecimento do estudante reflete exatamente essa informação, sendo responsável por determinar quais problemas o aluno já é capaz de solucionar e quais ainda não.

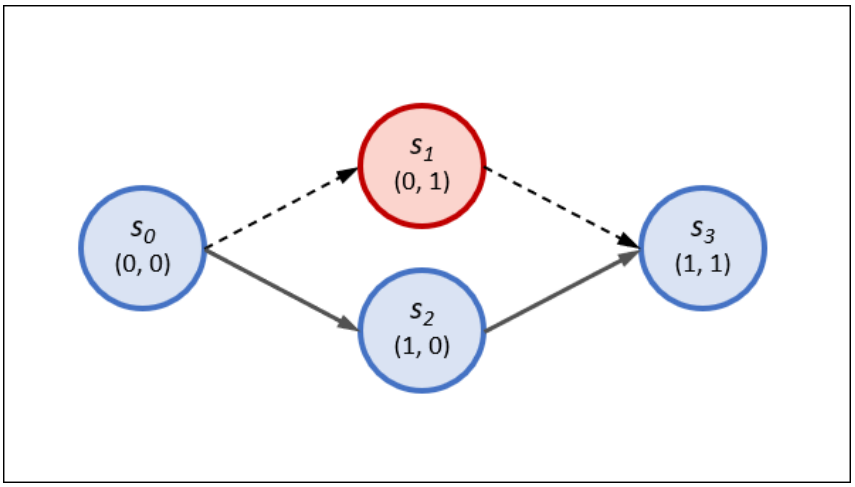
Os estados são codificados por meio de vetores binários $s = \{s_{P_1}, s_{P_2}, s_{P_3}, \cdots, s_{P_n}\}$, em que cada componente s_{P_i} pode assumir os valores 0 ou 1, para $1 \leq i \leq n$, em que o valor 0 denota que o aluno ainda não é capaz de solucionar um problema, enquanto que o valor 1 indica o contrário.

Desse modo, o número de possíveis codificações de estado é exponencial, 2^n. Para

lidar com o crescimento exponencial do espaço de estados, pode-se utilizar relações de interdependência entre os problemas, como forma de eliminar estados inválidos e manter um espaço de estados de tamanho gerenciável.

Por exemplo, se o problema P_1 for pré-requisito de P_2, então, obrigatoriamente, o único estado de conhecimento possível, em que $s_{P_2} = 1$, é $(1, 1)$. Nesse caso, o estado $s_1 = (0, 1)$ não é válido, pois viola a relação de dependência entre os problemas.

Figura 14 – Estados de conhecimento e suas transições (para apenas dois problemas).



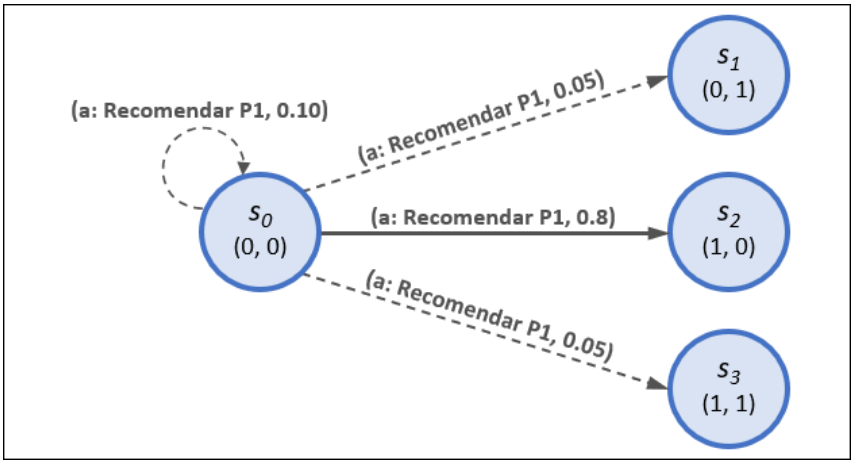
Fonte – Elaborada pelo autor.

5.2.2 Espaço de Ações

As ações representam as possibilidades de escolha que o tutor pode fazer. As ações indicam as recomendações de problemas computacionais realizadas pelo tutor durante o processo de ensino. Assim, deverá haver uma ação de recomendar, a_{q_i}, para cada questão $q_i \in Q$ existente na base de problemas.

Por exemplo, considerando o estado inicial como $s_0 = (0, 0)$, e supondo a existência de apenas dois problemas, P_1 e P_2, a Figura 15 retrata a distribuição de probabilidade com relação ao estado de destino para a ação de recomendar o problema P_1. Nesse exemplo, tem-se uma maior probabilidade de se atingir s_2 por meio da recomendação de P_1, mas, ainda assim, há uma pequena margem de probabilidade de que s_1 e s_3 sejam alcançados.

Figura 15 – Distribuição de probabilidade da ação de recomendar o problema P_1.



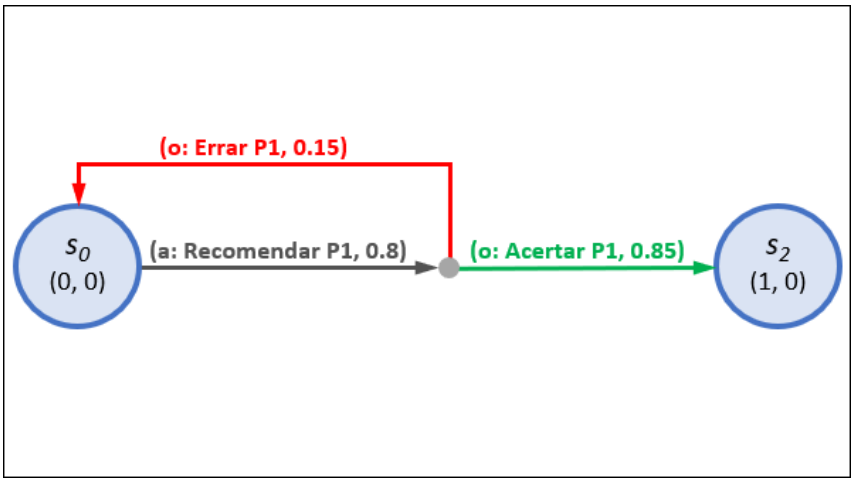
Fonte – Elaborada pelo autor.

5.2.3 Espaço de Observações

As observações resultam do confronto entre soluções submetidas pelos alunos e um conjunto de casos de teste. Como consequência disso, o tutor poderá observar ou uma resposta correta, o_1, ou uma resposta incorreta, o_2.

Tomando como base o exemplo mostrado na Figura 16, partindo-se do estado $(0, 0)$, e sendo recomendado o problema P_1, tem-se a probabilidade de se observar uma solução correta de 85%; e uma solução incorreta de 15%, conforme arestas nas cores verde e vermelho, respectivamente.

Figura 16 – Distribuição de probabilidade das observações para a ação de recomendar o problema P_1.



Fonte: Elaborada pelo autor.

5.2.4 Função de transição

Quando uma ação a_{q_i} é capturada em um estado $s \in \mathcal{S}$, existe uma probabilidade de transição para um novo estado s' que é expressa pela Equação 5.1, em que t indica um momento no tempo e s' é o novo estado no tempo $t+1$:

$$\mathcal{T}\Big(s' \mid s, a\Big) = P\Big(s_{t+1} = s' \mid s_t = s, a_t = a\Big) \tag{5.1}$$

Para computar as probabilidades iniciais, deve-se capturar as sequências de ações do sistema, juntamente com os estados em que as ações são executadas, as observações e os novos estados:

$$\cdots, \Big(s_t, a_t, o_t, s_{t+1}\Big), \Big(s_{t+1}, a_{t+1}, o_{t+1}, s_{t+2}\Big), \cdots \tag{5.2}$$

Assim, a função de transição ($\mathcal{T}$) será calculada pela seguinte expressão:

$$\mathcal{T}\Big(s'|s, a\Big) = \frac{\#\text{ transições de } s \text{ para } s' \text{ quando } a \text{ é tomada em } s}{\#\ a \text{ é tomada em } s} \tag{5.3}$$

em que # simboliza o número vezes que o evento ocorreu.

5.2.5 Função de Observação

Quando um estado s' é causado por uma ação a_{q_i}, existe uma probabilidade de se capturar uma observação o que é expressa pela Equação 5.4, em que t indica um momento no tempo e s' é o novo estado no tempo $t+1$:

$$\mathcal{O}\Big(o \mid s', a_{q_i}\Big) = P\Big(o_{t+1} = o \mid s_{t+1} = s', a_t = a_{q_i}\Big) \tag{5.4}$$

Assim, a função de observação ($\mathcal{O}$) é calculada pela seguinte expressão:

$$\mathcal{O}\Big(o \mid s', a\Big) = \frac{\#\ a \text{ é tomada, } s' \text{ é alcançado e } o \text{ é observado}}{\#\ a \text{ é tomada e } s' \text{ é alcançado}} \tag{5.5}$$

5.2.6 Função de Recompensa

A função de recompensa ($\mathcal{R}$) define os ganhos alcançados pelo tutor durante o processo de recomendação de problemas computacionais. O tutor conquista recompensas maiores quando um estudante é capaz de solucionar uma maior quantidade de problemas em um horizonte de planejamento mais curto.

No presente trabalho, considera-se que será distribuída uma recompensa de +100 quando estado de conhecimento do aluno transitar para um estado válido. Por outro lado, quando o aluno submete uma resposta incorreta, fazendo com que o estado permaneça inalterado, o tutor recebe uma penalização no valor de −50. Tais valores foram definidos de forma experimental, tendo como base a premissa de que o cometimento de um erro não implica na completa perda do conhecimento já adquirido por um aluno.

A função de recompensa será definida conforme a Equação 5.6:

$$\mathcal{R}\Big(s' \mid s, a_{q_i}\Big) = \begin{cases} 0 & \text{se} \quad s = s' \\ 0 & \text{se} \quad s \text{ é alcançável pela ação } a \\ +100 & \text{se} \quad s' \text{ é alcançável pela ação } a \\ -50 & \text{se} \quad s' \text{ não é alcançável pela ação } a \end{cases} \tag{5.6}$$

5.2.7 Fator de desconto

Para fins de análise do modelo proposto, o fator de desconto (γ) é estipulado em 0.95, considerando um horizonte infinito de ações de ensino. Esse fator é responsável por determinar que recompensas futuras sejam descontadas de acordo com o momento em que forem recebidas.

5.2.8 Estado de Crença Inicial

O estado de crença inicial é estabelecido sob a suposição de que aluno não possui incialmente nenhuma capacidade para resolver os problemas recomendados pelo tutor, isto é, a probabilidade inicial do estado s_0 é 1 e dos demais estados é 0.

5.3 Criação de Instâncias do Modelo Pedagógico

Essa etapa é responsável por construir as instâncias do modelo pedagógico que são utilizadas na simulação, para diferentes números de problemas computacionais. Essa atividade exige que alguns desafios sejam enfrentados e superados.

O principal desafio diz respeito ao crescimento exponencial das funções de transição e observação. Como forma de suplantar esse desafio, faz-se a criação dos modelos de forma parametrizada. Para tanto, as probabilidades de transição e observação são definidas por meio de regras uniformes para todas as instâncias do modelo pedagógico.

Assim, a função de transição é definida conforme a Equação 5.7, em que a probabilidade de transição assume: 1.0, se o próximo estado é identico ao atual e alcançável por meio da ação selecionada; 0.3, se o próximo estado for idêntico ao estado atual e não for

alcançável por meio da ação selecionada; 0.7, se o próximo estado for diferente do atual e válido; 0.0, se o próximo estado for diferente do atual e inválido:

$$\mathcal{T}\Big(s' \mid s, a\Big) = \begin{cases} 1.0 & \text{se} \quad s = s' \quad \text{e} \quad s \text{ é alcançável pela ação } a \\ 0.3 & \text{se} \quad s = s' \quad \text{e} \quad s \text{ não é alcançável pela ação } a \\ 0.7 & \text{se} \quad s \neq s' \quad \text{e} \quad s' \text{ é alcançável pela ação } a \\ 0.0 & \text{se} \quad s \neq s' \quad \text{e} \quad s' \text{ não é alcançável pela ação } a \end{cases} \tag{5.7}$$

A função de observação assume a forma expressa na Equação 5.8, em que a probabilidade de observação assume: $0,85$ se observada uma resposta correta e o próximo estado é alcançável pela ação a; $0,15$ se observada uma resposta correta e o próximo estado não é alcançável por meio da ação a; $0,15$ se observada uma resposta correta e o próximo estado não é alcançável por meio da ação a; e $0,85$ se observada uma resposta errada e o próximo estado não é alcançável por meio da ação a:

$$\mathcal{O}\Big(o \mid s', a_{q_i}\Big) = \begin{cases} 0.85 & \text{se} \quad o = \textit{certa} \quad \text{e} \quad s' \text{ é alcançável pela ação } a \\ 0.15 & \text{se} \quad o = \textit{errada} \quad \text{e} \quad s' \text{ é alcançável pela ação } a \\ 0.15 & \text{se} \quad o = \textit{certa} \quad \text{e} \quad s' \text{ não é alcançável pela ação } a \\ 0.85 & \text{se} \quad o = \textit{errada} \quad \text{e} \quad s' \text{ não é alcançável pela ação } a \end{cases} \tag{5.8}$$

Outro desafio igualmente importante diz respeito ao crescimento exponencial do modelo em decorrência de uma variação linear no número de problemas. Isso faz com que o esforço necessário para a codificação do modelo também cresça de forma exponencial, conforme o aumento do número de problemas.

Por isso, foi necessário o desenvolvimento de uma ferramenta para, com base em um conjunto de parâmetros, automatizar a criação de instâncias do modelo pedagógico. Essa ferramenta foi codificada na linguagem Julia, usando o pacote JuliaPOMDP, que possui módulos para definição, instanciação e execução de modelos MDP e POMDP.

Com a ferramenta já desenvolvida, deu-se início à criação das instâncias. Ao todo, foram criadas 5 (cinco) instâncias, seguindo os parâmetros definidos anteriormente, para representar conjuntos com 3, 4, 5, 6 e 7 problemas computacionais. Conforme se verifica na Tabela 4, as regras das funções de transição, observação e recompensa crescem de forma exponencial, bem como o número de linhas de código e o tamanho dos modelos em quilobytes.

Tabela 4 – Resumo das Instâncias.

Instância	**Número de Problemas**	**Número de Estados**	**Número de Ações**	**Número de Observações**	**Regras da Função de Transição**	**Regras da Função de Observação**	**Regras da Função de Recompensa**	**Linhas de Código**	**Tamanho do arquivo (kB)**
1	3	8	3	2	192	384	192	1.027	29
2	4	16	4	2	1024	2048	1024	4.619	134
3	5	32	5	2	5.120	10.240	5.120	21.619	644
4	6	64	6	2	24.576	49.152	24.576	100.860	3.056
5	7	128	7	2	114.688	229.376	114.688	464.517	14.244

Fonte – Elaborada pelo Autor.

As instâncias desenvolvidas nesta atividade são utilizadas como entradas do processo de simulação.

6 Simulação

Nesse estágio, ocorre a execução das instâncias criadas na etapa anterior. Antes, contudo, deve-se definir os algoritmos de planejamento utilizados no processo de simulação. Como resultado desta etapa, tem-se o registro de um conjunto de dados muito significativo, relativo à recompensa descontada média e ao tempo de execução dos algoritmos escolhidos, viabilizando a comparação deles com uma política aleatória, para cada instância do modelo Nesse capítulo, apresentam-se os resultados da etapa metodológica de simulação, descrita na seção 3.3.

6.1 Detalhamento dos Algoritmos de Planejamento

Os algoritmos de planejamento empregados neste trabalho foram escolhidos dentre os disponíveis na ferramenta JuliaPOMDP. Ela oferece suporte a um conjunto expressivo de algoritmos para a resolução de POMDPs, dentre os quais, pode-se citar o QMDP, FIB, SARSOP, POMCP e POMCPOW. Esses, inclusive, foram os algoritmos selecionados para a resolução das instâncias criadas anteriormente, pois dipõem de documentação ampla e suporte da comunidade acadêmica. Nas subseções seguintes, apresentam-se as principais características dos algoritmos escolhidos.

6.1.1 *Q-Value Markov Decision Process* (QMDP)

Uma das técnicas de aproximação mais simples é o algoritmo QMDP. Esse método assume que o sistema pode ser interpretado como completamente observável no espaço de crenças logo após o primeiro passo, já que a cada instante de tempo t, a observação o_t, a crença atual $b_t(s)$ e a ação tomada a_t determinam univocamente a próxima crença, $b_{t+1}(s)$ (KOCHENDERFER, 2015). Dessa forma, cada POMDP pode ser tratado como um MDP formalmente equivalente, no qual o conjunto de estados corresponde ao conjunto de crenças do POMDP original.

Um único vetor alfa é construído para cada ação usando a iteração de valor. Cada vetor alfa é inicializado com 0, $\alpha_a^1 = 0$, para cada valor $a \in \mathcal{A}$ e então iterado:

$$\alpha_a^{(k+1)}(s) \longleftarrow \mathcal{R}(s,a) + \gamma \sum_{s'} \mathcal{T}(s,a,s') \times \max_{a'} \alpha_{a'}^{(k)}(s) \tag{6.1}$$

Cada iteração requer $O(|\mathcal{A}|^2|\mathcal{S}|^2)$ operações. Como $k \to \infty$, os vetores alfa resultantes aproximam-se da função de valor ótimo de acordo com $\max_a \alpha_a^\top b$. A política resultante

executa a ação que maximiza essa função para a crença atual. O algoritmo QMDP, quando executado para convergência, produz um limite superior para a função de valor verdadeira.

A hipótese de observabilidade pode fazer com que o QMDP não aproxime tão bem o valor das ações de coleta de informação, que são ações que reduzem significativamente a incerteza no estado. Além disso, o QMDP pode ter um bom desempenho em problemas em que a política ótima não exige uma coleta muito onerosa de informações.

6.1.2 *Fast Informed Bound* (FIB)

O limite superior calculado pelo QMDP não leva em consideração a observabilidade parcial do ambiente. Em particular, as ações de coleta de informações que podem ajudar a identificar o estado atual sempre se encontram abaixo do nível ótimo de acordo com esse limite (ROSS et al., 2008). Para resolver este problema, em (HAUSKRECHT, 2000) é proposto um novo método para calcular limites superiores, denominado *Fast Informed Bound* (FIB), que é capaz de levar em consideração (até certo ponto) a observabilidade parcial do ambiente.

Assim como o QMDP, o FIB calcula um único vetor alfa para cada ação. Conforme (KOCHENDERFER, 2015), o processo de atualização do vetor alfa é realizado da seguinte forma:

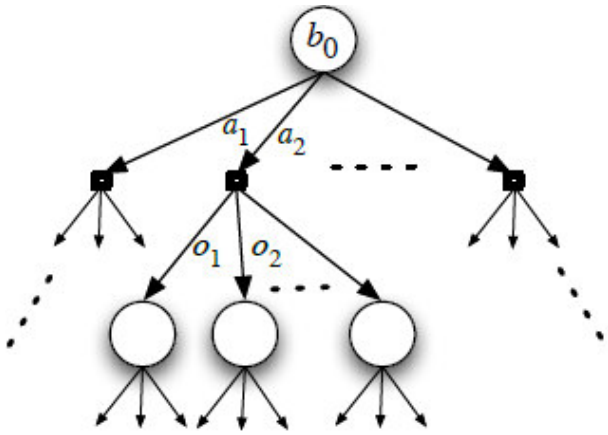
$$\alpha_a^{(k+1)}(s) \longleftarrow \mathcal{R}(s,a) + \gamma \sum_o \max_{a'} \sum_{s'} \mathcal{O}(o,a,s') \times \mathcal{T}(s,a,s') \times \alpha_{a'}^{(k)}(s') \tag{6.2}$$

Cada iteração requer $O(|\mathcal{A}|^2|\mathcal{S}|^2|\Omega|)$ operações, somente um fator de $|\Omega|$ a maios que o QMDP. Em todos os estados de crença, o algoritmo fornece um limite superior na função de valor ideal que é sempre inferior ao limite do QMDP (KOCHENDERFER, 2015).

6.1.3 *Successive Approximations of the Reachable Space under Optimal Policies* (SARSOP)

Consiste em um algoritmo de planejamento baseado em pontos que utiliza árvores como estrutura de dados para representar o conjunto de pontos de crença, sendo a raiz da árvore a crença inicial.

Figura 17 – Árvore de Crenças $T_{\mathcal{R}}$.



Fonte: (KURNIAWATI; HSU; LEE, 2008).

Algoritmos baseados em pontos têm sido particularmente bem-sucedidos na computação de soluções aproximadas para POMDPs. A maioria deles usa iteração de valor. Explorando o fato de que a função de valor ótimo deve satisfazer a equação de Bellman, os algoritmos de iteração de valor começam com uma política inicial representada como uma função de valor V e realizam operações de *backup* em V iterando na equação de Bellman até convergir (KURNIAWATI; HSU; LEE, 2008).

Uma característica compartilhada pelos algoritmos baseados em pontos é amostrar um conjunto representativo de pontos do espaço de crença $\mathcal{B}$ e calcular uma função de valor aproximadamente ótima, executando operações de *backup* sobre os pontos amostrados, em vez de todo o espaço $\mathcal{B}$. Basicamente, esses algoritmos diferem em como amostram o espaço de crença e como executam as operações de *backup*.

Como todos os algoritmos baseados em pontos, o SARSOP coleta amostras de um conjunto de pontos do espaço de crenças. Os pontos amostrados formam uma árvore $T_{\mathcal{R}}$ (Figura 18). A raiz de $T_{\mathcal{R}}$ é o ponto de crença inicial b_0. Para amostrar um novo ponto b_0, escolhe-se um nó b de $T_{\mathcal{R}}$, bem como uma ação $a \in \mathcal{A}$ e uma observação $o \in$ de acordo com distribuições de probabilidade ou heurísticas adequadas (KURNIAWATI; HSU; LEE, 2008). Em seguida, calcula-se b_0 usando a fórmula:

$$\begin{aligned} b'\big(s'\big) &= \tau\big(b, a, o\big) \\ &= \eta \times \mathcal{O}\big(s', a, o\big) \times \sum_{s \in \mathcal{S}} \mathcal{T}\big(s, a, s'\big) b\big(s\big) \end{aligned} \tag{6.3}$$

em que η é uma constante de normalização. Todo ponto b' é inserido em $T_{\mathcal{R}}$ como filho de b. Cada ponto amostrado desta forma é alcançavel a partir de b_0.

Para conseguir isso, o algoritmo mantém um limite inferior $\underline{V}$ e um limite superior $\overline{V}$ na função de valor ótimo V^*. O conjunto Γ de vetores alfa representa uma aproximação

linear por partes para V^*, e também serve como limite inferior quando inicializado adequadamente. Para o limite superior $\overline{V}$, o SARSOP usa a aproximação dente de serra, mas pode ser inicializado de outras maneiras, a exemplo do algoritmo *Fast Informed Bound* (HAUSKRECHT, 2000).

Para amostrar novos pontos de crença, o algoritmo define um tamanho de intervalo entre os limites superiores e inferiores e segue caminhando do nó raiz para os nós filhos, no sentido descendente da árvore. Em cada nó, uma ação é escolhida com o limite superior mais alto enquanto a observação escolhida é aquela que faz a maior contribuição para o espaço no nó raiz da árvore. O algoritmo termina quando o intervalo dos limites é menor que um dado limiar ou quando um limite de tempo pré-especificado é atingido.

6.1.4 *Partially Observable Monte-Carlo Planning* (POMCP)

O POMCP é um algoritmo de planejamento online que combina uma busca em árvore de Monte-Carlo (do inglês, *Monte-Carlo Tree Search* (MCTS)) com atualizações de Monte-Carlo sobre estado de crença do agente. Ele constrói dinamicamente uma árvore de busca, em que cada nó armazena um valor correspondente a uma estimativa. A amostragem de Monte-Carlo é usada para quebrar a maldição da dimensionalidade (do inglês, *curse of dimensionality*) durante as atualizações do estado de crença e as etapas de planejamento. (SILVER; VENESS, 2010).

O agente usa um simulador $\mathcal{G}$ como um modelo generativo do POMDP. O simulador fornece uma amostra do estado sucessor, observação e recompensa, $\Big(s_{t+1}, o_{t+1}, r_{t+1}\Big) \sim \mathcal{G}\Big(s, a\Big)$. O simulador é usado para gerar sequências de estados, observações e recompensas. Essas simulações são usadas para atualizar a função de valor para o estado corrente, sem nunca olhar para dentro da caixa preta que descreve a dinâmica do modelo (SILVER; VENESS, 2010).

A função de valor é estimada pela média dos retornos de N execuções a partir do estado de crenças b, sendo $V(b,a) = \frac{1}{N}\sum_{i=1}^{N} R^i$, em que R^i é o valor retornado da i-ésima simulação. Toda ação $a \in \mathcal{A}$ é avaliada e aquela que contém o maior valor $V(b,a)$ é selecionada. Na prática, centenas de milhares de simulações podem ser executadas em menos de um segundo, permitindo que sejam construídas árvores de buscas extensas (SILVER; VENESS, 2010).

A busca em árvore de Monte-Carlo (MCTS) é uma técnica de simulação de Monte-Carlo para problemas de árvore de busca. A cada iteração desse algoritmo, um novo nó é escolhido para ser expandido e ter uma simulação executada a partir de um de seus filhos. A estratégia de seleção, portanto, é responsável por balancear a diversificação (exploration) e a intensificação (exploitation) da busca. Para balancear a escolha entre diversificação e intensificação, o POMCP faz uso do algoritmo UCT (do inglês, *Upper Confidence bounds*

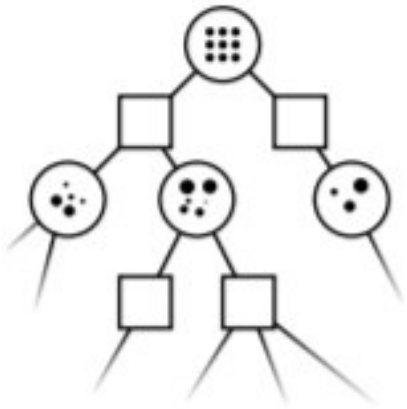
applied to Trees) (KOCSIS; SZEPESVÁRI, 2006).

Esse algoritmo pode ser aplicado em problemas que são muito largos ou muito complexos para serem representados por meio de uma definição explícita de distribuições de probabilidades. O diferencial desse algoritmo é a capacidade de superar as maldições de dimensionalidade e de história, mediante a amostragem de transições de estados, em vez de considerar todas as transições de estados possíveis.

6.1.5 *Partially Observable Monte Carlo Planning with Observation Widening* (POMCPOW)

Esse algoritmo é baseado no POMCP. Agora, contudo, as atualizações de crenças são ponderadas, mas também se expandem gradualmente conforme mais simulações são adicionadas. Além disso, como a riqueza da representação de crenças está relacionada ao número de vezes que o nó é visitado, as crenças com maior probabilidade de serem alcançadas pela política ótima têm mais partículas. Em cada etapa, o estado simulado é inserido na coleção de partículas ponderadas que representa a crença, e um novo estado é amostrado a partir dessa crença (SUNBERG; KOCHENDERFER, 2018).

Figura 18 – Árvore de Crenças do POMCPOW



Fonte: Sunberg e Kochenderfer (2018).

6.2 Execução das Instâncias

Nesse estágio, as instâncias criadas no capítulo anterior são resolvidas por algoritmos de planejamento específicos (QMDP, FIB, SARSOP, POMCP e POMCPOW) e, também, por uma política aleatória, seguindo parâmetros idênticos, de forma a permitir a comparação dos resultados.

Assim, os algoritmos específicos e a política aleatória são executados para horizontes de tamanhos variáveis (a saber: 1, 2, 3, 4, 5, 10, 20, 50, 100, 200, 500, 1000) e, para cada horizonte de planejamento, são coletadas 10 (dez) amostras.

Os dados brutos registrados para as instâncias 1, 2, 3, 4 e 5 são apresentados, respectivamente, nos Apêndices B, C, D, E, F, por meio de tabelas que detalham a recompensa descontada e o tempo de execução por horizonte de planejamento e amostra.

As Tabelas 5, 6, 7, 8 e 9 mostram os valores médios das recompensas descontadas e dos tempos de execução, por horizonte e instância, dos algoritmos QMDP, FIB, SARSOP, POMCP e POMCPOW. Por sua vez, a Tabela 10 apresenta os resultados da política aleatória.

Tabela 5 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo QMDP.

Horizonte	**Recompensa Descontada Média**					**Tempo Médio de Execução**				
	Inst. 1	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**	**Inst. 1**	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**
1	90,00	70,00	60,00	80,00	90,00	131	502	5582	221280	13880939
2	145,50	117,00	156,00	146,00	165,00	132	502	5582	221279	13690039
3	210,65	179,20	207,23	228,20	248,18	131	502	5582	221279	13658987
4	233,00	234,95	223,59	241,64	278,67	131	502	5582	221280	13778285
5	254,97	343,71	301,30	319,39	364,79	131	502	5582	221279	13660199
10	273,72	317,67	423,08	469,23	542,76	131	502	5582	221280	13929143
20	262,82	353,63	429,08	497,27	556,03	131	502	5582	221282	13535627
50	277,47	344,19	429,28	488,05	564,44	132	502	5593	221283	13571691
100	276,59	357,43	411,97	492,77	570,49	132	505	5583	221290	13638082
200	272,04	348,82	421,36	511,20	589,79	132	502	5586	221301	13935590
500	265,81	358,08	412,19	498,04	567,08	132	505	5598	221360	13681235
1000	271,53	356,01	436,80	481,50	547,14	133	510	5601	221369	13586586

Fonte – Elaborada pelo autor.

Tabela 6 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo FIB.

Horizonte	**Recompensa Descontada Média**					**Tempo Médio de Execução (em ms)**				
	Inst. 1	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**	**Inst. 1**	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**
1	50,00	90,00	40,00	80,00	90,00	320	445	3420	55805	1233483
2	156,50	146,50	106,50	117,00	135,00	315	435	3465	54705	1232823
3	199,18	162,13	218,70	172,60	196,42	315	415	3605	55135	1233081
4	213,07	280,17	268,32	251,54	289,20	315	435	3555	55285	1233171
5	251,21	305,74	369,58	297,06	343,06	315	420	3635	55175	1233105
10	267,37	348,25	402,76	433,80	503,32	315	420	3645	55175	1233105
20	265,48	347,76	430,85	477,87	547,96	315	410	3445	54820	1232892
50	269,18	344,46	414,18	487,57	559,46	315	415	3455	55360	1233216
100	274,98	348,15	417,65	484,86	545,41	315	410	3540	54635	1232781
200	267,60	359,68	429,07	485,65	562,40	315	415	3690	55785	1233471
500	269,25	353,39	407,17	490,34	575,44	315	435	3600	54770	1232862
1000	276,32	349,22	424,16	486,54	559,14	320	430	3615	55480	1233288

Fonte – Elaborada pelo autor.

Tabela 7 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo SARSOP.

Horizonte	**Recompensa Descontada Média**					**Tempo Médio de Execução (em ms)**				
	Inst. 1	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**	**Inst. 1**	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**
1	70,00	50,00	60,00	90,00	100,00	1046	1902	6840	215641	9929229
2	98,00	127,00	146,50	156,00	175,00	1045	1967	6841	215638	9871934
3	217,70	180,18	237,23	199,68	231,45	1046	1985	6835	215630	9828151
4	236,80	268,22	258,72	248,31	288,75	1047	1949	6835	215657	10008842
5	263,57	307,71	362,33	327,70	373,70	1056	1937	6833	215656	9921020
10	270,24	344,62	427,06	452,73	514,92	1054	1955	6865	215645	9870458
20	273,41	341,76	412,15	476,11	548,33	1050	1937	6846	215643	9894266
50	274,18	336,22	427,78	499,48	574,16	1046	1895	6840	215652	9707599
100	272,80	342,29	419,85	488,36	558,12	1043	1883	6836	215640	9895356
200	268,75	353,75	410,97	481,44	551,51	1051	1897	6853	215645	10024698
500	277,59	359,26	415,20	482,11	560,13	1054	1896	6856	215729	9834991
1000	276,66	355,75	423,92	489,54	563,63	1054	1894	6857	215704	9911220

Fonte – Elaborada pelo autor.

Tabela 8 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo POMCP.

Horizonte	**Recompensa Descontada Média**					**Tempo Médio de Execução (em ms)**				
	Inst. 1	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**	**Inst. 1**	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**
1	70,00	60,00	40,00	50,00	60,00	204	805	6475	218373	10586826
2	157,00	127,00	137,00	165,50	175,00	204	804	6486	218377	10290967
3	209,18	210,65	189,20	198,23	227,25	202	808	6480	218381	10456032
4	185,45	250,19	214,04	288,24	334,36	203	808	6494	218386	10561095
5	244,25	255,65	325,33	304,40	359,15	206	809	6497	218400	10621760
10	282,15	329,39	380,40	480,01	555,97	212	817	6499	218421	10603698
20	276,10	357,06	405,09	468,46	534,45	220	835	6522	218459	10696238
50	272,70	339,23	434,12	497,76	572,21	259	881	6594	218598	10679764
100	263,74	349,87	414,04	496,27	563,55	292	965	6718	218815	10611677
200	270,78	349,65	415,61	489,34	558,08	397	1107	6960	219268	10730815
500	276,71	339,34	399,40	491,41	567,03	651	1520	7704	220688	10535859
1000	269,11	352,53	422,18	474,70	548,49	1107	2301	8884	223134	10704214

Fonte – Elaborada pelo autor.

Tabela 9 – Recompensa Descontada Média e Tempo Médio de Execução do Algoritmo POMCPOW.

Horizonte	**Recompensa Descontada Média**					**Tempo Médio de Execução (em ms)**				
	Inst. 1	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**	**Inst. 1**	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**
1	70,00	70,00	70,00	70,00	80,00	302	747	5938	247472	13783141
2	127,50	127,50	126,50	156,00	185,00	311	738	5965	247510	13643542
3	171,65	161,65	161,15	209,68	246,75	314	749	5959	247519	13802654
4	195,52	215,52	270,69	216,40	269,25	317	752	5970	247535	13736740
5	235,32	277,38	318,66	285,17	329,37	321	756	5980	247565	13787399
10	265,57	344,04	354,66	429,59	494,95	339	796	6032	247669	13437316
20	251,65	328,34	392,44	472,46	545,39	374	841	6135	247894	13589795
50	258,90	315,27	401,54	477,25	553,47	481	994	6448	248513	13804508
100	254,26	344,91	412,04	474,80	546,89	668	1253	6636	249540	13640205
200	265,37	342,94	405,31	486,27	552,27	1045	1651	7312	251615	13943614
500	271,37	333,69	397,97	480,61	546,74	2136	2916	9364	258401	14331316
1000	269,39	352,92	417,67	477,38	547,61	3998	5201	12841	269303	14893245

Fonte – Elaborada pelo autor.

Tabela 10 – Recompensa Descontada Média e Tempo Médio de Execução da Política Aleatória.

Horizonte	**Recompensa Descontada Média**					**Tempo Médio de Execução (em ms)**				
	Inst. 1	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**	**Inst. 1**	**Inst. 2**	**Inst. 3**	**Inst. 4**	**Inst. 5**
1	50,00	60,00	60,00	60,00	70,00	140	599	8243	245430	8762647
2	97,50	137,00	166,50	117,00	135,50	140	599	8243	245420	8762643
3	117,53	191,63	200,65	182,13	208,18	140	599	8243	245420	8762643
4	169,75	187,95	214,54	212,12	241,46	140	599	8243	245420	8762644
5	222,78	212,33	238,65	292,32	323,50	140	599	8243	245421	8762646
10	239,46	292,90	316,52	348,65	397,55	140	599	8243	245420	8762645
20	266,48	327,88	363,05	390,40	447,99	140	599	8243	245420	8762648
50	262,05	309,34	377,29	424,43	481,55	140	599	8243	245420	8762637
100	251,22	324,18	390,85	449,69	507,61	140	599	8243	245420	8762638
200	263,04	326,96	387,14	441,01	505,57	140	599	8243	245425	8762645
500	263,26	304,19	357,39	422,70	489,05	140	599	8243	245425	8762648
1000	260,22	330,53	392,22	431,97	498,23	140	599	8244	245427	8762649

Fonte – Elaborada pelo autor.

As tabelas apresentadas neste capítulo servem de insumo para a análise realizada no Capítulo 7, em que se apresenta, de forma gráfica, e sob diferentes óticas, o comportamento dos algoritmos de planejamento e da política aleatória.

7 Análise e Discussão dos Resultados

De posse dos dados registrados durante a etapa de simulação, foram geradas ilustrações para representar o comportamento de cada algoritmo (POMCP, POMCPOW, QMDP, SARSOP e FIB) com relação à obtenção de **recompensas descontadas médias** e aos **tempos de execução médios**, para cada uma das instâncias criadas no Capítulo 6. Neste capítulo, apresentam-se os resultados da etapa metodológica 3.4.

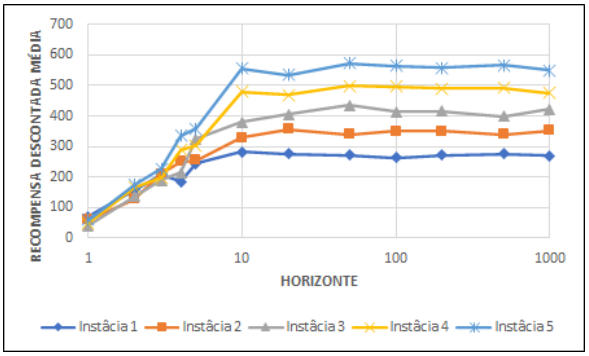
7.1 Comparação dos Resultados por Instância

Nesse sentido, as Figuras 19a, 20a, 21a, 22a e 23a ilustram o desempenho dos algoritmos no que diz respeito à obtenção de recompensas descontadas médias. Pode-se observar que todos os algoritmos, inclusive a política aleatória, têm um comportamento similar, isto é, o aumento do número de problemas faz com que o algoritmo alcance recompensas maiores.

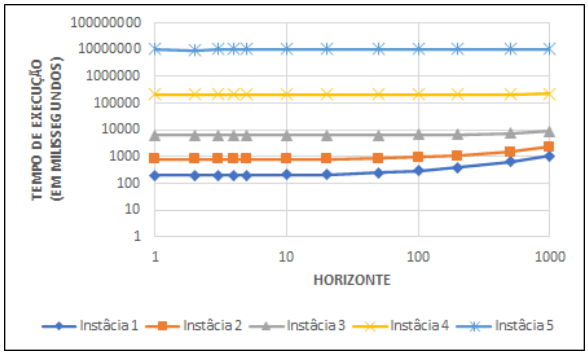
Por sua vez, as Figuras 19b, 20b, 21b, 22b e 23b espelham o comportamento com relação aos tempos de execução. Nota-se que os tempos médios de execução crescem de forma exponencial com o aumento da complexidade das instâncias, isto é, quanto maior o número de problemas contidos na instância, maior o tempo para resolução do modelo.

Figura 19 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo POMCP.

(a) Recompensa descontada média



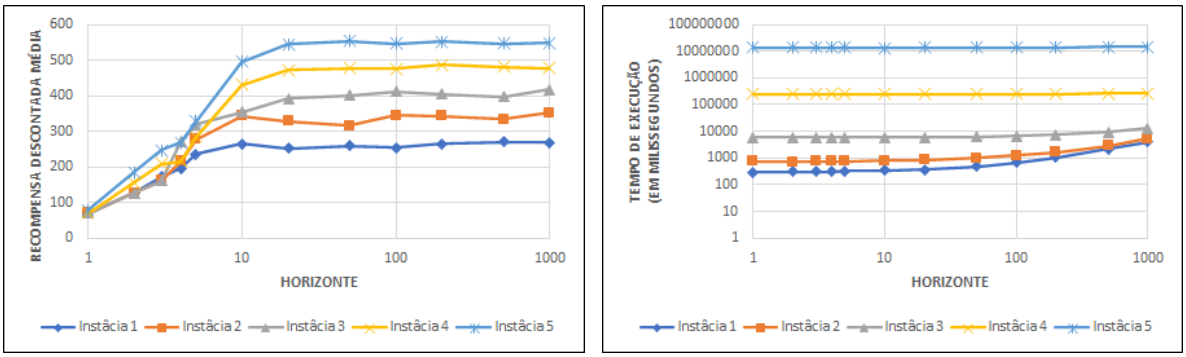
(b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 20 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo POMCPOW.

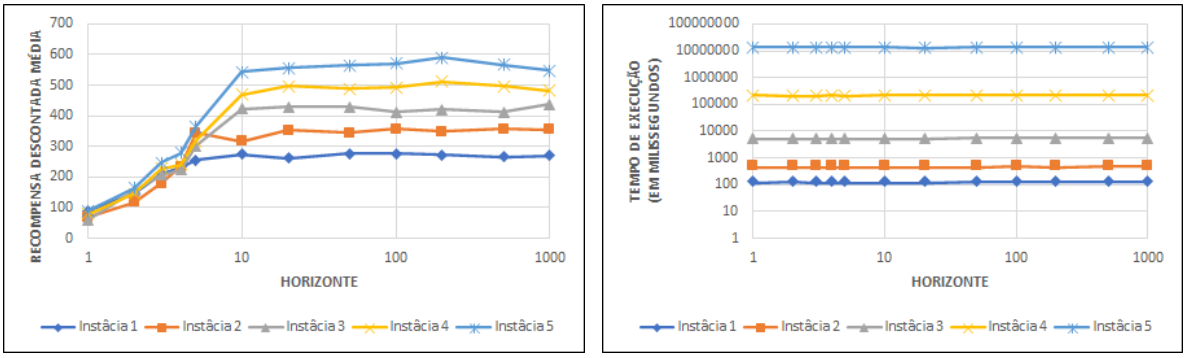
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 21 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo QMDP.

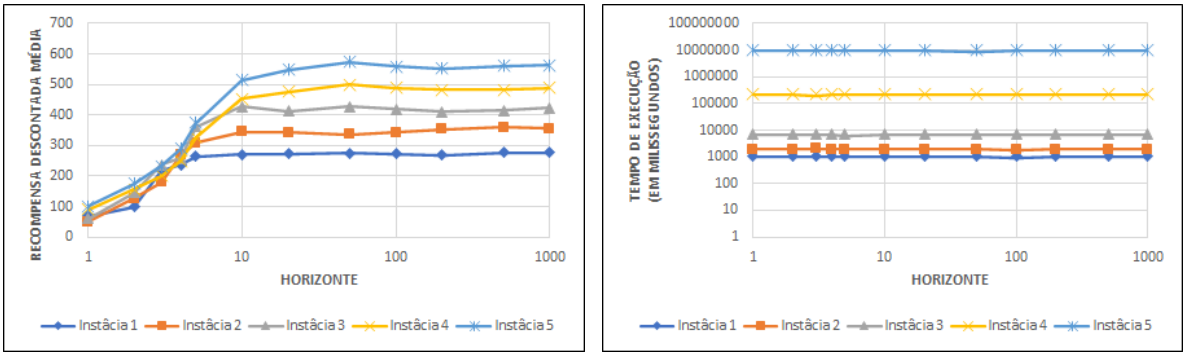
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 22 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo QMDP.

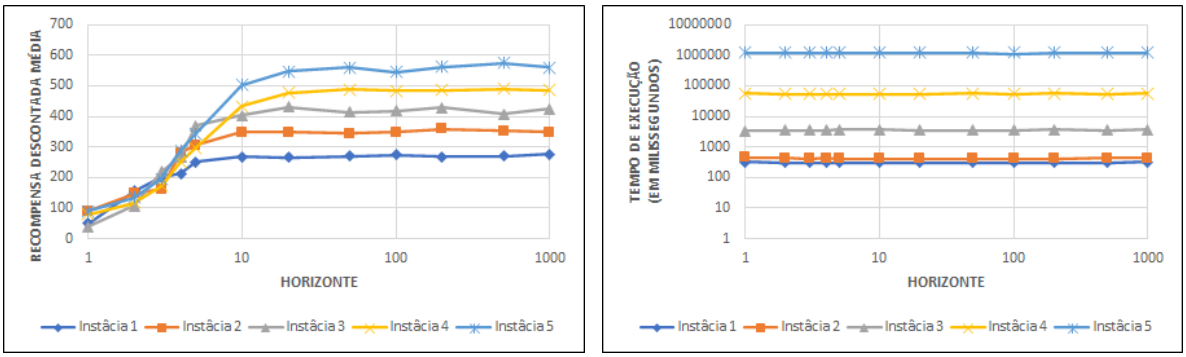
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 23 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio do algoritmo FIB.

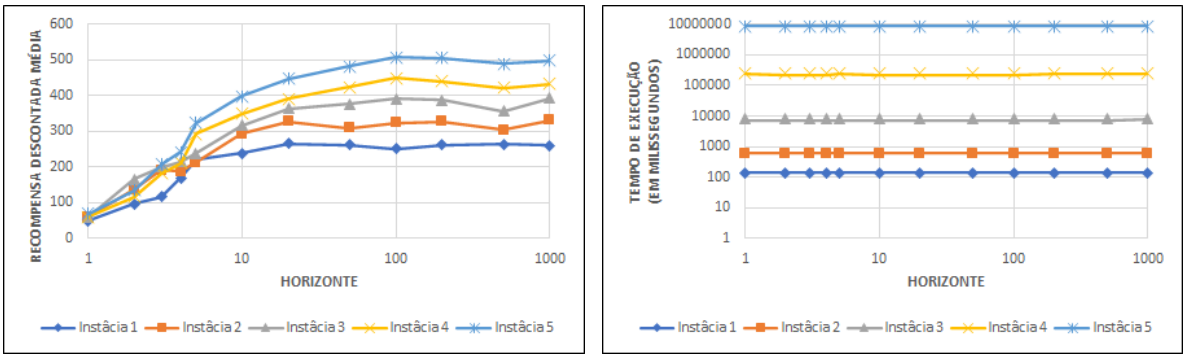
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte: Elaborada pelo autor.

Figura 24 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Política Aleatória.

(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

7.2 Comparação dos Resultados por Algoritmo

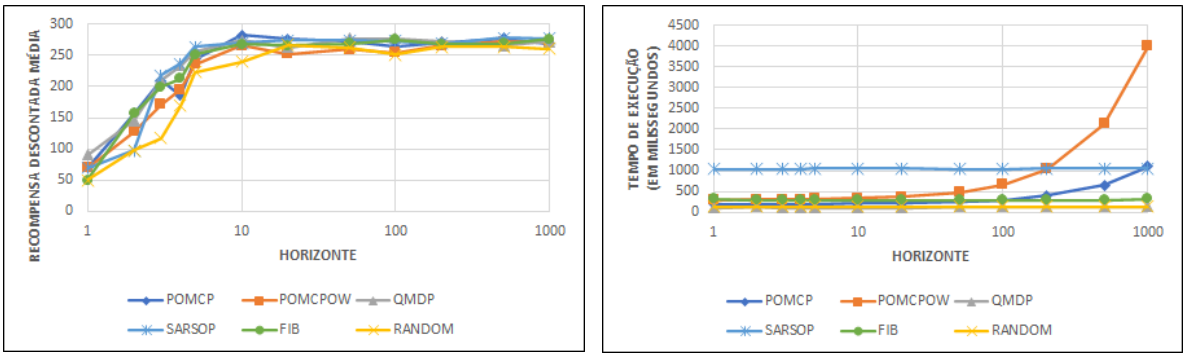
Para analisar o comportamento de cada algoritmo, fez-se também a comparação dos resultados obtidos por instância. Dessa forma, as Figuras 25a, 26a, 27a, 28a e 29a ilustram o desempenho dos algoritmos no que diz respeito à obtenção de recompensas descontadas médias. Já as Figuras 25b, 26b, 27b, 28b e 29b espelham o comportamento com relação aos tempos de execução.

Os gráficos permitem inferir que o comportamento dos algoritmos, no que concerne ao ganho de recompensas descontadas médias, é praticamente idêntico quando se lida com instâncias pequenas (poucas questões). Porém, com o acréscimo de mais questões, verifica-se que a política aleatória passa a apresentar um comportamento inferior aos demais algoritmos.

No que diz respeito ao tempo de execução, o algoritmo *Fast Informed Bound* (FIB) apresenta resultados melhores que os demais algoritmos para as cinco instâncias analisadas. Além disso, os algoritmos POMCP e POMCPOW denotam um crescimento exponencial no tempo médio de execução com relação ao horizonte de planejamento, enquanto o tempo dos demais algoritmos se mantém praticamente constante, independente do horizonte de planejamento considerado.

Figura 25 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 1.

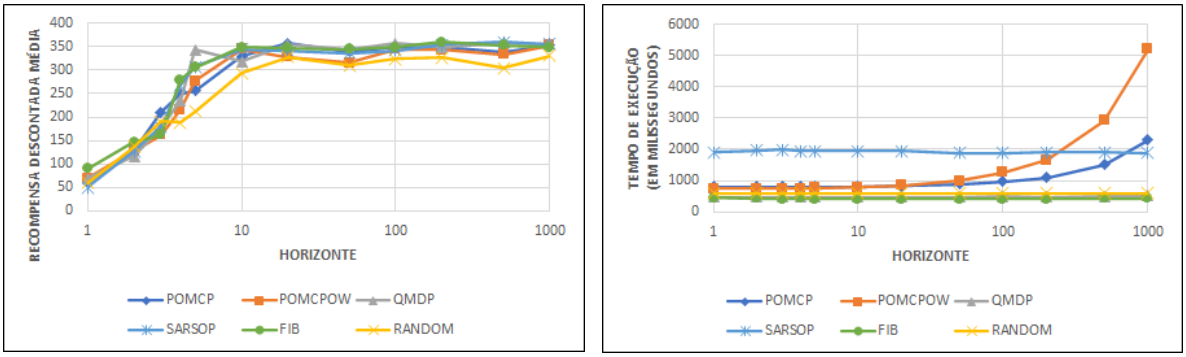
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 26 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 2.

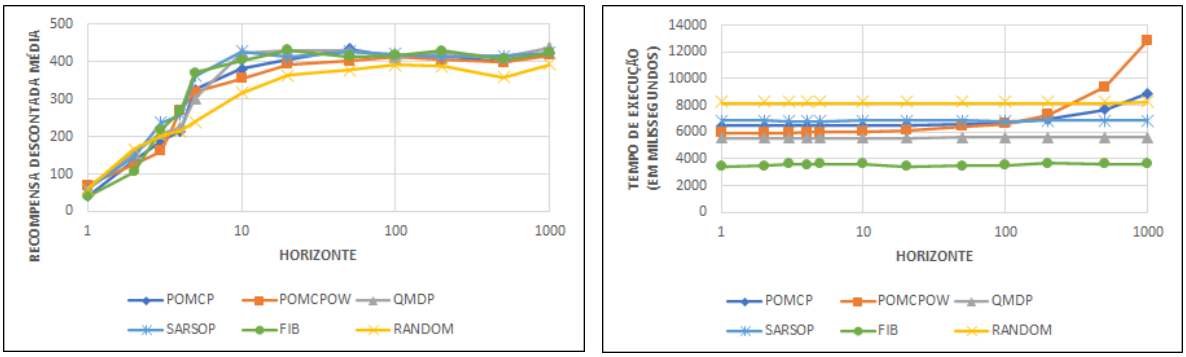
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 27 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 3.

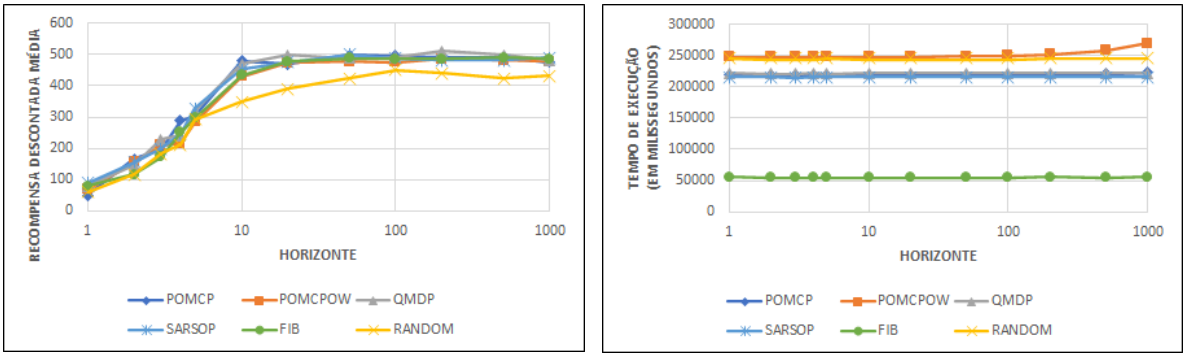
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 28 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 4.

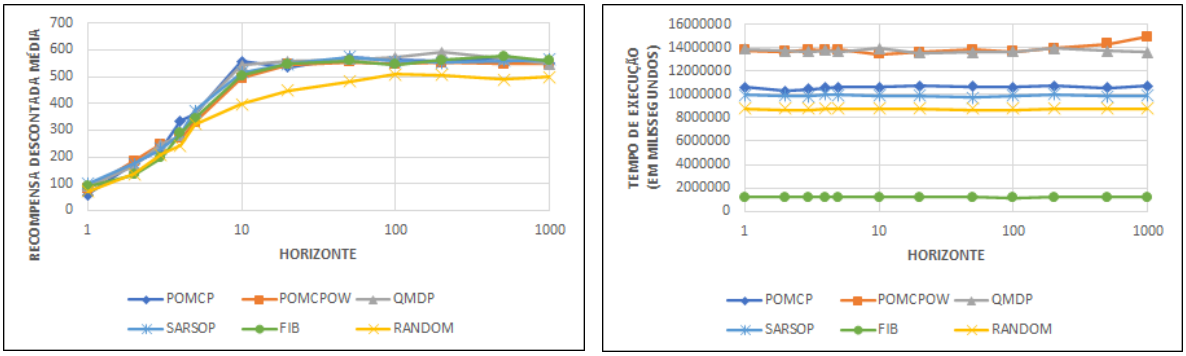
(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

Figura 29 – Gráficos da Recompensa Descontada Média e do Tempo de Execução Médio da Instância 5.

(a) Recompensa descontada média (b) Tempo de execução médio



Fonte – Elaborada pelo autor.

7.3 Análise Estatística dos Resultados

Nessa etapa, investiga-se a contribuição dos algoritmos específicos para o aumento da recompensa descontada média, ou seja, quer-se verificar se a recompensa descontada média **antes** da aplicação de um algoritmo específico é **menor** do que a média **após** sua aplicação. Assim, supõe-se que, se os algoritmos específicos são eficazes, suas recompensas finais são, em média, maiores que as de uma política aleatória.

Trata-se, portanto, de uma situação em que se quer comparar as médias de duas **distribuições normais** (conforme demonstrado no Apêndice F, por meio do teste de normalidade de Kolmogorov-Smirnov) **de uma mesma população**, mas em dois momentos distintos: antes e depois a aplicação de algoritmos específicos. Tem-se, então, uma situação em que é exigida uma decisão, configurando um problema clássico de teste de hipóteses.

Para o cenário em análise, pode-se aplicar o teste de diferenças entre médias populacionais (teste t pareado). Tem-se, então, a comparação entre as médias de duas amostras dependentes $X_1, \cdots, X_n$ e $Y_1, \cdots, Y_n$. Neste caso, consideram-se observações pareadas, isto é, tem-se na realidade uma amostra de pares $(X_1, Y_1), \cdots, (X_n, Y_n)$ (BUSSAB; MORETTIN, 2010).

Nessa situação, a comparação é realizada por meio da análise das diferenças, D_i, entre os valores de cada par. Assim, obtém-se uma amostra das diferenças, $D_1, \cdots, D_n$, em que $D_i = X_i - Y_i$, para $i = 1, 2, \cdots, n$. Embora as amostras sejam dependentes, pode-se admitir que D_i segue uma distribuição normal, isto é, $D_i \sim N(\mu_D, \sigma_D^2)$ (BUSSAB; MORETTIN, 2010).

Sendo assim, busca-se avaliar as seguintes hipóteses, mediante o teste t pareado:

$$\begin{cases} H_0 : \mu_D = 0 \text{ (a recompensa descontada média não sofre influência)} \\ H_1 : \mu_D < 0 \text{ (a recompensa descontada média sofre influência)} \end{cases} \tag{7.1}$$

O parâmetro populacional μ_D é estimado pela média amostral das diferenças, ou seja, $\overline{D}$. Já o parâmetro σ_D^2 é estimado pela variância amostral das diferenças, isto é:

$$s_D^2 = \frac{\sum_{i=1}^{n}(D_i - \overline{D})}{n-1} \tag{7.2}$$

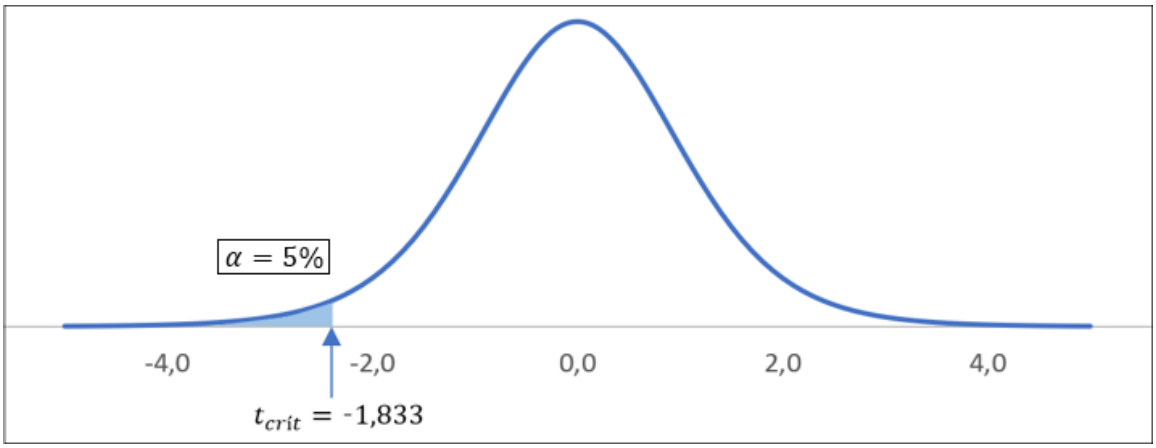
O estatística de teste, t_{teste}, será obtida pela expressão:

$$t_{teste} = \frac{\overline{D} - \mu_D}{\frac{s_D}{\sqrt{n}}} \tag{7.3}$$

que sob a hipótese nula, H_0, segue uma distribuição t de Student com $n-1$ graus de liberdade. Dessa forma, basta substituir a média por μ_D e o desvio padrão amostral por s_D.

Com isso, tem-se que, a um nível de significância $\alpha = 5\%$, o ponto crítico é determinado por $-t_{crít}$ para o unilateral à esquerda. Se $t_{teste} < -t_{crít}$, rejeita-se H_0. Caso contrário, não se pode rejeitar H_0.

Figura 30 – Gráfico da distribuição t de Student para 9 Graus de Liberdade.



Fonte – Elaborada pelo autor.

O p-value será dado por:

$$p\text{-}value = \mathbb{P}[t < t_{teste} | H_0] \tag{7.4}$$

Os resultados dos testes de hipóteses realizados para os dados obtidos na etapa de simulação das instâncias 1, 2, 3, 4 e 5 estão resumidos nas Tabelas 11, 12, 13, 14 e 15, respectivamente. Note-se que, para cada horizonte de planejamento considerado, as tabelas trazem as diferenças das médias, D_i; a média das diferenças, $\overline{D}$; o desvio padrão amostral, s_d; a estatística de teste; t_{teste}, o p-*value* da estatística de teste; e a decisão de rejeitar ou não a hipótese H_0 a um nível de significância de 5%.

Tabela 11 – Testes de Hipóteses para a Instância 1.

Amostra	**Horizonte**											
	1	**2**	**3**	**4**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
D_1	-100,00	-118,00	-129,15	79,71	-79,32	-111,94	2,17	-0,15	13,16	-16,64	-26,99	-16,28
D_2	-60,00	-23,00	-18,90	-146,59	-96,48	-13,16	39,58	-16,96	-26,46	9,00	10,61	-25,23
D_3	20,00	-195,00	-87,25	51,49	-73,39	-11,18	-4,61	-25,00	4,04	-24,96	-3,38	-30,98
D_4	20,00	-37,00	-115,20	116,25	29,96	-17,55	-22,14	19,88	-6,51	-4,48	-26,17	-2,88
D_5	-100,00	-37,00	-167,20	-74,39	-46,83	-18,94	-4,16	9,90	-34,44	-17,26	9,31	-10,95
D_6	40,00	97,00	-95,15	-117,00	6,26	-13,36	13,78	-37,82	7,59	14,88	-1,13	-3,83
D_7	-60,00	-56,00	25,80	6,70	-63,03	-11,88	16,24	21,43	-45,87	-10,80	0,15	-11,16
D_8	60,00	-41,00	-113,05	-245,35	50,01	-95,76	-17,86	-37,55	-48,73	-19,47	-7,01	21,49
D_9	60,00	-62,00	-90,20	8,51	29,64	-11,26	7,42	-12,66	-24,15	3,48	-20,08	-16,10
D_{10}	-80,00	78,00	-51,15	-109,50	-27,64	-18,49	-24,48	-5,45	-11,15	7,55	-24,15	-27,89
$\overline{D}$	-20,00	-39,40	-84,15	-43,02	-27,08	-32,35	0,59	-8,44	-17,25	-5,87	-8,88	-12,38
s_D	66,00	84,23	56,10	114,03	52,60	37,99	19,79	21,56	22,07	13,91	14,42	15,26
t_{teste}	-0,96	-1,48	-4,74	-1,19	-1,63	-2,69	0,09	-1,24	-2,47	-1,33	-1,95	-2,57
p**-value**	0,18	0,09	0,00	0,13	0,07	0,01	0,54	0,12	0,02	0,11	0,04	0,02
Rejeitar H_0	Não	Não	Sim	Não	Não	Sim	Não	Não	Sim	Não	Sim	Sim

Fonte – Elaborada pelo autor.

Tabela 12 – Testes de Hipóteses para a Instância 2.

Amostra	**Horizonte**											
	1	**2**	**3**	**4**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
D_1	40,00	38,00	43,85	22,75	44,16	25,10	-19,19	-61,15	-33,43	0,35	-57,27	-17,08
D_1	-60,00	58,00	14,10	-77,80	-120,98	-14,94	10,92	-5,75	-39,78	-28,08	-61,38	-42,60
D_1	20,00	-95,00	90,15	-80,89	-104,16	-50,62	-57,79	-74,29	-22,08	-58,70	-15,81	-0,21
D_1	-80,00	-1,00	-90,15	-58,71	-226,61	-159,27	2,54	-29,79	-21,26	-5,50	-35,03	9,73
D_1	40,00	-16,00	39,10	-38,85	-210,46	-2,41	-32,18	-12,40	-4,15	1,00	-74,96	-58,92
D_1	-100,00	76,00	23,85	-136,05	-50,22	-80,91	-1,70	-19,04	-41,30	-27,76	-26,15	6,71
D_1	80,00	1,00	-53,20	-34,05	-44,03	-80,88	-81,89	35,04	21,24	-64,42	-39,58	-29,66
D_1	20,00	116,00	4,80	-43,85	-12,02	-22,40	-5,55	-17,56	-3,76	-17,34	-24,85	-55,96
D_1	-60,00	-157,00	76,10	-123,64	-25,97	-74,30	14,87	-43,81	-8,08	-36,52	-95,60	-11,69
D_1	20,00	60,00	-19,95	-47,54	-106,77	21,74	-8,27	-36,62	-90,92	-3,14	-14,98	-27,84
$\overline{D}$	-8,00	8,00	12,87	-61,86	-85,71	-43,89	-17,82	-26,54	-24,35	-24,01	-44,56	-22,75
s_D	61,25	82,28	55,64	45,87	85,83	56,87	31,13	30,64	30,24	23,72	26,91	24,64
t_{teste}	-0,41	0,31	0,73	-4,26	-3,16	-2,44	-1,81	-2,74	-2,55	-3,20	-5,24	-2,92
p**-value**	0,34	0,62	0,76	0,00	0,01	0,02	0,05	0,01	0,02	0,01	0,00	0,01
Rejeitar H_0	Não	Não	Não	Sim	Sim	Sim	Não	Sim	Sim	Sim	Sim	Sim

Fonte – Elaborada pelo autor.

Tabela 13 – Testes de Hipóteses para a Instância 3.

Amostra	**Horizonte**											
	1	**2**	**3**	**4**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
D_1	-20,00	39,00	-117,00	-56,34	-133,70	-179,25	-26,62	-47,90	44,02	-7,45	-34,36	-44,40
D_2	20,00	60,00	-17,95	-3,75	-201,36	-141,12	4,04	4,65	-44,68	-80,45	-9,51	-7,51
D_3	-80,00	-37,00	-16,00	1,00	-116,41	-56,80	-88,96	-48,37	-31,68	-35,99	-53,04	19,29
D_4	60,00	58,00	-34,20	17,25	44,24	-99,27	-27,17	-59,38	-71,50	-0,43	-69,34	-28,25
D_5	100,00	78,00	74,10	-79,00	-83,47	40,18	-19,33	-23,04	-8,82	-10,29	-63,80	-14,87
D_6	-40,00	97,00	6,80	7,61	-214,70	-5,68	12,40	-21,81	23,02	-69,31	-61,87	-49,34
D_7	40,00	39,00	5,80	21,00	-40,42	-95,21	-144,26	-60,55	-30,58	29,67	-31,57	-83,95
D_8	20,00	38,00	4,85	-92,49	-120,11	-94,85	-149,21	-0,26	-36,51	-76,08	-58,09	8,76
D_9	20,00	23,00	79,00	-85,59	-21,52	-100,31	-76,89	-122,63	-82,00	-25,45	-67,77	-50,22
D_{10}	-60,00	-75,00	-5,90	-54,95	-80,46	-78,39	7,29	-61,61	-3,88	-17,40	-40,62	-76,78
$\overline{D}$	6,00	32,00	-2,05	-32,53	-96,79	-81,07	-50,87	-44,09	-24,26	-29,32	-49,00	-32,73
s_D	55,82	51,88	55,04	45,40	79,09	62,64	60,62	36,89	39,13	36,14	19,43	34,45
t_{teste}	0,34	1,95	-0,12	-2,27	-3,87	-4,09	-2,65	-3,78	-1,96	-2,57	-7,97	-3,00
p-**value**	0,63	0,96	0,45	0,02	0,00	0,00	0,01	0,00	0,04	0,02	0,00	0,01
Rejeitar H_0	Não	Não	Não	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim

Fonte – Elaborada pelo autor.

Tabela 14 – Testes de Hipóteses para a Instância 4.

Amostra	**Horizonte**											
	1	**2**	**3**	**4**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
D_1	-60,00	-41,00	78,05	-111,15	80,48	-205,55	-151,64	-79,05	-57,17	-99,38	-73,99	-28,98
D_2	-80,00	58,00	-5,90	49,39	-126,14	84,54	-99,78	-36,65	-46,14	-90,04	-47,71	-17,56
D_3	-60,00	-175,00	93,10	-172,91	-6,80	-17,71	0,84	-28,49	-29,62	-55,64	-49,36	-51,48
D_4	40,00	-35,00	77,05	-195,86	-36,86	-88,90	-53,49	-89,58	-71,94	-17,12	-80,11	-76,88
D_5	60,00	59,00	-35,15	184,50	-12,11	-88,55	-107,72	-75,88	-17,86	6,41	-53,86	-72,86
D_6	20,00	-136,00	-108,20	94,10	-17,09	-188,44	-78,40	-22,82	-69,60	-2,25	-97,18	-77,74
D_7	0,00	39,00	-53,10	-86,54	81,47	-206,84	-111,27	-47,91	-78,43	-141,49	-63,10	-73,93
D_8	20,00	-62,00	-127,25	16,25	-128,91	-108,78	-152,42	-100,66	-2,12	-51,13	-67,92	-48,99
D_9	-80,00	38,00	95,10	-93,44	-55,67	-180,82	19,83	-84,87	3,06	30,89	-34,09	3,89
D_{10}	0,00	-56,00	-209,20	-55,44	77,36	-43,13	-146,33	-89,99	-7,45	-77,92	-90,70	-55,12
$\overline{D}$	-14,00	-31,10	-19,55	-37,11	-14,43	-104,42	-88,04	-65,59	-37,73	-49,77	-65,80	-49,97
s_D	51,68	81,07	106,29	120,98	77,81	94,93	60,99	28,71	30,91	54,07	20,10	27,92
t_{teste}	-0,86	-1,21	-0,58	-0,97	-0,59	-3,48	-4,57	-7,23	-3,86	-2,91	-10,35	-5,66
p-**value**	0,21	0,13	0,29	0,18	0,29	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Rejeitar H_0	Não	Não	Não	Não	Não	Sim	Sim	Sim	Sim	Sim	Sim	Sim

Fonte – Elaborada pelo autor.

Tabela 15 – Testes de Hipóteses para a Instância 5.

Amostra	**Horizonte**											
	1	**2**	**3**	**4**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
D_1	-80,00	-60,00	-33,46	-132,96	105,09	-231,11	-170,30	-80,77	-32,44	-105,66	-102,82	-32,13
D_2	-80,00	-80,00	-9,85	38,32	-138,46	86,82	-126,87	-69,34	-34,50	-89,55	-49,98	-37,20
D_3	-80,00	-175,00	-34,80	-193,02	8,79	-19,60	20,32	-34,00	-86,58	-54,93	-46,32	-51,01
D_4	20,00	20,00	25,62	-225,50	-42,99	-85,78	-43,52	-110,93	-89,68	-10,08	-87,15	-83,70
D_5	40,00	20,00	-24,05	158,20	3,42	-113,21	-122,79	-78,76	-38,49	-36,09	-77,24	-59,70
D_6	20,00	20,00	-32,81	52,59	-5,89	-232,62	-109,75	-50,21	-70,03	5,75	-95,69	-102,05
D_7	0,00	-100,00	70,65	-110,88	19,61	-225,09	-101,74	-59,85	-56,27	-146,97	-76,62	-69,07
D_8	0,00	20,00	-27,82	79,65	-167,00	-145,43	-174,57	-114,32	-54,29	-46,84	-79,53	-51,88
D_9	20,00	-20,00	-55,84	-135,86	-60,32	-223,28	24,04	-109,81	-22,46	26,86	-30,75	14,94
D_{10}	0,00	40,00	-95,96	-36,40	-27,39	-59,05	-179,21	-123,98	-8,08	-114,85	-96,27	-77,94
$\overline{D}$	-14,00	-31,50	-21,83	-50,59	-30,51	-124,84	-98,44	-83,20	-49,28	-57,24	-74,24	-54,97
s_D	47,19	70,16	44,80	128,18	78,56	107,98	75,38	30,52	27,00	56,36	24,11	32,56
t_{teste}	-0,94	-1,42	-1,54	-1,25	-1,23	-3,66	-4,13	-8,62	-5,77	-3,21	-9,74	-5,34
p-**value**	0,19	0,09	0,08	0,12	0,13	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Rejeitar H_0	Não	Não	Não	Não	Não	Sim	Sim	Sim	Sim	Sim	Sim	Sim

Fonte – Elaborada pelo autor.

Os resultados dos testes de hipóteses demonstram que para as instâncias 1, 2 e 3, que contêm, respectivamente, 3, 4 e 5 problemas, os algoritmos de planejamento conseguem se sobressair, em relação à política aleatória, a partir do horizonte de planejamento de 4 etapas. Por sua vez, para as instâncias 4 e 5, que possuem 6 e 7 problemas, os resultados indicam que os algoritmos específicos são melhores que uma política aleatória a partir do horizonte de tamanho 10.

Tabela 16 – Resumo dos Testes Hipóteses.

Instância	**Horizonte**											
	1	**2**	**3**	**4**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
1	Não	Não	Não	Sim	Sim	Sim	Não	Sim	Sim	Sim	Sim	Sim
2	Não	Não	Não	Sim	Sim	Sim	Não	Sim	Sim	Sim	Sim	Sim
3	Não	Não	Não	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim
4	Não	Não	Não	Não	Não	Sim	Sim	Sim	Sim	Sim	Sim	Sim
5	Não	Não	Não	Não	Não	Sim	Sim	Sim	Sim	Sim	Sim	Sim

Fonte – Elaborada pelo autor.

Por fim, no que concerne ao tempo de execução, verifica-se que o algoritmo FIB tende a apresentar os melhores resultados para as instâncias 2, 3, 4 e 5. Como o desempenho dos algoritmos específicos é muito similar, pode-se usar a variável tempo de execução para fundamentar melhor a escolha do algoritmo.

8 Considerações Finais e Trabalhos Futuros

O presente trabalho investiga o emprego de Processos de Decisão de Markov Parcialmente Observáveis (POMDP) como modelo para a tomada de decisões pedagógicas em ambientes voltados ao aprendizado de algoritmos. A questão enfrentada é o sequenciamento de problemas computacionais em ambientes de tutoria inteligente, tendo em vista este ser um recurso comumente adotado em cursos introdutórios de algoritmos.

Neste contexto, o trabalho de organização das atividades de forma personalizada é encarado como um processo de tomada de decisão sequencial, em que um tutor inteligente é responsável por monitorar as soluções submetidas pelos alunos e por propor novas ações de ensino de forma sequencial, levando em consideração as respostas observadas anteriormente.

Assim, para analisar a aplicabilidade do modelo POMDP, simula-se o sequenciamento de um conjunto de problemas computacionais, os quais são ofertados segundo probabilidades preestabelecidas.

Os resultados mostram que, em termos de recompensa descontada média, para o algoritmos de planejamento específicos (QMDP, FIB, SARSOP, POMCP e POMCPOW) convergem para valores que se aproximam muito. Para um horizonte de planejamento de tamanho grande (ou infinito), a recompensa descontada média dos algoritmos específicos convergem para $272,60$; $353,29$; $424,95$; $481,93$; e $553,20$, respectivamente, para as instâncias 1, 2, 3, 4 e 5.

Por sua vez, os resultados também indicam que, em termos de recompensa descontada média, uma política aleatória sempre converge para um valor inferior ao dos algoritmos específicos. Para um horizonte de planejamento de tamanho grande (ou infinito), a recompensa descontada média da política aleatória converge para $260,22$; $330,53$; $392,22$; $431,96$; e $498,23$, respectivamente, para as instâncias 1, 2, 3, 4 e 5.

No que concerne ao tempo de execução, verifica-se que o algoritmo FIB apresenta os melhores resultados para as instâncias 2, 3, 4 e 5. Como o desempenho dos algoritmos específicos é muito similar, pode-se usar a variável tempo de execução para fundamentar a escolha do algoritmo.

A principal contribuição deste trabalho foi mostrar que o modelo POMDP pode ser aplicado de forma satisfatória ao processo de tomada de decisões pedagógicas, vez que se pode computar políticas de sequenciamento que proporcionam ganhos próximos do valor considerado ótimo.

Outro importante resultado deste trabalho foi a demonstração, por meio de simulação, de que algoritmos de planejamento podem ser aplicados de forma satisfatória ao sequenciamento de problemas computacionais, superando os resultados de políticas aleatórias.

Com relação aos trabalhos futuros, verifica-se necessidade de se ampliar o experimento para um conjunto número maior de problemas computacionais, assim como para outros algoritmos, proporcionando uma base mais robusta para a comparação dos resultados. Além disso, deve-se analisar a escalabilidade do modelo, visando maximizar o número de problemas que podem ser recomendados.

Por fim, também se pode aplicar o modelo proposto neste trabalho de uma forma prática, incorporando-o a um ambiente de tutoria inteligente. Nesse caso, o modelo serve como uma espécie de sistema de recomendação, fazendo o direcionamento quanto à próxima questão a ser respondida. O modelo pode ser construído levando em consideração os dados de um aluno em particular ou de uma turma, por meio de estatísticas-resumo (média, moda e mediana).

Referências

ALBRECHT, E.; GRABOWSKI, J. Towards a framework for mining students' programming assignments. In: IEEE. *2016 IEEE Global Engineering Education Conference (EDUCON)*. [S.l.], 2016. p. 1096–1100. Citado na página 51.

BARBOSA, A. d. A.; COSTA, E. d. B.; BRITO, P. H. Adaptive clustering of codes for assessment in introductory programming courses. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2018. p. 13–22. Citado na página 55.

Bez, J. L.; Tonin, N. A.; Rodegheri, P. R. Uri online judge academic: A tool for algorithms and programming classes. In: *2014 9th International Conference on Computer Science Education*. [S.l.: s.n.], 2014. p. 149–152. Citado na página 19.

BOUTILIER, C.; POOLE, D. Computing optimal policies for partially observable decision processes using compact representations. In: CITESEER. *Proceedings of the National Conference on Artificial Intelligence*. [S.l.], 1996. p. 1168–1175. Citado na página 39.

BRAZIUNAS, D. Pomdp solution methods. *University of Toronto*, 2003. Citado na página 23.

BUSSAB, W. d. O.; MORETTIN, P. A. Estatística básica. In: *Estatística básica*. [S.l.: s.n.], 2010. p. xvi–540. Citado na página 78.

CASSANDRA, A. R.; KAELBLING, L. P.; LITTMAN, M. L. Acting optimally in partially observable stochastic domains. In: *AAAI*. [S.l.: s.n.], 1994. v. 94, p. 1023–1028. Citado na página 39.

CHOUDHURY, R. R.; YIN, H.; FOX, A. Scale-driven automatic hint generation for coding style. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2016. p. 122–132. Citado na página 54.

COMANICI, G. *Representation Discovery for Markov Decision Processes Using Behavioural Similarity*. Tese (Doutorado) — McGill University Libraries, 2016. Citado na página 24.

FWA, H. L. Predicting non-completion of programming exercises using action logs and keystrokes. In: IEEE. *2019 International Symposium on Educational Technology (ISET)*. [S.l.], 2019. p. 271–275. Citado na página 54.

HAMID, O. H.; ALAIWY, F. H.; HUSSIEN, I. O. Uncovering cognitive influences on individualized learning using a hidden markov models framework. In: IEEE. *2015 Global Summit on Computer & Information Technology (GSCIT)*. [S.l.], 2015. p. 1–6. Citado na página 54.

HARSLEY, R. et al. Incorporating analogies and worked out examples as pedagogical strategies in a computer science tutoring system. In: ACM. *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. [S.l.], 2016. p. 675–680. Citado na página 53.

HAUSKRECHT, M. Value-function approximations for partially observable markov decision processes. *Journal of artificial intelligence research*, v. 13, p. 33–94, 2000. Citado 2 vezes nas páginas 66 e 68.

HOOSHYAR, D. et al. Flowchart-based bayesian intelligent tutoring system for computer programming. In: IEEE. *2015 International Conference on Smart Sensors and Application (ICSSA)*. [S.l.], 2015. p. 150–154. Citado na página 54.

III, C. C. W.; SCHERER, W. T. Solution procedures for partially observed markov decision processes. *Operations Research*, INFORMS, v. 37, n. 5, p. 791–797, 1989. Citado na página 39.

JIMÉNEZ, S. et al. Architecting an intelligent tutoring system with an affective dialogue module. In: IEEE. *2016 4th International Conference in Software Engineering Research and Innovation (CONISOFT)*. [S.l.], 2016. p. 122–129. Citado na página 53.

JIN, W. et al. Evaluation of guided-planning and assisted-coding with task relevant dynamic hinting. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2014. p. 318–328. Citado na página 55.

JUNIOR, G.; FREITAS, E. et al. *Uma abordagem para planejamento instrucional apoiada em processos de decisão Markovianos parcialmente observáveis*. Dissertação (Mestrado), 2018. Citado 2 vezes nas páginas 24 e 35.

KAELBLING, L. P.; LITTMAN, M. L.; CASSANDRA, A. R. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, Elsevier, v. 101, n. 1-2, p. 99–134, 1998. Citado 2 vezes nas páginas 37 e 38.

KITCHENHAM, B.; CHARTERS, S. Procedures for performing systematic literature reviews in software engineering. *Keele University & Durham University, UK*, 2007. Citado 2 vezes nas páginas 41 e 47.

KOCHENDERFER, M. J. *Decision making under uncertainty: theory and application*. [S.l.]: MIT press, 2015. Citado 2 vezes nas páginas 65 e 66.

KOCSIS, L.; SZEPESVÁRI, C. Bandit based monte-carlo planning. In: SPRINGER. *European conference on machine learning*. [S.l.], 2006. p. 282–293. Citado na página 69.

KOLOBOV, A. *Scalable methods and expressive models for planning under uncertainty*. Tese (Doutorado), 2013. Citado na página 24.

KUMAR, A. N. Providing the option to skip feedback in a worked example tutor. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2016. p. 101–110. Citado na página 52.

KUMAR, A. N. Using cloze procedure questions in worked examples in a programming tutor. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2016. p. 416–422. Citado na página 55.

KURNIAWATI, H.; HSU, D.; LEE, W. S. Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces. In: CITESEER. *Robotics: Science and systems*. [S.l.], 2008. v. 2008. Citado na página 67.

LEONG, F. H. Automatic detection of frustration of novice programmers from contextual and keystroke logs. In: IEEE. *2015 10th International Conference on Computer Science & Education (ICCSE)*. [S.l.], 2015. p. 373–377. Citado na página 53.

LISTER, R. et al. A multi-national study of reading and tracing skills in novice programmers. *ACM SIGCSE Bulletin*, ACM New York, NY, USA, v. 36, n. 4, p. 119–150, 2004. Citado na página 19.

LITTMAN, M. L. *Algorithms for sequential decision making*. [S.l.]: Brown University Providence, RI, 1996. Citado na página 24.

LITTMAN, M. L.; CASSANDRA, A. R.; KAELBLING, L. P. Learning policies for partially observable environments: Scaling up. In: *Machine Learning Proceedings 1995*. [S.l.]: Elsevier, 1995. p. 362–370. Citado na página 39.

MCCRACKEN, M. et al. A multi-national, multi-institutional study of assessment of programming skills of first-year cs students. In: *Working group reports from ITiCSE on Innovation and technology in computer science education*. [S.l.: s.n.], 2001. p. 125–180. Citado na página 19.

MELIANA, S.; NURJANAH, D. Adopting good-learners' paths in an intelligent tutoring system. In: IEEE. *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*. [S.l.], 2018. p. 877–882. Citado na página 55.

MOORE, A. W.; ATKESON, C. G. Prioritized sweeping: Reinforcement learning with less data and less time. *Machine learning*, Springer, v. 13, n. 1, p. 103–130, 1993. Citado na página 32.

OTTERLO, M. V.; WIERING, M. Reinforcement learning and markov decision processes. In: *Reinforcement Learning*. [S.l.]: Springer, 2012. p. 3–42. Citado 3 vezes nas páginas 24, 26 e 35.

PAS, A. T. *Simulation based planning for partially observable markov decision processes with continuous observation spaces*. Dissertação (Mestrado) — Faculty of Humanities and Sciences, Maastricht University, 2012. Citado 3 vezes nas páginas 34, 35 e 36.

PINEAU, J. et al. Point-based value iteration: An anytime algorithm for pomdps. In: *IJCAI*. [S.l.: s.n.], 2003. v. 3, p. 1025–1032. Citado na página 39.

RAFFERTY, A. N. et al. Faster teaching by pomdp planning. In: SPRINGER. *International Conference on Artificial Intelligence in Education*. [S.l.], 2011. p. 280–287. Citado na página 20.

RENS, G.; FERREIN, A.; POEL, E. V. D. Extending dtgolog to deal with pomdps. PRASA, 2008. Citado na página 34.

REVILLA, M. A.; MANZOOR, S.; LIU, R. Competitive learning in informatics: The uva online judge experience. *Olympiads in Informatics*, Institute of Mathematics and Informatics, v. 2, n. 10, p. 131–148, 2008. Citado na página 19.

ROSS, S. et al. Online planning algorithms for pomdps. *Journal of Artificial Intelligence Research*, v. 32, p. 663–704, 2008. Citado na página 66.

RUSSEL, S.; NORVIG, P. et al. *Artificial intelligence: a modern approach*. [S.l.]: Pearson Education Limited, 2016. Citado na página 23.

SAWAKI, K.; ICHIKAWA, A. Optimal control for partially observable markov decision processes over an infinite horizon. *Journal of the Operations Research Society of Japan*, The Operations Research Society of Japan, v. 21, n. 1, p. 1–16, 1978. Citado na página 39.

SILVER, D.; VENESS, J. Monte-carlo planning in large pomdps. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2010. p. 2164–2172. Citado 2 vezes nas páginas 39 e 68.

SMITH, T.; SIMMONS, R. Heuristic search value iteration for pomdps. *arXiv preprint arXiv:1207.4166*, 2012. Citado na página 33.

SMITH, T.; SIMMONS, R. Point-based pomdp algorithms: Improved analysis and implementation. *arXiv preprint arXiv:1207.1412*, 2012. Citado na página 39.

SOMANI, A. et al. Despot: Online pomdp planning with regularization. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2013. p. 1772–1780. Citado na página 36.

SONDIK, E. J. The optimal control of partially observable markov processes over the infinite horizon: Discounted costs. *Operations research*, INFORMS, v. 26, n. 2, p. 282–304, 1978. Citado na página 39.

SUNBERG, Z.; KOCHENDERFER, M. Online algorithms for pomdps with continuous state, action, and observation spaces. In: *Proceedings of the International Conference on Automated Planning and Scheduling*. [S.l.: s.n.], 2018. v. 28, n. 1. Citado na página 69.

SWAMY, V. et al. Deep knowledge tracing for free-form student code progression. In: SPRINGER. *International Conference on Artificial Intelligence in Education*. [S.l.], 2018. p. 348–352. Citado na página 53.

THEOCHAROUS, G. et al. Tractable pomdp planning algorithms for optimal teaching in "spais". In: *IJCAI PAIR Workshop*. [S.l.: s.n.], 2009. Citado na página 20.

TIAM-LEE, T. J.; SUMI, K. Analysis and prediction of student emotions while doing programming exercises. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2019. p. 24–33. Citado na página 55.

VAIL, A. K. et al. Predicting learning from student affective response to tutor questions. In: SPRINGER. *International Conference on Intelligent Tutoring Systems*. [S.l.], 2016. p. 154–164. Citado na página 52.

VESIN, B.; KLAŠNJA-MILIĆEVIĆ, A.; IVANOVIĆ, M. Protus 2.1: Applying collaborative tagging for providing recommendation in programming tutoring system. In: SPRINGER. *International Conference on Web-Based Learning*. [S.l.], 2016. p. 236–245. Citado na página 52.

WANG, F. Reinforcement learning in a pomdp based intelligent tutoring system for optimizing teaching strategies. *International Journal of Information and Education Technology*, v. 8, n. 8, 2018. Citado 2 vezes nas páginas 19 e 40.

WOOLF, B. P. *Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning*. [S.l.]: Morgan Kaufmann, 2010. Citado 2 vezes nas páginas 19 e 40.

XHAKAJ, F.; ALEVEN, V. Towards improving introductory computer programming with an its for conceptual learning. In: SPRINGER. *International Conference on Artificial Intelligence in Education*. [S.l.], 2018. p. 535–538. Citado na página 53.

ZHANG, N. L.; LIU, W. *Planning in stochastic domains: Problem characteristics and approximation*. [S.l.], 1996. Citado na página 39.

Apêndices

APÊNDICE A – Criação de instâncias na linguagem Julia

A criação das instâncias na linguagem Julia é feita por meio do pacote JuliaPOMDP. Essa ferramenta oferece suporte a uma interface para implementação de modelos POMDP com espaços de estados, ações e observações discretos, em que se faz necessária a implementação do conjunto de métodos descrito a seguir:

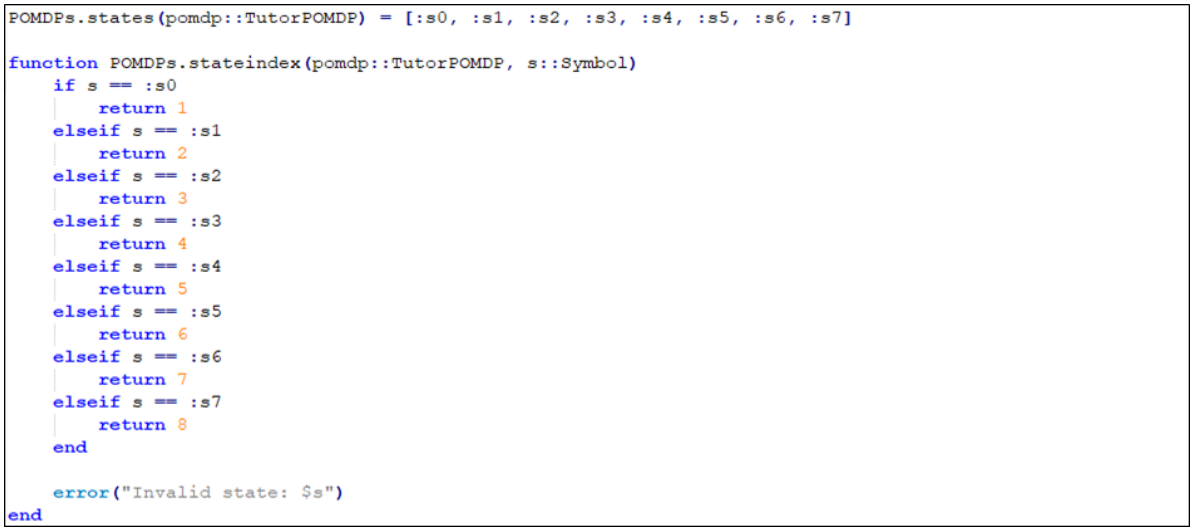
1. definição da distribuição de probabilidade do estado inicial (Figura A1);
2. declaração do espaço de estados (Figura A2);
3. declaração do espaço de ações (Figura A3);
4. declaração do espaço de observações (Figura A4);
5. declaração da função de transição (Figura A5);
6. declaração da função de observação (Figura A6);
7. declaração da função de recompensa (Figura A7);
8. declaração do fator de desconto (Figura A8);
9. instanciação do modelo (Figura A8).

Figura A1 – Definição da distribuição de probabilidade do estado inicial.

```
POMDPs.initialstate_distribution(pomdp::TutorPOMDP) =
SparseCat(
    [:s0, :s1, :s2, :s3, :s4, :s5, :s6, :s7],
    [1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
)
```

Fonte – Elaborada pelo autor.

Figura A2 – Definição do espaço de estados.



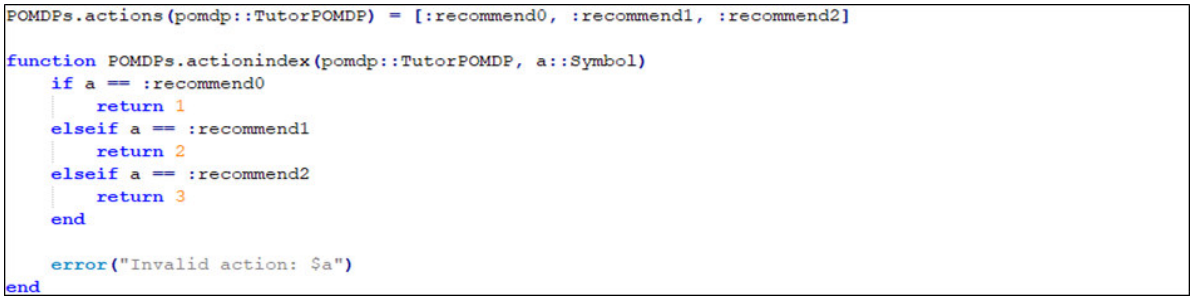
```
POMDPs.states(pomdp::TutorPOMDP) = [:s0, :s1, :s2, :s3, :s4, :s5, :s6, :s7]

function POMDPs.stateindex(pomdp::TutorPOMDP, s::Symbol)
    if s == :s0
        return 1
    elseif s == :s1
        return 2
    elseif s == :s2
        return 3
    elseif s == :s3
        return 4
    elseif s == :s4
        return 5
    elseif s == :s5
        return 6
    elseif s == :s6
        return 7
    elseif s == :s7
        return 8
    end

    error("Invalid state: $s")
end
```

Fonte – Elaborada pelo autor.

Figura A3 – Definição do espaço de ações.



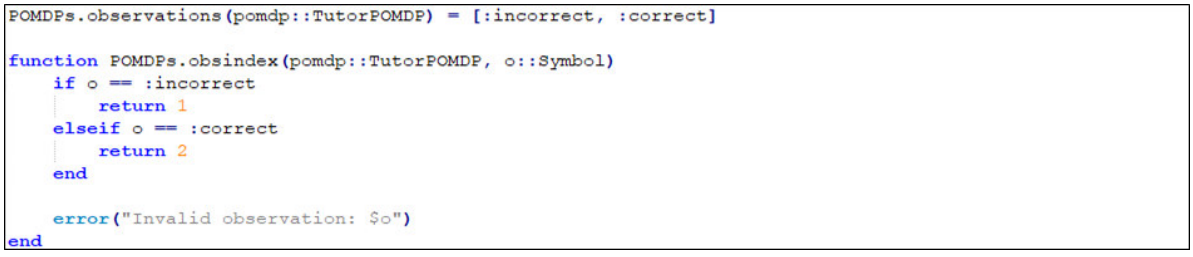
```
POMDPs.actions(pomdp::TutorPOMDP) = [:recommend0, :recommend1, :recommend2]

function POMDPs.actionindex(pomdp::TutorPOMDP, a::Symbol)
    if a == :recommend0
        return 1
    elseif a == :recommend1
        return 2
    elseif a == :recommend2
        return 3
    end

    error("Invalid action: $a")
end
```

Fonte – Elaborada pelo autor.

Figura A4 – Definição do espaço de observações.



```
POMDPs.observations(pomdp::TutorPOMDP) = [:incorrect, :correct]

function POMDPs.obsindex(pomdp::TutorPOMDP, o::Symbol)
    if o == :incorrect
        return 1
    elseif o == :correct
        return 2
    end

    error("Invalid observation: $o")
end
```

Fonte – Elaborada pelo autor.

Figura A5 – Definição da função de transição.

```
function POMDPs.transition(pomdp::TutorPOMDP, s::Symbol, a::Symbol)
    if a == :recommend0
        if s == :s0
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.3, 0.0, 0.0, 0.0, 0.7, 0.0, 0.0, 0.0])
        elseif s == :s1
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.3, 0.0, 0.0, 0.0, 0.7, 0.0, 0.0])
        elseif s == :s2
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.0, 0.3, 0.0, 0.0, 0.0, 0.7, 0.0])
        elseif s == :s3
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.0, 0.0, 0.3, 0.0, 0.0, 0.0, 0.7])
        elseif s == :s4
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0])
        elseif s == :s5
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0])
        elseif s == :s6
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0])
        elseif s == :s7
            return SparseCat([:s0,:s1,:s2,:s3,:s4,:s5,:s6,:s7], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0])
        end
        .
        .
        .
    end
end
```

Fonte – Elaborada pelo autor.

Figura A6 – Definição da função de observação.

```
function POMDPs.observation(pomdp::TutorPOMDP, s::Symbol, a::Symbol, sp::Symbol)
    if a == :recommend0
        if s == :s0
            if sp == :s0
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s1
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s2
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s3
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s4
                return SparseCat([:incorrect, :correct], [0.15, 0.85])
            elseif sp == :s5
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s6
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s7
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            end
        elseif s == :s1
            if sp == :s0
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s1
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s2
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s3
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s4
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s5
                return SparseCat([:incorrect, :correct], [0.15, 0.85])
            elseif sp == :s6
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            elseif sp == :s7
                return SparseCat([:incorrect, :correct], [0.85, 0.15])
            end
        elseif s == :s2
            .
            .
            .
        end
    end
end
```

Fonte – Elaborada pelo autor.

Figura A7 – Definição da função de recompensa.

```
function POMDPs.reward(pomdp::TutorPOMDP, s::Symbol, a::Symbol, sp::Symbol)
    if a == :recommend0
        if s == :s0
            if sp == :s0
                return 0
            elseif sp == :s1
                return -50
            elseif sp == :s2
                return -50
            elseif sp == :s3
                return -50
            elseif sp == :s4
                return 100
            elseif sp == :s5
                return -50
            elseif sp == :s6
                return -50
            elseif sp == :s7
                return -50
            end
        elseif s == :s1
            if sp == :s0
                return -50
            elseif sp == :s1
                return 0
            elseif sp == :s2
                return -50
            elseif sp == :s3
                return -50
            elseif sp == :s4
                return -50
            elseif sp == :s5
                return 100
            elseif sp == :s6
                return -50
            elseif sp == :s7
                return -50
            end
        .
        .
        .
    end
end
```

Fonte – Elaborada pelo autor.

Figura A8 – Definição do fator de desconto.

```
POMDPs.discount(pomdp::TutorPOMDP) = pomdp.discount_factor
```

Fonte – Elaborada pelo autor.

Figura A9 – Instanciação do modelo.

```
model = TutorPOMDP()
```

Fonte – Elaborada pelo autor.

APÊNDICE B – Dados da Instância 1

Tabela B1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	0,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
2	95,00	100,00	195,00	95,00	195,00	95,00	195,00	95,00	195,00	195,00
3	195,00	285,25	190,25	190,25	285,25	195,00	90,25	195,00	285,25	195,00
4	175,99	285,25	185,74	185,25	270,99	285,25	285,25	285,25	195,00	175,99
5	285,25	285,25	275,99	285,25	275,99	285,25	285,25	181,45	195,00	195,00
10	285,25	251,28	285,25	275,99	275,99	275,99	285,25	285,25	244,57	272,38
20	236,62	255,08	266,70	280,74	253,37	276,45	285,25	244,57	267,19	262,19
50	285,25	270,99	280,74	285,25	258,76	280,74	285,25	275,99	270,99	280,74
100	285,25	262,19	285,25	285,25	270,99	285,25	263,12	285,25	258,12	285,25
200	285,25	251,59	264,83	266,70	285,25	270,99	272,38	275,99	262,19	285,25
500	275,99	255,57	285,25	249,08	285,25	285,25	228,38	270,99	245,89	276,45
1000	259,25	258,12	285,25	272,38	241,14	280,74	267,19	285,25	280,74	285,25

Fonte – Elaborada pelo autor.

Tabela B2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	128	138	132	125	134	136	137	137	135	108
2	128	139	135	134	128	135	136	138	139	108
3	130	124	127	120	124	137	123	120	132	173
4	123	135	128	134	133	126	129	127	126	149
5	127	138	138	131	133	129	124	126	140	124
10	132	123	131	128	137	136	124	117	129	153
20	137	124	132	122	134	141	123	134	124	139
50	131	131	128	135	132	135	135	129	133	131
100	129	124	127	123	132	135	133	125	136	156
200	138	123	126	139	119	122	125	130	124	174
500	129	120	122	133	137	134	131	135	127	152
1000	139	131	139	143	138	128	123	135	128	126

Fonte – Elaborada pelo autor.

Tabela B3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 1, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	0,00	100,00	100,00	100,00	0,00	0,00	0,00	100,00	0,00
2	100,00	195,00	195,00	100,00	195,00	100,00	195,00	195,00	195,00	95,00
3	95,00	285,25	185,25	285,25	195,00	100,00	185,25	285,25	190,25	185,25
4	195,00	280,74	270,99	180,74	285,25	185,25	175,99	285,25	185,74	85,74
5	276,45	285,25	266,70	275,99	171,70	195,00	285,25	285,25	285,25	185,25
10	245,89	275,99	275,99	258,12	285,25	285,25	253,83	285,25	253,83	254,25
20	285,25	220,64	257,44	285,25	270,99	276,45	263,12	257,44	280,74	257,44
50	276,45	270,99	270,99	257,44	267,19	275,99	272,38	270,99	253,37	275,99
100	285,25	267,63	280,74	267,63	275,99	280,74	267,19	271,70	267,63	285,25
200	262,19	259,25	266,70	280,74	285,25	285,25	251,88	254,96	258,12	271,70
500	266,70	285,25	275,99	257,44	238,72	263,12	285,25	258,76	285,25	275,99
1000	258,76	285,25	285,25	275,99	267,19	285,25	285,25	268,51	270,99	280,74

Fonte – Elaborada pelo autor.

Tabela B4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	329	330	320	305	301	321	296	329	301	368
2	314	305	300	326	304	310	316	295	313	367
3	314	316	310	296	302	336	333	328	315	300
4	329	307	296	312	311	315	333	310	323	314
5	317	290	306	322	312	290	327	310	331	345
10	331	297	309	319	305	324	301	311	327	326
20	323	332	332	297	313	325	305	312	317	294
50	308	319	318	301	307	285	306	305	301	400
100	326	326	334	305	299	315	306	293	326	320
200	314	297	315	333	287	313	319	323	297	352
500	316	323	315	341	319	331	335	330	314	226
1000	308	307	314	328	311	319	305	342	318	348

Fonte – Elaborada pelo autor.

Tabela B5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	0,00	100,00	100,00	100,00	100,00	0,00	0,00	100,00
2	100,00	0,00	195,00	195,00	100,00	0,00	195,00	0,00	100,00	95,00
3	285,25	95,00	185,25	285,25	285,25	185,25	185,25	195,00	285,25	190,25
4	195,00	185,74	285,25	180,74	285,25	285,25	285,25	95,00	285,25	285,25
5	280,74	285,25	285,25	195,00	285,25	276,45	280,74	185,74	285,25	275,99
10	266,70	262,19	280,74	267,19	271,70	285,25	266,70	267,19	270,99	263,76
20	280,74	280,74	267,19	285,25	275,99	285,25	244,29	280,74	257,44	276,45
50	267,19	285,25	285,25	238,04	285,25	285,25	275,99	275,99	267,63	275,99
100	235,86	266,70	285,25	285,25	270,99	280,74	275,99	270,99	270,99	285,25
200	285,25	262,19	258,83	267,63	271,70	266,70	285,25	267,63	280,74	241,53
500	263,12	285,25	275,99	285,25	285,25	285,25	276,45	280,74	253,37	285,25
1000	270,99	285,25	280,74	285,25	275,99	285,25	285,25	262,63	268,51	266,70

Fonte – Elaborada pelo autor.

Tabela B6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1009	995	1060	1046	1040	1044	1025	1038	1009	1194
2	976	985	1034	969	998	1136	1037	1056	990	1269
3	1096	1089	1096	1099	1050	1008	1034	1015	1008	965
4	1026	1063	990	1037	1070	1063	1034	1055	1082	1050
5	1080	1009	1006	983	1049	1003	1017	1049	1095	1269
10	1080	1111	1084	1067	996	1050	1110	1029	1004	1009
20	1021	974	996	1078	1031	1116	1045	1035	1047	1157
50	984	1025	1051	1048	1058	1055	1050	1046	1023	1120
100	1025	1107	1058	1084	1077	1040	1035	1083	1065	856
200	1064	1033	1038	1090	1069	1047	1125	1020	1112	912
500	1033	996	1030	1099	1057	1061	1048	1024	1075	1117
1000	1110	1064	1055	960	1045	1019	1036	1054	1033	1164

Fonte – Elaborada pelo autor.

Tabela B7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	100,00	0,00	100,00	0,00	0,00	100,00
2	195,00	195,00	195,00	100,00	100,00	195,00	195,00	195,00	100,00	100,00
3	285,25	185,25	285,25	100,00	285,25	185,25	285,25	195,00	100,00	185,25
4	275,99	270,99	190,25	95,00	180,74	90,25	95,00	285,25	180,74	190,25
5	271,70	266,7	257,44	280,74	285,25	285,25	171,70	175,99	275,99	171,70
10	285,25	270,99	285,25	285,25	268,51	285,25	285,25	285,25	285,25	285,25
20	285,25	257,44	285,25	285,25	275,99	285,25	253,83	270,99	285,25	276,45
50	280,74	271,7	285,25	285,25	248,27	285,25	252,08	262,19	270,99	285,25
100	285,25	245,21	285,25	275,99	285,25	225,17	258,12	267,19	280,74	229,26
200	280,74	239,01	285,25	257,44	285,25	266,7	285,25	242,21	280,74	285,25
500	285,25	244,29	280,74	280,74	285,25	258,83	280,74	285,25	285,25	280,74
1000	266,7	262,19	285,25	257,44	270,99	244,57	285,25	275,99	266,7	275,99

Fonte – Elaborada pelo autor.

Tabela B8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	198	204	216	198	200	200	199	212	211	202
2	205	203	203	198	200	221	212	205	205	188
3	202	187	211	193	200	203	204	206	201	213
4	202	204	186	201	206	215	202	200	193	221
5	196	219	199	199	209	209	201	201	198	229
10	205	209	207	201	222	211	191	204	213	257
20	217	204	216	208	229	232	203	215	223	253
50	257	252	275	270	261	264	276	243	243	249
100	307	298	281	294	284	302	292	299	277	286
200	394	397	393	393	415	422	368	362	410	416
500	653	692	699	682	638	678	678	690	618	482
1000	1068	1054	1055	1048	1147	1134	1098	1015	1060	1391

Fonte – Elaborada pelo autor.

Tabela B9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	0,00	100,00	100,00	0,00	100,00	0,00	100,00
2	100,00	100,00	195,00	195,00	95,00	100,00	0,00	195,00	195,00	100,00
3	285,25	195,00	90,25	190,25	285,25	285,25	100,00	195,00	90,25	0,00
4	185,74	185,25	190,25	180,74	275,99	190,25	100,00	275,99	85,74	285,25
5	257,44	267,19	185,25	171,70	167,19	285,25	267,19	276,45	190,25	285,25
10	266,70	271,70	262,19	275,99	260,08	270,99	262,19	263,12	276,45	246,28
20	208,01	214,45	236,58	264,83	240,70	233,97	275,99	280,74	275,99	285,25
50	229,26	276,45	262,19	238,33	270,99	280,74	233,38	285,25	267,19	245,21
100	246,28	264,83	223,26	256,59	250,57	272,38	255,08	285,25	272,38	215,96
200	276,45	276,45	285,25	256,59	255,08	262,19	272,38	251,59	280,74	237,02
500	285,25	258,83	257,44	280,74	285,25	271,70	270,99	258,12	260,08	285,25
1000	244,48	280,74	285,25	285,25	258,83	285,25	270,99	226,41	280,74	275,99

Fonte – Elaborada pelo autor.

Tabela B10 – Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	284	329	306	287	281	292	296	318	308	319
2	315	321	315	296	297	294	304	321	295	352
3	297	289	320	295	319	301	308	306	328	377
4	308	310	293	288	292	317	310	292	339	421
5	326	334	317	342	315	326	323	326	337	264
10	354	336	338	337	350	329	335	331	323	357
20	361	349	370	376	383	395	358	366	394	388
50	482	460	484	508	476	508	492	476	463	461
100	731	673	666	664	674	646	634	679	681	632
200	1103	1138	1053	1021	1087	1010	1091	1037	1089	821
500	2189	2089	2195	2147	2103	2184	2160	2181	2126	1986
1000	4150	4022	3988	3943	4107	3838	4019	4035	3781	4097

Fonte – Elaborada pelo autor.

Tabela B11 – Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	0,00	100,00	100,00	0,00	100,00	0,00	100,00	100,00	0,00
2	0,00	95,00	0,00	100,00	100,00	195,00	100,00	95,00	95,00	195,00
3	100,00	190,25	100,00	95,00	100,00	95,00	195,00	100,00	100,00	100,00
4	285,25	95,00	275,99	280,74	185,25	90,25	195,00	0,00	195,00	95,00
5	195,00	181,45	180,74	271,70	190,25	271,70	195,00	270,99	275,99	195,00
10	158,02	253,27	266,70	254,96	253,37	267,19	258,76	181,45	254,96	245,89
20	261,34	285,25	258,02	258,12	259,25	285,25	280,74	249,04	280,74	247,08
50	267,63	258,12	251,88	280,74	275,99	243,77	285,25	236,53	253,37	267,19
100	280,74	234,85	275,99	267,63	236,32	276,45	218,03	227,35	245,82	249,04
200	261,34	266,70	247,21	261,34	259,25	285,25	262,63	239,01	275,99	271,70
500	248,27	276,45	271,70	244,48	285,25	271,70	268,51	263,76	245,89	256,59
1000	243,76	249,08	253,37	272,38	251,88	272,38	267,63	285,25	257,44	249,04

Fonte – Elaborada pelo autor.

Tabela B12 – Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 1.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	149	141	141	147	128	147	130	137	148	132
2	147	146	135	140	130	133	143	141	143	142
3	132	135	149	135	135	145	143	137	144	145
4	150	129	134	142	151	135	145	132	140	142
5	142	129	139	141	148	147	136	145	133	140
10	139	142	139	144	141	141	138	139	138	139
20	126	136	133	139	138	135	126	142	142	183
50	141	143	130	134	135	139	132	140	133	173
100	135	132	143	142	128	145	144	131	137	163
200	135	128	151	140	150	134	132	138	135	157
500	146	140	131	139	135	135	137	137	130	170
1000	133	132	141	129	140	146	147	136	152	144

Fonte – Elaborada pelo autor.

APÊNDICE C – Dados da Instância 2

Tabela C1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	100,00	100,00	0,00	100,00	0,00	100,00	100,00	100,00
2	195,00	95,00	195,00	95,00	95,00	100,00	100,00	100,00	100,00	95,00
3	285,25	185,25	90,25	190,25	90,25	90,25	285,25	95,00	285,25	195,00
4	370,99	285,25	280,74	195,00	280,74	185,25	190,25	275,99	100,00	185,25
5	266,70	370,99	370,99	352,44	271,70	357,44	370,99	352,44	370,99	352,44
10	285,25	358,12	349,96	352,44	370,99	370,99	210,82	370,99	370,99	136,18
20	349,50	319,75	362,19	370,99	357,44	362,63	354,25	326,41	370,99	362,19
50	370,99	370,99	349,50	370,99	310,97	358,12	270,16	351,59	352,44	336,14
100	370,99	352,44	320,17	370,99	370,99	358,12	344,57	348,37	366,70	370,99
200	362,19	326,87	327,95	311,55	370,99	370,99	370,99	370,99	352,44	323,20
500	362,19	339,57	326,87	370,99	366,70	370,99	362,63	352,44	370,99	357,44
1000	357,44	370,99	335,70	333,59	370,99	358,12	370,99	349,08	350,57	362,63

Fonte – Elaborada pelo autor.

Tabela C2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	509	500	510	485	494	497	515	491	471	548
2	498	500	494	474	475	506	487	525	528	533
3	490	532	531	530	508	479	480	495	489	486
4	507	494	525	498	495	493	509	521	519	459
5	462	523	471	505	511	534	476	491	535	512
10	515	465	473	540	490	520	465	482	493	577
20	523	513	489	511	516	514	485	458	455	556
50	492	493	510	495	514	502	487	546	492	489
100	505	509	532	496	528	521	460	524	528	447
200	502	493	508	533	499	481	521	535	515	433
500	493	544	462	508	533	494	522	505	480	509
1000	501	512	516	502	469	537	518	471	497	577

Fonte – Elaborada pelo autor.

Tabela C3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	0,00	100,00	100,00	100,00	100,00	100,00
2	100,00	100,00	195,00	95,00	195,00	100,00	195,00	95,00	195,00	195,00
3	0,00	190,25	0,00	195,00	185,25	195,00	285,25	285,25	95,00	190,25
4	285,25	280,74	370,99	175,99	195,00	285,25	275,99	370,99	280,74	280,74
5	175,99	362,19	352,44	280,74	262,19	370,99	357,44	370,99	262,19	262,19
10	370,99	357,44	352,44	327,95	294,20	370,99	341,14	366,70	338,04	362,63
20	355,08	370,99	362,19	370,99	370,99	370,99	334,82	317,99	332,46	291,05
50	352,44	306,46	370,99	329,73	357,44	332,34	324,51	370,99	366,70	333,04
100	370,99	303,80	352,44	370,99	332,02	353,37	345,12	346,28	349,08	357,44
200	354,25	341,53	370,99	370,99	334,82	370,99	362,19	362,63	357,44	370,99
500	324,46	357,44	345,12	370,99	370,99	370,99	353,83	344,57	338,04	357,44
1000	366,70	339,57	370,99	357,44	357,44	328,59	370,99	370,99	327,34	302,17

Fonte – Elaborada pelo autor.

Tabela C4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	453	422	424	423	443	443	436	445	473	488
2	453	436	435	442	415	443	432	405	466	423
3	410	414	439	433	404	394	418	402	418	418
4	441	451	424	408	439	414	399	434	421	519
5	433	395	426	393	430	411	409	416	401	486
10	432	390	425	421	431	383	407	440	429	442
20	442	383	383	408	415	418	384	408	376	483
50	380	426	440	392	401	415	409	401	413	473
100	439	421	439	392	378	378	392	421	388	452
200	390	438	443	384	415	425	405	378	431	441
500	431	407	439	415	446	399	420	438	423	532
1000	414	447	413	416	442	409	428	417	456	458

Fonte – Elaborada pelo autor.

Tabela C5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	0,00	0,00	100,00	100,00	100,00	0,00	100,00	0,00	100,00
2	195,00	100,00	195,00	95,00	195,00	195,00	0,00	100,00	100,00	95,00
3	195,00	185,25	0,00	185,25	100,00	185,25	190,25	190,25	285,25	285,25
4	370,99	275,99	270,99	285,25	280,74	285,25	175,99	270,99	280,74	185,25
5	357,44	280,74	167,19	362,19	370,99	362,19	370,99	185,74	267,19	352,44
10	362,19	333,53	331,53	344,57	345,82	336,62	335,70	335,70	349,50	370,99
20	334,82	366,70	370,99	334,82	370,99	349,08	315,44	322,59	319,79	332,34
50	353,37	307,04	353,37	349,08	319,12	313,24	349,50	335,70	315,04	366,70
100	370,99	280,00	292,73	370,99	357,44	357,44	314,96	362,19	349,50	366,70
200	370,99	320,46	344,57	340,82	362,19	370,99	358,12	353,37	353,37	362,63
500	327,27	344,50	362,19	370,99	366,70	370,99	370,99	362,19	370,99	345,82
1000	357,44	353,83	345,89	340,82	370,99	370,99	370,99	339,57	349,50	357,44

Fonte – Elaborada pelo autor.

Tabela C6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1826	1875	1899	1883	1863	1928	1927	1742	1813	2264
2	2005	2073	2026	2040	1995	1959	1996	1895	2080	1601
3	2045	2014	1980	2031	1862	2024	2062	1951	2005	1876
4	1957	1915	1863	1872	2008	1883	1797	1850	2071	2274
5	1953	2077	1969	2020	1839	1957	1966	2069	1886	1634
10	1963	2024	1860	1993	2014	1939	2007	1914	2035	1801
20	1988	1875	1943	2006	2029	2045	1923	1836	1918	1807
50	1974	1976	1858	1844	1889	1912	1960	1913	2001	1623
100	1810	1938	1856	1900	1972	1904	1919	1919	2035	1577
200	1985	1988	2000	1919	1946	1931	1854	1918	1896	1533
500	1889	1803	1963	1912	1889	1906	1925	1973	1919	1781
1000	1729	2041	1881	1966	1883	2014	1978	1918	1864	1666

Fonte – Elaborada pelo autor.

Tabela C7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	0,00	100,00	100,00	100,00	100,00	0,00	100,00	0,00	0,00
2	195,00	195,00	195,00	95,00	0,00	100,00	100,00	0,00	195,00	195,00
3	185,25	100,00	285,25	195,00	195,00	285,25	195,00	190,25	285,25	190,25
4	190,25	280,74	180,74	285,25	180,74	180,74	180,74	370,99	370,99	280,74
5	95,00	195,00	181,45	352,44	366,70	195,00	370,99	352,44	262,19	185,25
10	254,25	353,83	262,19	347,08	323,66	325,65	362,19	354,25	353,37	357,44
20	340,61	349,96	370,99	370,99	370,99	353,83	370,99	330,95	344,57	366,70
50	349,96	344,57	333,97	314,32	344,57	362,19	353,37	297,34	339,57	352,44
100	348,37	340,61	332,02	349,08	334,82	370,99	344,57	353,37	370,99	353,83
200	347,08	370,99	350,57	344,57	314,40	354,25	350,57	330,86	362,19	370,99
500	370,99	366,70	322,59	302,17	349,50	357,44	353,37	327,27	281,18	362,19
1000	370,99	362,19	322,50	352,44	334,82	339,57	352,44	370,99	370,99	348,37

Fonte – Elaborada pelo autor.

Tabela C8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	828	775	788	771	825	815	747	810	793	898
2	835	836	803	820	813	816	769	811	822	715
3	797	815	798	822	779	851	784	774	737	923
4	785	837	806	838	737	843	833	799	800	802
5	839	823	784	852	797	766	817	823	832	757
10	801	755	893	741	881	817	824	849	790	819
20	842	803	833	801	770	762	794	854	843	1048
50	874	842	857	880	869	886	931	851	889	931
100	898	1015	921	968	954	918	987	928	970	1091
200	1100	1078	1036	1089	1154	1055	1091	1138	1171	1158
500	1591	1505	1517	1646	1562	1506	1565	1520	1484	1304
1000	2126	2305	2331	2296	2377	2331	2303	2313	2178	2450

Fonte – Elaborada pelo autor.

Tabela C9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	0,00	100,00	100,00	0,00	0,00	100,00	100,00
2	100,00	195,00	195,00	100,00	95,00	100,00	100,00	100,00	195,00	95,00
3	90,25	195,00	100,00	185,25	185,25	100,00	285,25	190,25	95,00	190,25
4	95,00	195,00	275,99	280,74	185,74	195,00	275,99	285,25	85,74	280,74
5	195,00	370,99	352,44	285,25	280,74	276,45	175,99	180,74	370,99	285,25
10	370,99	311,55	247,08	350,57	370,99	362,19	362,19	370,99	339,57	354,25
20	313,51	328,59	333,59	306,38	324,37	245,61	358,12	348,37	370,99	353,83
50	370,99	274,68	316,59	332,02	287,66	327,87	314,45	321,66	336,58	270,21
100	370,99	348,37	362,19	327,34	328,53	303,85	315,84	349,96	370,99	370,99
200	305,09	358,76	366,70	357,44	323,66	345,21	352,44	366,70	307,60	345,82
500	366,70	325,17	295,37	358,12	325,22	370,99	370,99	316,39	338,04	269,93
1000	344,50	370,99	355,08	354,25	370,99	338,34	370,99	322,55	357,44	344,08

Fonte – Elaborada pelo autor.

Tabela C10 – Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	747	738	764	705	768	753	728	760	739	768
2	731	798	702	760	735	718	684	708	724	820
3	801	743	741	756	711	791	772	787	733	655
4	760	726	752	762	727	722	722	729	700	920
5	753	745	691	779	741	807	773	760	725	786
10	793	851	850	812	795	866	830	798	756	609
20	907	806	865	808	803	785	825	820	844	947
50	987	931	962	1033	957	940	947	994	1005	1184
100	1194	1280	1319	1147	1316	1252	1167	1305	1240	1310
200	1642	1686	1632	1594	1705	1524	1662	1561	1688	1816
500	3031	3113	2886	2809	2864	2985	2872	3066	2925	2609
1000	5358	5248	5273	5160	5015	5651	5431	5308	5470	4096

Fonte – Elaborada pelo autor.

Tabela C11 – Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	0,00	100,00	0,00	100,00	0,00	100,00	100,00	0,00	100,00
2	195,00	195,00	100,00	95,00	100,00	195,00	100,00	195,00	0,00	195,00
3	195,00	185,25	185,25	100,00	190,25	195,00	195,00	195,00	285,25	190,25
4	285,25	185,74	195,00	185,74	185,74	90,25	185,74	270,99	100,00	195,00
5	262,19	195,00	180,74	100,00	100,00	262,19	285,25	276,45	280,74	180,74
10	353,83	327,95	258,02	185,25	338,72	272,38	241,53	337,33	275,99	338,04
20	319,51	358,12	302,20	353,37	326,78	334,73	264,83	323,71	362,63	332,95
50	298,40	315,00	270,59	309,44	311,55	319,71	357,44	317,90	298,26	295,09
100	333,04	285,26	309,83	336,62	340,61	307,45	354,25	348,27	353,37	273,07
200	348,27	315,64	293,46	339,57	342,21	334,73	294,44	339,57	310,09	351,59
500	293,05	285,30	314,62	319,62	280,86	342,13	322,78	315,72	244,25	323,58
1000	342,33	316,91	345,82	357,44	302,13	353,83	337,62	294,68	339,48	315,10

Fonte – Elaborada pelo autor.

Tabela C12 – Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 2.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	596	602	591	610	623	579	596	559	608	626
2	587	573	611	640	586	583	624	632	614	540
3	595	621	576	631	597	626	564	648	588	544
4	655	641	621	635	622	588	589	634	631	374
5	612	593	644	570	605	608	620	572	575	591
10	588	615	623	594	586	574	567	607	601	635
20	615	614	571	655	621	607	600	597	605	505
50	587	614	597	610	631	601	604	587	573	586
100	576	555	616	574	565	584	639	561	606	714
200	584	608	566	594	597	552	598	597	562	732
500	652	603	610	559	590	569	569	572	602	664
1000	585	611	542	595	637	586	588	590	544	712

Fonte – Elaborada pelo autor.

APÊNDICE D – Dados da Instância 3

Tabela D1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	0,00	0,00	0,00	100,00	100,00	100,00	100,00	0,00
2	195,00	195,00	195,00	195,00	100,00	95,00	195,00	100,00	95,00	195,00
3	190,25	285,25	185,25	285,25	190,25	185,25	95,00	185,25	185,25	285,25
4	190,25	370,99	190,25	275,99	185,74	190,25	280,74	85,74	95,00	370,99
5	275,99	357,44	185,25	352,44	285,25	270,99	370,99	267,19	285,25	362,19
10	422,59	427,27	439,57	426,87	426,87	412,98	402,17	399,56	452,44	420,46
20	452,44	431,62	439,57	409,12	413,91	444,08	427,34	412,23	444,08	416,39
50	422,32	417,59	434,82	452,44	452,44	427,87	427,27	370,78	452,44	434,82
100	412,98	440,21	382,86	448,37	420,74	364,57	404,65	427,95	399,29	418,08
200	429,82	452,44	436,53	427,27	434,82	363,24	391,69	436,14	440,82	400,78
500	452,44	384,67	390,57	384,20	434,01	452,44	376,99	429,82	398,68	418,08
1000	432,02	448,37	434,82	452,44	423,66	439,57	429,82	452,44	429,82	425,02

Fonte – Elaborada pelo autor.

Tabela D2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	5574	5584	5581	5589	5573	5587	5577	5578	5586	5591
2	5574	5584	5586	5581	5578	5579	5585	5574	5576	5603
3	5575	5587	5579	5580	5593	5582	5569	5586	5573	5596
4	5580	5573	5571	5578	5593	5579	5569	5586	5587	5604
5	5576	5585	5588	5588	5574	5584	5576	5583	5580	5586
10	5581	5583	5579	5587	5584	5580	5585	5577	5576	5588
20	5579	5590	5577	5583	5573	5574	5584	5572	5592	5596
50	5586	5601	5593	5591	5606	5594	5597	5598	5591	5573
100	5575	5582	5581	5574	5583	5594	5579	5587	5572	5603
200	5591	5599	5585	5574	5589	5596	5575	5582	5596	5573
500	5599	5604	5605	5601	5585	5589	5593	5597	5599	5608
1000	5594	5599	5606	5598	5603	5592	5602	5605	5590	5621

Fonte – Elaborada pelo autor.

Tabela D3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	0,00	100,00	100,00	0,00	0,00	0,00	0,00	100,00	100,00
2	195,00	95,00	195,00	0,00	95,00	0,00	95,00	195,00	0,00	195,00
3	190,25	285,25	185,25	285,25	195,00	185,25	285,25	190,25	285,25	100,00
4	370,99	185,74	370,99	180,74	370,99	175,99	275,99	100,00	370,99	280,74
5	362,19	452,44	370,99	362,19	452,44	352,44	452,44	352,44	175,99	362,19
10	389,54	452,44	432,46	346,28	400,78	410,42	402,17	362,19	429,82	401,46
20	424,08	452,44	397,17	452,44	434,82	434,82	436,14	429,82	394,29	452,44
50	434,82	430,95	386,98	422,59	388,85	321,09	452,44	444,50	452,44	407,10
100	401,16	372,80	425,95	452,44	434,82	439,57	401,41	386,14	436,53	425,65
200	434,82	444,08	448,37	452,44	386,15	439,57	402,48	390,81	439,57	452,44
500	383,58	413,08	423,66	434,82	332,22	422,06	376,18	428,53	452,44	405,15
1000	417,59	448,37	398,80	379,20	452,44	430,95	401,01	433,04	436,14	444,08

Fonte – Elaborada pelo autor.

Tabela D4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	3421	3412	3415	3419	3413	3408	3420	3416	3410	3466
2	3459	3462	3453	3460	3472	3453	3473	3452	3462	3504
3	3603	3593	3603	3605	3592	3608	3603	3602	3605	3636
4	3561	3563	3554	3559	3554	3563	3555	3563	3541	3537
5	3632	3629	3638	3634	3631	3624	3636	3630	3638	3658
10	3635	3650	3638	3649	3643	3658	3637	3643	3639	3658
20	3437	3447	3451	3437	3450	3446	3450	3443	3446	3443
50	3452	3463	3466	3456	3444	3456	3464	3456	3451	3442
100	3533	3551	3527	3540	3539	3533	3543	3536	3541	3557
200	3688	3696	3691	3696	3696	3685	3698	3687	3691	3672
500	3598	3591	3596	3597	3600	3604	3595	3611	3605	3603
1000	3626	3602	3612	3607	3607	3620	3612	3611	3619	3634

Fonte – Elaborada pelo autor.

Tabela D5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	100,00	0,00	0,00	0,00	100,00	100,00	100,00	100,00
2	95,00	95,00	100,00	195,00	195,00	100,00	195,00	195,00	100,00	195,00
3	185,25	285,25	190,25	190,25	285,25	185,25	285,25	195,00	285,25	285,25
4	370,99	275,99	275,99	285,25	270,99	180,74	175,99	285,25	280,74	185,25
5	257,44	362,19	452,44	366,70	270,99	452,44	270,99	370,99	366,70	452,44
10	434,82	452,44	332,46	430,95	390,97	427,34	448,37	452,44	452,44	448,37
20	428,59	439,57	429,82	349,46	448,37	405,53	423,66	394,65	411,47	390,36
50	452,44	381,94	435,70	452,44	413,08	452,44	430,95	410,84	412,31	435,70
100	397,17	387,91	425,95	452,44	452,44	452,44	401,16	429,82	440,21	358,96
200	407,04	405,53	397,85	425,95	403,72	436,14	344,13	402,04	452,44	434,82
500	427,27	430,95	394,20	413,52	439,57	444,50	420,17	415,04	385,67	381,15
1000	452,44	403,53	444,08	429,82	423,78	390,09	433,04	452,44	434,82	375,18

Fonte – Elaborada pelo autor.

Tabela D6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	6846	6843	6837	6853	6851	6829	6848	6828	6845	6820
2	6838	6847	6847	6851	6835	6844	6852	6839	6844	6813
3	6823	6829	6831	6823	6833	6836	6831	6834	6837	6873
4	6831	6834	6829	6835	6827	6835	6844	6832	6844	6839
5	6826	6826	6831	6831	6823	6831	6837	6826	6831	6868
10	6862	6873	6872	6863	6862	6853	6861	6865	6858	6881
20	6853	6853	6843	6840	6832	6853	6843	6839	6835	6869
50	6837	6840	6848	6838	6838	6835	6843	6835	6839	6847
100	6827	6829	6839	6847	6831	6830	6839	6834	6834	6850
200	6850	6852	6852	6853	6854	6853	6860	6850	6850	6856
500	6858	6850	6857	6849	6860	6858	6862	6860	6857	6849
1000	6849	6849	6854	6853	6855	6858	6845	6858	6860	6889

Fonte – Elaborada pelo autor.

Tabela D7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	100,00	0,00	0,00	0,00	100,00	100,00	0,00	0,00
2	100,00	195,00	100,00	100,00	0,00	195,00	195,00	195,00	95,00	195,00
3	285,25	95,00	185,25	195,00	190,25	190,25	90,25	285,25	185,25	190,25
4	95,00	280,74	280,74	270,99	190,25	90,25	195,00	280,74	370,99	85,74
5	357,44	370,99	257,44	280,74	357,44	452,44	262,19	276,45	357,44	280,74
10	425,95	239,01	338,04	412,10	355,08	399,65	452,44	336,14	439,57	405,97
20	398,80	428,53	415,42	399,10	372,85	378,87	410,42	452,44	402,17	392,34
50	452,44	433,04	452,44	418,08	402,97	415,46	444,08	421,87	448,37	452,44
100	393,02	404,65	414,33	429,82	427,27	413,08	419,79	434,82	431,62	372,04
200	415,42	436,14	422,59	444,08	397,97	359,68	452,44	387,06	405,91	434,82
500	354,93	373,39	364,18	417,32	430,95	385,09	429,82	363,24	430,95	444,08
1000	418,03	402,55	432,02	370,99	452,44	432,02	440,21	410,97	452,44	410,10

Fonte – Elaborada pelo autor.

Tabela D8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	6423	6477	6484	6474	6478	6481	6476	6481	6467	6509
2	6485	6480	6485	6483	6493	6479	6485	6493	6487	6490
3	6478	6485	6485	6476	6483	6479	6480	6479	6480	6475
4	6495	6484	6487	6496	6489	6495	6492	6493	6503	6506
5	6495	6500	6500	6496	6498	6499	6492	6489	6485	6516
10	6508	6493	6508	6506	6506	6499	6499	6492	6487	6492
20	6534	6523	6525	6519	6529	6527	6513	6524	6516	6510
50	6595	6596	6597	6593	6599	6596	6602	6586	6590	6586
100	6714	6721	6705	6719	6721	6713	6708	6717	6724	6738
200	6955	6955	6949	6959	6957	6965	6961	6951	6968	6980
500	7707	7709	7695	7702	7704	7715	7702	7710	7703	7693
1000	8880	8888	8874	8873	8887	8882	8882	8881	8878	8915

Fonte – Elaborada pelo autor.

Tabela D9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	100,00	100,00	0,00	100,00	0,00	100,00	100,00	100,00
2	195,00	95,00	95,00	195,00	195,00	100,00	100,00	100,00	95,00	95,00
3	185,25	90,25	285,25	190,25	195,00	195,00	190,25	95,00	90,25	95,00
4	180,74	285,25	280,74	280,74	280,74	275,99	370,99	185,74	285,25	280,74
5	366,70	370,99	267,19	271,70	362,19	452,44	181,45	262,19	280,74	370,99
10	362,63	434,82	343,76	428,53	324,37	232,95	425,65	257,44	394,93	341,53
20	381,23	389,92	380,74	405,96	368,23	434,82	396,25	351,08	367,75	448,37
50	395,57	422,59	410,42	417,32	385,62	410,53	352,36	387,49	430,86	402,59
100	376,62	434,82	421,43	429,82	388,01	380,46	440,21	374,76	452,44	421,78
200	407,10	408,85	452,44	410,53	408,24	334,09	383,20	444,08	395,57	408,95
500	363,18	427,95	444,08	374,58	407,86	406,47	310,07	444,50	411,47	389,54
1000	394,60	444,08	435,70	412,10	372,47	401,12	434,01	389,97	440,21	452,44

Fonte – Elaborada pelo autor.

Tabela D10 – Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	5949	5924	5940	5940	5941	5940	5936	5941	5935	5934
2	5964	5958	5952	5965	5963	5965	5962	5969	5969	5983
3	5963	5963	5947	5957	5970	5962	5961	5952	5963	5952
4	5978	5966	5966	5964	5956	5964	5970	5979	5969	5988
5	5971	5980	5977	5984	5982	5990	5972	5973	5989	5982
10	6041	6030	6029	6029	6022	6031	6030	6024	6022	6062
20	6143	6124	6128	6122	6132	6137	6135	6128	6133	6168
50	6439	6447	6453	6449	6453	6441	6448	6450	6448	6452
100	6639	6637	6624	6639	6637	6637	6631	6640	6645	6631
200	7303	7310	7303	7307	7312	7315	7310	7307	7308	7345
500	9355	9359	9357	9368	9351	9355	9362	9359	9356	9418
1000	12840	12844	12847	12840	12834	12842	12844	12838	12846	12835

Fonte – Elaborada pelo autor.

Tabela D11 – Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	0,00	100,00	100,00	0,00	100,00	100,00	100,00	0,00
2	195,00	195,00	100,00	195,00	195,00	195,00	195,00	195,00	100,00	100,00
3	90,25	190,25	190,25	195,00	285,25	195,00	195,00	195,00	285,25	185,25
4	185,25	275,99	280,74	275,99	180,74	190,25	280,74	95,00	195,00	185,74
5	190,25	181,45	190,25	370,99	262,19	181,45	267,19	185,74	271,70	285,25
10	227,86	260,08	320,46	309,68	419,79	370,99	330,95	266,70	333,53	325,17
20	390,41	432,46	323,58	376,05	388,31	432,02	274,50	258,83	327,06	427,27
50	383,62	421,87	375,70	373,19	385,55	383,67	360,87	406,84	316,65	364,92
100	440,21	363,40	382,42	371,08	415,84	433,04	382,86	374,19	350,02	395,42
200	411,39	348,96	395,57	431,62	395,89	317,23	424,46	335,95	401,41	408,96
500	361,92	396,50	350,30	335,55	345,12	360,24	351,08	358,14	348,07	366,98
1000	378,54	421,87	448,37	380,66	410,09	369,41	343,67	436,53	388,47	344,58

Fonte – Elaborada pelo autor.

Tabela D12 – Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 3.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	8244	8235	8247	8247	8244	8240	8239	8238	8235	8261
2	8244	8254	8241	8233	8243	8251	8237	8229	8234	8264
3	8239	8236	8246	8236	8240	8238	8235	8250	8228	8282
4	8245	8246	8243	8249	8249	8245	8244	8249	8228	8232
5	8237	8235	8250	8249	8242	8240	8251	8235	8239	8252
10	8237	8239	8230	8243	8249	8242	8242	8238	8244	8266
20	8248	8248	8241	8238	8243	8239	8234	8247	8234	8258
50	8253	8249	8232	8239	8242	8247	8253	8242	8244	8229
100	8249	8249	8251	8239	8240	8242	8251	8251	8250	8208
200	8246	8243	8253	8245	8246	8231	8245	8242	8236	8243
500	8238	8238	8233	8235	8239	8252	8237	8236	8231	8291
1000	8235	8244	8254	8239	8249	8238	8244	8237	8243	8257

Fonte – Elaborada pelo autor.

APÊNDICE E – Dados da Instância 4

Tabela E1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	100,00	0,00	100,00	0,00	100,00	100,00
2	195,00	95,00	195,00	195,00	95,00	95,00	100,00	195,00	195,00	100,00
3	185,25	195,00	190,25	285,25	285,25	285,25	285,25	285,25	90,25	195,00
4	185,74	185,25	270,99	280,74	185,74	285,25	195,00	280,74	275,99	270,99
5	285,25	452,44	357,44	452,44	366,70	185,25	190,25	352,44	180,74	370,99
10	517,59	455,45	393,13	415,46	503,33	499,65	529,82	458,03	499,65	420,17
20	508,33	500,69	522,27	477,31	503,33	453,60	521,87	459,94	525,95	499,44
50	503,33	490,41	397,45	503,85	508,33	507,10	517,59	503,33	419,28	529,82
100	529,82	491,71	492,00	478,45	460,73	521,87	517,59	452,37	506,47	476,67
200	521,87	503,33	504,65	529,82	485,90	490,36	529,82	513,08	503,33	529,82
500	457,31	377,53	517,59	529,82	496,16	529,82	503,33	525,95	513,08	529,82
1000	481,15	483,93	529,82	492,34	510,04	481,81	513,08	456,24	466,33	400,25

Fonte – Elaborada pelo autor.

Tabela E2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	202106	221827	216343	242012	233006	216812	216215	207787	228446	228246
2	214302	220459	233652	216738	220589	217602	224954	222045	216150	226299
3	220693	235910	207348	213663	232749	211414	225613	220153	217230	228017
4	221408	221132	231883	202322	221660	224586	218210	236109	225958	209532
5	229613	230677	216676	208816	218987	227789	222286	209797	224980	223169
10	223938	226105	228570	212664	238638	237052	239038	221581	224065	161149
20	218012	220428	227221	226652	217730	214161	234695	224089	228030	201802
50	214143	227381	226401	226689	219107	234506	214944	224862	229617	195180
100	220492	227519	235501	208288	236723	212586	237951	218239	232281	183320
200	233906	230204	219750	211532	214651	211847	236078	224261	222884	207897
500	240543	215164	208914	211648	226278	215497	235211	232645	217328	210372
1000	222074	220776	208685	229246	230503	222228	221576	225596	221948	211058

Fonte – Elaborada pelo autor.

Tabela E3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	100,00	100,00	0,00	100,00	100,00	100,00	100,00	100,00
2	100,00	100,00	95,00	95,00	95,00	100,00	195,00	100,00	195,00	95,00
3	195,00	285,25	285,25	95,00	100,00	95,00	100,00	190,25	95,00	285,25
4	370,99	370,99	95,00	275,99	185,25	285,25	270,99	180,74	285,25	195,00
5	352,44	257,44	257,44	285,25	190,25	270,99	266,70	352,44	366,70	370,99
10	418,78	501,16	358,12	499,65	324,79	385,61	423,78	518,20	495,78	412,16
20	450,77	406,63	483,20	503,33	480,25	496,71	487,91	502,40	467,85	499,65
50	458,55	456,58	487,91	481,01	458,69	505,97	503,33	479,47	518,20	525,95
100	420,95	501,46	508,33	513,08	499,65	459,73	480,91	513,08	481,98	469,48
200	454,12	529,82	513,91	472,42	505,91	467,02	465,95	513,08	454,25	480,04
500	521,87	503,77	503,33	412,66	500,78	483,07	486,14	475,91	486,10	529,82
1000	529,82	463,35	510,42	503,33	501,46	415,20	529,82	476,10	446,26	489,69

Fonte – Elaborada pelo autor.

Tabela E4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	53539	52877	56511	59193	58054	57826	59474	56238	55195	49143
2	53169	54581	51387	54336	59154	51551	56247	50096	53119	63410
3	59156	53664	56309	54230	54499	56076	54891	54320	53854	54351
4	56859	58710	59389	53786	58408	58535	50602	52277	55277	49007
5	52275	55373	53914	56106	54734	52724	53363	58364	51602	63295
10	55893	51855	53898	53207	58401	55317	50880	51279	59505	61515
20	54632	55462	53929	53639	52791	57235	56519	54228	57904	51861
50	50389	50974	58261	55940	53793	56621	56492	55270	57273	58587
100	56648	54982	55798	51841	56766	53102	53280	53152	55317	55464
200	54749	52816	55797	58455	55180	54781	54455	53069	58000	60548
500	54305	52979	55788	58099	54531	52948	59887	54234	54944	49985
1000	58547	56535	56386	51206	52357	55524	54752	51666	58392	59435

Fonte – Elaborada pelo autor.

Tabela E5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	0,00	100,00	100,00	100,00	100,00	100,00
2	95,00	195,00	195,00	195,00	100,00	95,00	195,00	195,00	100,00	195,00
3	185,25	195,00	100,00	285,25	195,00	185,25	90,25	285,25	285,25	190,25
4	370,99	180,74	270,99	370,99	185,74	95,00	270,99	275,99	275,99	185,74
5	276,45	266,70	366,70	276,45	452,44	452,44	275,99	276,45	366,70	266,70
10	513,91	358,12	522,27	314,31	419,71	497,84	500,78	467,12	432,46	500,78
20	521,87	487,91	450,81	510,42	461,70	503,85	473,45	508,33	387,66	455,10
50	510,04	529,82	510,42	505,91	418,88	482,91	506,55	529,82	490,36	510,04
100	510,84	474,99	529,82	480,08	402,05	529,82	508,33	525,95	464,46	457,27
200	479,94	496,16	478,07	458,76	488,93	489,90	518,20	491,71	479,29	433,46
500	505,97	481,10	510,84	455,45	509,40	525,95	445,72	451,87	513,91	420,87
1000	492,00	485,61	500,78	475,74	485,90	500,96	505,91	491,71	503,33	453,42

Fonte – Elaborada pelo autor.

Tabela E6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	213371	217536	199081	225754	201628	206319	210058	209363	218175	255125
2	222289	205503	222314	214675	225536	219109	223078	229373	222493	172010
3	208015	222178	220930	213060	217968	197209	224793	199117	218693	234337
4	227717	218403	231340	215565	224222	213638	209809	220974	217967	176935
5	217571	203861	225239	215068	222177	219678	222493	210261	214908	205304
10	215935	223186	212442	222438	229624	217490	228880	224677	203832	177946
20	212884	216039	213341	214308	222742	222346	218962	215942	201083	218783
50	222823	208690	215445	227441	201869	213843	203711	198671	211905	252122
100	208600	203080	220164	204725	214972	207334	205073	209722	217651	265079
200	195494	226111	231659	218507	214071	207173	212228	213638	202997	234572
500	207251	217638	220186	211689	219736	208786	226897	213528	216703	214876
1000	208527	219877	211715	198690	224187	202163	229266	218686	210168	233761

Fonte – Elaborada pelo autor.

Tabela E7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	0,00	0,00	0,00	100,00	100,00	100,00	0,00	100,00
2	195,00	195,00	195,00	95,00	195,00	195,00	95,00	195,00	100,00	195,00
3	185,25	90,25	190,25	185,25	285,25	190,25	195,00	190,25	285,25	185,25
4	370,99	280,74	370,99	285,25	180,74	100,00	370,99	370,99	370,99	180,74
5	171,70	357,44	452,44	357,44	257,44	275,99	452,44	285,25	352,44	81,45
10	495,78	503,33	508,33	529,82	525,95	513,91	525,95	426,87	362,63	407,59
20	521,87	456,58	493,97	478,65	513,08	431,15	428,85	522,27	497,84	340,37
50	525,95	468,60	485,61	480,54	529,82	488,93	481,39	472,61	514,33	529,82
100	488,22	480,74	525,95	495,78	529,82	489,90	518,20	471,92	456,64	505,53
200	487,15	529,82	495,78	521,87	463,83	459,27	495,78	488,09	421,97	529,82
500	497,20	502,59	502,48	478,16	482,06	515,46	521,87	436,68	493,61	483,93
1000	503,95	465,40	448,19	444,97	453,78	471,62	510,42	409,20	517,59	521,87

Fonte – Elaborada pelo autor.

Tabela E8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	222702	218417	227015	209622	224677	207205	225809	213889	214505	219889
2	221195	217759	217636	225465	208996	206857	219107	214622	230838	221295
3	220116	209410	234449	217292	228336	206560	216437	234990	213340	202880
4	228887	222900	219289	225730	235338	208688	210850	212796	221623	197759
5	223508	210444	224506	214881	207823	212881	217420	214931	202778	254828
10	218739	215146	234852	221073	221528	224497	198833	204903	234544	210095
20	223658	227580	215179	237748	227701	212044	228062	230045	229139	153434
50	208110	226520	210134	223659	215826	226964	226284	218837	206024	223622
100	221448	228249	212206	211289	226127	224046	219361	209553	211580	224291
200	211123	226933	238667	210919	206841	217839	220257	224722	222026	213353
500	228237	218105	221844	222282	219324	233444	223416	231420	206378	202430
1000	208560	216136	210726	217710	220475	222882	222633	213851	213548	284819

Fonte – Elaborada pelo autor.

Tabela E9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	0,00	0,00	0,00	100,00	100,00	100,00	100,00	100,00	100,00
2	95,00	100,00	195,00	95,00	195,00	195,00	195,00	100,00	195,00	195,00
3	285,25	190,25	195,00	190,25	285,25	285,25	95,00	185,25	195,00	190,25
4	185,74	90,25	285,25	195,00	195,00	190,25	275,99	190,25	185,25	370,99
5	366,70	271,70	362,19	195,00	175,99	280,74	262,19	285,25	366,70	285,25
10	440,21	249,08	487,42	518,78	395,00	425,10	436,14	431,62	382,73	529,82
20	452,72	502,04	462,84	369,15	513,91	525,95	474,47	477,62	475,66	470,29
50	447,89	521,87	487,15	459,36	460,65	490,57	472,82	466,08	464,04	502,04
100	499,56	483,61	514,71	493,41	419,73	443,62	434,58	459,63	505,91	493,22
200	529,82	465,62	489,92	483,76	521,87	505,97	494,69	494,25	415,95	460,83
500	476,73	483,20	459,05	483,20	398,62	495,49	477,36	507,16	495,49	529,82
1000	452,00	452,31	476,99	495,49	472,76	476,39	461,38	495,49	495,49	495,49

Fonte – Elaborada pelo autor.

Tabela E10 – Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	262685	251217	244672	243575	251244	252882	238912	233420	243883	252230
2	248239	264497	254939	250693	253988	249235	252830	248260	254318	198101
3	258086	247147	239477	258997	249352	229628	247255	249433	256199	239616
4	242223	239338	252350	257002	257517	237803	242290	257439	245562	243826
5	245638	238265	241250	246883	228570	257286	242936	234249	253809	286764
10	254799	247987	251764	267859	255323	251775	249316	234737	238656	224474
20	245530	244807	257242	233751	256183	230508	232562	246854	258680	272823
50	249615	249523	246310	253516	240222	230838	229284	239928	238642	307252
100	251721	254658	245073	251529	264354	254756	252404	246010	227331	247564
200	234644	248157	257452	260429	248223	242480	254851	247018	256019	266877
500	267407	235654	271490	258335	241003	278561	246383	242544	265149	277484
1000	255913	247187	270738	287225	256433	272136	267995	264724	258893	311786

Fonte – Elaborada pelo autor.

Tabela E11 – Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	0,00	0,00	100,00	100,00	100,00	100,00	100,00	0,00	100,00
2	95,00	195,00	0,00	100,00	195,00	0,00	195,00	95,00	195,00	100,00
3	285,25	185,25	285,25	285,25	195,00	100,00	100,00	100,00	285,25	0,00
4	185,74	270,99	85,74	85,74	370,99	285,25	190,25	275,99	185,25	185,25
5	370,99	195,00	352,44	276,45	276,45	275,99	370,99	181,45	270,99	352,44
10	271,70	497,97	436,14	366,70	345,21	275,99	276,45	351,59	253,83	410,97
20	339,48	370,99	483,45	414,28	386,74	403,85	366,04	341,69	490,82	306,65
50	410,10	456,81	445,22	396,55	399,39	472,28	448,42	389,59	396,37	429,54
100	432,71	440,37	484,54	420,22	444,54	419,39	413,49	482,47	486,15	472,98
200	395,20	414,90	440,83	476,21	499,70	480,25	359,40	448,91	485,85	408,87
500	417,83	421,93	449,30	391,74	423,55	412,77	423,78	411,59	466,35	408,15
1000	462,80	452,57	441,76	405,49	411,92	391,46	430,19	416,76	489,69	417,03

Fonte – Elaborada pelo autor.

Tabela E12 – Tempo de Execução da Política Aleatória, por Horizonte e Amostra, para a instância 4.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	248456	241870	244176	236399	244436	247347	244485	227053	251858	268220
2	248855	239738	243042	243054	246675	236358	242901	251348	256686	245543
3	248203	247415	239286	256218	235047	234088	257914	261200	239017	235812
4	239957	248791	244109	243705	244522	242280	249590	240444	253626	247176
5	245587	229821	224022	263296	244111	259789	241145	258308	246218	241913
10	233242	236027	244184	248219	253145	233315	240165	250192	227919	287792
20	232357	228558	254644	252132	231662	262145	237845	229979	253324	271554
50	262089	245108	240755	243032	244756	247769	250306	247527	251091	221767
100	235515	255720	225763	239798	244334	244041	239051	239251	247351	283376
200	244131	227134	243181	246047	244088	243418	226465	249496	236391	293899
500	243836	241677	247904	252696	238746	259497	245438	234854	251135	238467
1000	254044	253989	229215	242592	243936	243342	236440	243984	237004	269724

Fonte – Elaborada pelo autor.

APÊNDICE F – Dados da Instância 5

Tabela F1 – Recompensa Descontada do Algoritmo QMDP, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	100,00	0,00	100,00	100,00	100,00	100,00
2	95,00	95,00	195,00	195,00	195,00	195,00	195,00	195,00	95,00	195,00
3	285,25	195,00	285,25	285,25	195,00	285,25	285,25	195,00	185,25	285,25
4	213,11	221,38	311,23	331,36	220,53	322,84	227,23	329,92	306,35	302,78
5	329,49	521,22	401,89	520,28	412,38	218,94	213,43	388,52	207,44	434,29
10	616,13	544,47	453,26	460,00	577,88	593,70	604,70	526,37	574,62	476,49
20	561,38	561,63	595,25	530,82	564,15	505,29	574,71	508,80	597,55	560,69
50	583,79	583,06	447,13	587,59	564,21	595,23	599,30	593,05	478,17	612,85
100	600,09	560,08	589,16	568,04	512,99	624,71	573,88	526,89	583,09	565,93
200	584,62	567,49	571,14	612,70	576,73	568,68	633,59	609,83	569,02	604,11
500	548,64	421,01	572,19	583,10	556,72	621,94	592,62	591,25	591,15	592,20
1000	555,81	536,93	592,62	555,66	573,70	554,13	587,28	529,60	543,80	441,91

Fonte – Elaborada pelo autor.

Tabela F2 – Tempo de Execução do Algoritmo QMDP, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	14109699	13729011	14304543	14236067	14369837	13456498	14243479	13816343	13695743	12848170
2	12834154	13507455	14005924	14308765	13632254	12529524	13744846	13461561	14120653	14755254
3	12583908	13762872	13388826	14015765	12781876	14045284	13649046	13435517	13788528	15138248
4	14213120	14609558	14833075	14169739	13260013	13262480	13220710	14099494	13843091	12271570
5	13353101	12939361	13820172	12917192	13820086	14232795	13401293	14225125	13453477	14439388
10	13652914	15182534	13395564	14119925	13878757	14879416	13543084	14464289	14562614	11612333
20	13756320	13778978	13456772	12962929	13083701	13923729	13212908	13778885	13419845	13982203
50	12921603	13778838	14315499	13370328	13778245	13949268	12938512	13690996	13235957	13737664
100	13995611	12890583	14295071	14160167	13461458	13566883	12472233	13848320	13815707	13874787
200	13715324	13977563	14344820	14396033	14253743	14233516	14226231	14360585	13980255	11867830
500	13306293	14285320	14371586	14493176	13946460	14259876	13526555	14346253	13148043	11128788
1000	14788895	13491187	14470763	13317538	13397897	13833568	14303156	12964895	13814084	11483877

Fonte – Elaborada pelo autor.

Tabela F3 – Recompensa Descontada do Algoritmo FIB, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	0,00	100,00	100,00	100,00	100,00	100,00
2	95,00	195,00	95,00	195,00	95,00	95,00	195,00	95,00	195,00	95,00
3	233,11	317,41	318,20	113,16	110,35	107,38	115,84	210,98	112,74	325,04
4	413,27	421,31	108,19	330,00	221,05	325,35	302,75	201,40	341,20	227,51
5	405,18	285,13	294,30	334,64	217,29	325,08	304,97	414,82	429,37	419,78
10	483,42	582,93	421,50	589,42	365,71	443,42	507,97	594,54	578,71	465,54
20	504,14	448,47	539,19	567,04	548,09	583,97	555,69	584,35	553,78	594,89
50	538,96	528,68	545,98	546,42	511,31	562,74	599,86	574,14	575,25	611,26
100	474,17	554,97	585,42	573,11	552,80	523,97	531,69	587,11	533,82	537,06
200	502,03	609,98	615,51	546,02	605,20	516,31	525,93	614,44	533,98	554,55
500	600,84	601,96	577,81	491,75	589,22	569,85	574,76	561,68	571,76	614,77
1000	604,78	555,01	587,63	603,28	557,83	477,62	599,08	548,43	502,33	555,40

Fonte – Elaborada pelo autor.

Tabela F4 – Tempo de Execução do Algoritmo FIB, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1173238	1310997	1165995	1208377	1300470	1332478	1278562	1203415	1307208	1054090
2	1181813	1232710	1272020	1163817	1156692	1270670	1266685	1129736	1322845	1331242
3	1218558	1267597	1166605	1147400	1233202	1116670	1254386	1268689	1229313	1428390
4	1193057	1206464	1288292	1230321	1148507	1241872	1144537	1211883	1194932	1471845
5	1130961	1273694	1266789	1234895	1175986	1293179	1262911	1177882	1202381	1312372
10	1225094	1310186	1274601	1257351	1172470	1302800	1149038	1296339	1152711	1190460
20	1188550	1250921	1143054	1223784	1180267	1170137	1278364	1315746	1209230	1368867
50	1159488	1166068	1276469	1270537	1247707	1273457	1288694	1275680	1263916	1110144
100	1307956	1265123	1239396	1132861	1255980	1215303	1279029	1323509	1223701	1084952
200	1153937	1279874	1244669	1235098	1216008	1138417	1245231	1204866	1250130	1366480
500	1314831	1212128	1221529	1206703	1174371	1153953	1253838	1259728	1336704	1194835
1000	1228873	1260654	1278536	1244918	1297357	1285802	1189140	1227196	1145352	1175052

Fonte – Elaborada pelo autor.

Tabela F5 – Recompensa Descontada do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
2	195,00	195,00	195,00	195,00	195,00	195,00	195,00	195,00	95,00	95,00
3	204,71	231,34	116,29	329,26	219,38	222,17	101,41	327,64	336,98	225,28
4	432,42	211,34	315,73	429,36	215,66	111,81	315,41	327,39	322,74	205,60
5	304,81	306,97	406,06	319,27	517,29	501,51	326,36	324,41	412,07	318,24
10	569,89	413,93	575,31	370,43	501,11	573,78	555,12	536,15	501,16	552,32
20	604,58	584,87	515,28	604,51	515,12	599,60	525,61	560,69	445,33	527,69
50	581,27	619,68	574,53	581,50	491,38	569,45	594,88	595,43	560,73	572,70
100	571,72	538,19	607,24	528,48	480,76	614,67	562,12	618,17	516,89	542,93
200	528,06	580,50	541,02	536,34	549,62	584,14	571,24	561,77	548,64	513,79
500	596,88	575,59	574,56	542,14	594,04	619,14	498,72	533,23	568,57	498,42
1000	577,34	561,71	554,07	570,42	540,28	564,34	604,65	565,03	557,69	540,82

Fonte – Elaborada pelo autor.

Tabela F6 – Tempo de Execução do Algoritmo SARSOP, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	10030228	10294769	9760454	9019396	9363400	9954001	9695977	9423853	10077586	11672626
2	9324015	9316301	9569520	9692150	10558886	8923576	9432484	9378663	10100653	12423092
3	9503692	8977299	9405382	10060369	9957715	10340789	9624771	9958094	10383930	10069469
4	10098624	9799224	10038506	9950510	10504362	9218111	10546861	10224786	9873665	9833771
5	9541602	9593167	10252987	10894736	10045000	10259512	9478869	9582939	9192080	10369308
10	9536865	10372825	10272926	10596818	9920542	9756081	8957350	9415647	10138051	9737475
20	10473016	10114055	10037223	10217167	9989736	9990745	9724158	9835515	9741420	8819625
50	10392808	9949695	9675114	9022115	10394840	9580632	9433569	9580591	10078774	8967852
100	9625346	9658293	10337676	9970797	9884136	10001825	9282754	9883069	9953758	10355906
200	10125868	10167057	9596954	10129928	10269729	9928115	10111550	10281761	10197735	9438283
500	9828423	9307696	10374811	10083366	9906007	10092393	9417368	9721703	10276820	9341323
1000	10599945	9845311	10029193	9679329	9894800	9726207	9990162	9796262	9437159	10113832

Fonte – Elaborada pelo autor.

Tabela F7 – Recompensa Descontada do Algoritmo POMCP, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	100,00	0,00	100,00	0,00	100,00	100,00	100,00	0,00	100,00
2	195,00	195,00	195,00	95,00	195,00	195,00	195,00	195,00	95,00	195,00
3	185,25	90,25	190,25	285,25	285,25	190,25	285,25	190,25	285,25	285,25
4	370,99	280,74	370,99	285,25	180,74	370,99	370,99	370,99	370,99	370,99
5	275,99	357,44	452,44	357,44	257,44	275,99	452,44	452,44	352,44	357,44
10	529,82	503,33	603,33	603,33	603,33	603,33	603,33	503,33	503,33	503,33
20	603,33	505,48	555,11	550,69	577,67	476,09	499,85	587,92	580,07	408,29
50	596,53	521,85	546,53	552,12	619,26	539,06	561,46	552,22	599,82	633,21
100	543,02	534,36	580,06	577,67	623,27	541,01	588,91	531,27	536,30	579,65
200	554,12	620,51	556,73	610,55	516,92	517,80	562,29	541,79	474,73	625,36
500	577,36	594,21	595,22	539,56	549,82	593,51	581,08	513,05	567,99	558,45
1000	574,02	533,92	495,40	512,02	512,40	564,70	609,17	485,84	575,03	622,40

Fonte – Elaborada pelo autor.

Tabela F8 – Tempo de Execução do Algoritmo POMCP, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	11122088	10465738	10109314	10451958	9908111	10430245	10295114	10970519	10274669	11840504
2	9727134	9981644	10407040	10972539	9871003	9775013	10516921	10345452	10294566	11018358
3	10501770	10094098	11129834	9925596	10862190	9992769	10919003	10481843	10466259	10186958
4	10702219	11313713	10587987	10052074	10386538	10444042	10339049	11053316	9803007	10929005
5	10737832	11186201	10613898	10646702	10363237	10980808	10485017	10369959	11136933	9697013
10	10771921	10432849	10075092	10397023	10660344	10166663	10764888	10828396	10530668	11409136
20	10342022	10358062	10824267	10705648	11327350	10494871	10903139	9868154	10231378	11907489
50	11065910	10903874	9939033	10566805	9771682	10285960	10826242	10928568	9832871	12676695
100	11643383	9974404	10047135	11297702	11516007	10475175	10027536	11285448	10588476	9261504
200	9968026	10562052	11210669	11722392	11202105	11660661	10376427	10190340	11158339	9257139
500	11021900	10748172	11486527	11119737	10582359	10169764	9881260	10646171	10171702	9530998
1000	10230346	10384991	11339142	10264934	10298688	10407270	10188608	10897185	10984610	12046366

Fonte – Elaborada pelo autor.

Tabela F9 – Recompensa Descontada do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	100,00	0,00	100,00	0,00	100,00	100,00	100,00	100,00	100,00	100,00
2	195,00	195,00	195,00	195,00	195,00	195,00	195,00	195,00	95,00	195,00
3	185,25	190,25	190,25	285,25	285,25	285,25	285,25	190,25	285,25	285,25
4	210,02	99,89	333,99	226,54	225,97	313,02	313,02	226,98	313,02	430,07
5	421,28	318,66	425,90	227,95	207,58	332,93	313,22	320,66	407,50	318,00
10	513,59	294,17	557,02	572,72	447,27	476,58	496,18	508,69	448,92	634,38
20	514,51	588,33	522,55	420,44	567,91	629,44	549,05	565,30	561,60	534,74
50	524,89	622,72	567,72	540,43	527,43	585,53	536,23	517,35	552,71	559,66
100	554,24	547,47	582,45	587,98	491,40	520,10	483,82	518,31	601,62	581,50
200	603,52	517,77	552,26	540,50	578,68	575,18	549,54	568,79	485,59	550,89
500	545,28	579,32	550,42	539,84	474,55	528,59	539,32	583,86	504,87	621,38
1000	517,51	536,92	544,07	586,49	551,41	552,89	511,57	586,49	502,31	586,49

Fonte – Elaborada pelo autor.

Tabela F10 – Tempo de Execução do Algoritmo POMCPOW, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	13130179	13268774	13873288	14438865	13909511	13993299	13923179	14652248	13120387	13521680
2	13115864	13850848	13364138	13999924	13389933	13837605	13838780	13929128	13598829	13510371
3	13103957	13051594	13871314	13972176	13059175	14783092	13927009	14565265	13580912	14112046
4	12828713	14510063	14212601	14461612	12988830	13039035	13371146	14655945	13494569	13804886
5	13498159	13939542	13037469	13466149	14413657	14668751	13055029	13973663	13814758	14006813
10	13152220	12364208	13446321	13484237	12644303	14512847	13360440	13793298	13211420	14403866
20	14097979	14216128	14431657	13674631	14029322	13799429	13531224	12787944	13491664	11837972
50	14065777	13999896	13664178	14052334	13881275	13372909	13860798	13560953	14392212	13194748
100	14289608	12803537	13713403	13569277	14191713	13757269	13956639	13542406	12399859	14178339
200	12870177	14499323	14443734	14062724	14250875	13107635	14770856	13603734	14584638	13242444
500	14097293	14169676	13820852	14667822	15076565	14195954	14254535	13634673	15074873	14320917
1000	15077286	14428846	13663965	15384322	14955851	14856093	15017503	14550773	15023268	15974543

Fonte – Elaborada pelo autor.

Tabela F11 – Recompensa Descontada da Política Aleatória, por Horizonte e Amostra, para a instância 5.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,00	0,00	0,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
2	95,00	95,00	0,00	195,00	195,00	195,00	95,00	195,00	95,00	195,00
3	185,25	195,00	185,25	285,25	195,00	185,25	285,25	195,00	185,25	185,25
4	195,00	285,25	95,00	95,00	370,99	341,40	195,00	370,99	195,00	270,99
5	452,44	219,42	404,91	308,93	325,82	325,00	341,69	213,17	301,45	342,16
10	311,46	554,58	502,49	433,40	385,85	305,54	328,37	388,38	298,07	467,36
20	387,29	410,89	565,80	491,18	431,80	449,12	439,24	386,84	571,71	346,05
50	484,32	505,86	502,38	450,68	463,96	520,19	518,49	452,12	443,52	473,95
100	516,20	512,51	502,29	477,37	493,75	494,86	491,81	502,06	531,89	553,33
200	448,81	489,70	512,40	559,15	529,34	558,17	421,55	532,49	549,25	454,89
500	470,98	504,44	527,72	452,13	475,63	490,92	480,68	477,09	530,12	480,77
1000	533,76	507,70	503,75	481,88	487,43	440,68	513,28	491,19	551,17	471,46

Fonte – Elaborada pelo autor.

Tabela F12 – Tempo de Execução da Política Aleatória por Horizonte e Amostra.

Horizonte	**Amostra**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	8942295	8736281	8821031	8993076	8722660	8698939	7933366	8672161	9068748	9037913
2	9354803	9227145	8681337	8927046	8411856	9304133	8711154	8876386	8886327	7246243
3	8574321	8802238	8889012	9506807	9055651	8794732	8486413	8897348	9366814	7253094
4	8738442	9369915	9420521	9120686	9125138	8473587	8255101	8695555	8489167	7938328
5	8589374	9036363	8458865	8850422	9283759	8094104	8923735	9370727	8752028	8267083
10	8127048	9101203	8800957	9563918	9361901	8927581	9153756	8183373	9040668	7366045
20	9170283	9116837	8700866	8573791	8867255	8957706	9307211	8301313	8969936	7661282
50	8985665	8634472	8764578	9025138	8410759	9005304	8641597	9168004	8780187	8210666
100	8623445	8944231	8490293	9236986	9033657	8580595	8612168	9244199	8840855	8019951
200	9085573	8575281	8925217	9209872	8599219	9160282	8857973	8490430	9017700	7704903
500	8875995	9493760	9191453	8710354	8468546	8495874	8366541	8700152	9136176	8187629
1000	8469277	8850763	8886184	8566840	8328244	8713747	9065299	8723447	8419436	9603253

Fonte – Elaborada pelo autor.

APÊNDICE G – Teste de Normalidade para Amostras

Tabela G1 – Teste de Normalidade de Amostra do Algoritmo QMDP.

Amostra	Acumulado	Esperado	Rank	Transformar	Normais	Diferença 1	Diferença 2
244,57	1	0,1	0,0	-1,28155	0,02248	0,07752	0,02248
251,28	2	0,2	0,1	-0,84162	0,06136	0,13864	0,03864
272,38	3	0,3	0,2	-0,52440	0,46328	0,16328	0,26328
275,99	4	0,4	0,3	-0,25335	0,56204	0,16204	0,26204
275,99	5	0,5	0,4	0,00000	0,56204	0,06204	0,16204
275,99	6	0,6	0,5	0,25335	0,56204	0,03796	0,06204
285,25	7	0,7	0,6	0,52440	0,78613	0,08613	0,18613
285,25	8	0,8	0,7	0,84162	0,78613	0,01387	0,08613
285,25	9	0,9	0,8	1,28155	0,78613	0,11387	0,01387
285,25	10	1,0	0,9		0,78613	0,21387	0,11387
Kolmogorov-Smirnov Calculado							**0,26328**
Kolmogorov-Smirnov Tabelado (5%)							**0,40925**
Conclusão: não se tem evidências para rejeitar a hipótese nula, e a amostra tem distribuição aproximadamente normal.							

Tabela G2 – Teste de Normalidade de Amostra do Algoritmo POMCP.

Amostra	Acumulado	Esperado	Rank	Transformar	Normais	Diferença 1	Diferença 2
100,00	1	0,1	0,0	-1,28155	0,06930	0,03070	0,06930
100,00	2	0,2	0,1	-0,84162	0,06930	0,13070	0,03070
185,25	3	0,3	0,2	-0,52440	0,37276	0,07276	0,17276
185,25	4	0,4	0,3	-0,25335	0,37276	0,02724	0,07276
185,25	5	0,5	0,4	0,00000	0,37276	0,12724	0,02724
195,00	6	0,6	0,5	0,25335	0,42376	0,17624	0,07624
285,25	7	0,7	0,6	0,52440	0,84896	0,14896	0,24896
285,25	8	0,8	0,7	0,84162	0,84896	0,04896	0,14896
285,25	9	0,9	0,8	1,28155	0,84896	0,05104	0,04896
285,25	10	1,0	0,9		0,84896	0,15104	0,05104
Kolmogorov-Smirnov Calculado							**0,24896**
Kolmogorov-Smirnov Tabelado (5%)							**0,40925**
Conclusão: não se tem evidências para rejeitar a hipótese nula, e a amostra tem distribuição aproximadamente normal.							

Fonte – Elaborada pelo autor.