Estimação de Parâmetros do Aprendizado por Reforço para o Problema de Planejamento de Rotas com Reabastecimento

André Luiz C. Ottoni [*] **Erivelton G. Nepomuceno** [**]
Marcos S. de Oliveira [***]

[*] *Centro de Ciências Exatas e Tecnológicas, Universidade Federal do Recôncavo da Bahia, Cruz das Almas, BA, Brasil (e-mail: andre.ottoni@ufrb.edu.br).*
[**] *Grupo de Controle e Modelagem, Departamento de Engenharia Elétrica, Universidade Federal de São João del-Rei, São João del-Rei, MG, Brasil (e-mail: nepomuceno@ufsj.edu.br)*
[***] *Departamento de Matemática e Estatística, Universidade Federal de São João del-Rei, São João del-Rei, MG, Brasil (e-mail: mso@ufsj.edu.br)*

Abstract: Path planning is a important problem in mobile robotics. One of the aspects of this type of autonomous vehicles planning refers to observe the fuel-constraints. In this sense, the objective of this work is to estimate the Reinforcement Learning parameters for the path planning problem with refueling. The results indicate that the parameters estimated with the Response Surface Methodology reached the best solutions in most of the experiments.

Resumo: O planejamento de rotas é um importante problema na robótica móvel. Uma das vertentes desse tipo de planejamento para veículos autônomos, refere-se a observar as restrições operacionais com combustível. Nesse sentido, o objetivo deste trabalho é estimar os parâmetros do Aprendizado por Reforço para o problema planejamento de rotas com reabastecimento. Os resultados apontam que os parâmetros estimados com a Metodologia de Superfície de Resposta alcançaram as melhores soluções na maioria dos experimentos.

Keywords: Reinforcement learning; Path planning; Refueling; Response surface methodology; Parameter estimation.

Palavras-chaves: Aprendizado por reforço; Planejamento de rotas; Reabastecimento; Metodologia de superfície de resposta; Estimação de parâmetros.

1. INTRODUÇÃO

O planejamento de rotas (*path planning*) é um importante campo da robótica móvel (Sipahioglu et al., 2008; Adorno and Borges, 2014; Macharet and Campos, 2018). O problema consiste em definir uma sequência de movimentos para que o agente autônomo saia de uma configuração inicial e alcance uma posição final (objetivo) (Adorno and Borges, 2014).

Uma área de pesquisa do planejamento de rotas para veículos autonômos refere-se a observar as restrições operacionais com combustível (Levy et al., 2014; Yoo et al., 2016). Nesses casos, o desafio é definir uma rota que garanta que o veículo realize todo o percurso sem cessar o combustível. Seguindo essa mesma linha, os problemas de reabastecimento buscam otimizar o gasto com a compra de combustível em rotas rodoviárias (Khuller et al., 2007; Suzuki, 2012; Rodrigues Junior and Cruz, 2013).

Os problemas de reabastecimento podem ser classificados em quatro grupos (Khuller et al., 2007): reabastecimento com rota fixa, reabastecimento com rota variável, Problema do Caixeiro Viajante (PCV) com custo uniforme em cada ponto e PCV com o custo de combustível variando em cada localidade. A última classe pode ser aplicada para tratar o reabastecimento em malhas rodoviárias no Brasil, onde são encontradas variações de preços de combustíveis em cada cidade, conforme pode ser verificado no site Agência Nacional do Petróleo (ANP) [1] .

O PCV é um dos problemas de otimização combinatória mais conhecidos e frequentemente é considerado no planejamento de rotas de veículos autônomos (Yu et al., 2002; Sipahioglu et al., 2008; Giardini and Kalmár-Nagy, 2011; Macharet and Campos, 2018). Nesse aspecto, generalizações do PCV podem englobar aspectos diversos da robótica móvel, como restrições do veículo (Macharet and Campos, 2018), ambientes dinâmicos (Sipahioglu et al., 2008) e múltiplos veículos (Yu et al., 2002; Almeida et al., 2017).

[1] http://anp.gov.br/preco/

DOI: 10.17648/sbai-2019-111113

Uma técnica de Inteligência Artificial com relevantes aplicações no planejamento de rotas e no PCV é o Aprendizado por Reforço (AR) (Gambardella and Dorigo, 1995; Konar et al., 2013; Rakshit et al., 2013; Li et al., 2015; Ottoni et al., 2018). No AR, um agente aprende a partir de sucessos e fracassos interagindo em um ambiente (Sutton and Barto, 2018). Um dos principais aspectos do AR é a estimação de parâmetros que otimizem o aprendizado, como taxa de aprendizado (α) e o fator de desconto (γ) (Even-Dar and Mansour, 2003; Schweighofer and Doya, 2003; Ottoni et al., 2019). A definição dos parâmetros podem influenciar diretamente no aprendizado de uma boa rota (Ottoni et al., 2018). Nesse sentido, Ottoni et al. (2018) apresentam uma metodologia para a estimação de parâmetros do AR utilizando modelos de Superfície de Resposta (RSM) (Myers et al., 2009).

Dessa forma, o objetivo neste trabalho é estimar os parâmetros do AR para o problema de planejamento de rotas com reabastecimento adotando modelos RSM. Será proposta a resolução de rotas baseadas no PCV com custo uniforme e não-uniforme em cada cidade, baseando-se em dados reais da ANP. Os experimentos envolvem simulações com dois tradicionais algoritmos de AR: *Q-learning* (Watkins and Dayan, 1992) e SARSA (Sutton and Barto, 2018).

Este artigo está organizado em seções. Na seção 2, são definidos aspectos teóricos do AR. A seção 3, apresenta as características do problema de planejamento de rotas com reabastecimento. As seções 4 e 5, apresentam a metodologia e os resultados, respectivamente. Finalmente, na seção 6 são apresentadas as conclusões do trabalho.

2. APRENDIZADO POR REFORÇO

O Aprendizado por Reforço (AR) é fundamentado nos Processos de Decisão de Markov (PDM). No AR, um agente interage com um ambiente aprendendo a selecionar as ações que maximizem a recompensa ao longo do tempo (Sutton and Barto, 2018).

Os parâmetros do AR, taxa de aprendizado (α) e fator de desconto (γ) são adotados em vários algoritmos. Esses parâmetros podem ser definidos entre 0 e 1. A taxa de aprendizado é responsável por controlar a velocidade que as novas informações se sobrepõem sobre o aprendizado já adquirido. Já o fator de desconto descreve a preferência de um agente entre as recompensas atuais e futuras. Se $\gamma = 1$, as recompensas no futuro tem alta significância. Caso contrário, se $\gamma \neq 1$, as recompensas atuais são mais relevantes no instante t do que as recompensas posteriores (descontadas) (Sutton and Barto, 2018; Russell and Norving, 2013).

Neste trabalho, são adotados dois tradicionais algoritmos de AR: *Q-learning*(Watkins and Dayan, 1992) e o SARSA (Sutton and Barto, 2018). O *Q-learning* se baseia na atualização da matriz de aprendizado Q, a partir de (1):

$$Q_{t+1} = Q_t(s,a) + \alpha[r + \gamma max_{a'} Q(s',a') - Q_t(s,a)], \quad (1)$$

em que, s e a são respectivamente, estado e ação no instante atual (t); s' e a' são respectivamente, estado e ação no próximo instante ($t+1$); $Q_t(s,a)$ é o valor no instante t na matriz de aprendizado Q para o par estado $\times$ ação (s,a); Q_{t+1} é a atualização da matriz de aprendizado em $t+1$ pela execução da ação a no estado s; r é o reforço pela execução do par (s,a); $max_{a'} Q(s',a')$ é a utilidade de s', ou seja, o valor máximo na linha de Q referente ao novo estado; α é a taxa de aprendizado; γ é o fator de desconto. O Algoritmo 1 representa o *Q-learning*.

```
1.  Para cada (s,a) inicialize Q(s,a)=0;
2.  Observe  o estado s;
3.  Repita até o critério de parada ser satisfeito
4.      Selecione a ação a usando a política  e-greedy;
5.      Execute a ação a;
6.      Receba a recompensa imediata  R(s,a);
7.      Observe o novo estado s';
8.      Atualize o item Q(s,a) de acordo com (1);
9.      s = s';
10. Fim Repita
```

Algoritmo 1: *Q-learning.*

Já o método SARSA é uma modificação do *Q-learning*. A atualização da matriz de aprendizado no SARSA é dada por (2):

$$Q_{t+1} = Q_t(s,a) + \alpha[r(s,a) + \gamma Q_t(s',a') - Q_t(s,a)]. \quad (2)$$

O Algoritmo 2 representa o SARSA.

```
1.  Para cada (s,a) inicialize Q(s,a)=0;
2.  Observe  o estado s;
3.  Selecione a ação a usando a política  e-greedy;
4.  Repita até o critério de parada ser satisfeito
5.      Execute a ação a;
6.      Receba a recompensa imediata  R(s,a);
7.      Observe o novo estado s';
8.      Selecione a nova ação a' usando e-greedy;
9.      Atualize o item Q(s,a) de acordo com (2);
10.     s = s';
11.     a = a';
12. Fim Repita
```

Algoritmo 2: SARSA.

Nos Algoritmos 1 e 2 é adotada a política de seleção de ações $\epsilon - greedy$. Esse método utiliza o parâmetro ϵ no controle entre gula e aleatoriedade na tomada de decisão (Sutton and Barto, 2018).

3. PROBLEMA DE PLANEJAMENTO DE ROTAS COM REABASTECIMENTO

3.1 Descrição do Problema

O problema considerado neste trabalho consiste no planejamento de rotas em uma malha rodoviária para veículos autônomos. O agente móvel deve percorrer um conjunto de cidades e decidir os locais de reabastecimento de combustível, de forma a buscar minimizar o gasto final da rota. Para isso, o Problema do Caixeiro Viajante (PCV) é adotado sob duas circunstâncias: custo uniforme e não-uniforme (Khuller et al., 2007). No primeiro caso, o veículo deve visitar um conjunto de localidades e retornar a cidade inicial ao final do percurso, sendo que o preço do combustível não varia entre os postos da rota. Já no problema com custo não-uniforme, cada cidade da rota pode apresentar um valor de venda distinto para o litro do combustível.

Algumas restrições devem ser consideradas, como a capacidade do reservatório de combustível do veículo, quantidade

DOI: 10.17648/sbai-2019-111113

mínima de combustível para reabastecimento e garantia de completar todo o percurso (Rodrigues Junior and Cruz, 2013).

Além disso, neste trabalho é considerada a possibilidade da utilização de um guincho, caso o combustível no veículo cesse entre duas localidades. Nesse caso, também é associado um custo pela utilização do guincho no percurso.

3.2 Modelagem Matemática

A formulação matemática para o problema proposto, baseada em (Bodin et al., 1983; Suzuki, 2009; Rodrigues Junior, 2011), é apresentada de (3) à (11):

$$Min \sum_{i=1}^{N} \sum_{j=1}^{N} c_j l_j u_{ij} + g_{ij} z_{ij}, \tag{3}$$

sujeito a:

$$\sum_{i=1}^{N} u_{ij} = 1 \qquad j = 1, \ldots, N, \tag{4}$$

$$\sum_{j=1}^{N} u_{ij} = 1 \qquad i = 1, \ldots, N, \tag{5}$$

$$f_j + l_j \leq L_{max} u_{ij} \qquad i, j = 1, \ldots, N, \tag{6}$$

$$l_j = (L_{max} - f_j) w_{ij} \qquad i, j = 1, \ldots, N, \tag{7}$$

$$l_j \geq L_{min} w_{ij} \qquad i, j = 1, \ldots, N, \tag{8}$$

$$u_{ij}, w_{ij}, z_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, N, \tag{9}$$

$$\{c_j, f_j, g_{ij}, l_j, L_{max}, L_{min}\} \geq 0 \qquad i, j = 1, \ldots, N, \tag{10}$$

$$U = u_{ij} \in V \qquad i, j = 1, \ldots, N, \tag{11}$$

em que, N é um conjunto de nós. Já a variável de decisão $u_{i,j}$, assume 1 se o arco (i, j) compor a solução e 0 caso contrário. O custo com reabastecimento na cidade j é c_j e l_j é a quantidade de combustível reabastecido em j. O gasto com guincho em um arco (i, j) é representado por g_{ij} e z_{ij} é uma variável de decisão que recebe 1 somente se o guincho for utilizado entre as localidades i e j. Dessa forma, em (3) é retratado o objetivo de minimizar o custo total na rota dado pelo somatório dos gastos com reabastecimento e guincho. As restrições de (4) e (5) asseguram que cada localidade será visitada uma única vez. A Equação (6), por sua vez, garante que a quantidade de combustível no reservatório $(fj + l_j)$ não ultrapasse capacidade máxima (L_{max}), sendo que, f_j é o nível do reservatório no momento de chegada na cidade j. Já (7) garante que o veículo sempre completará o nível máximo do tanque ao reabastecer, onde $w_{ij} = 1$ se ocorre reabastecimento na localidade j. Além disso, em (8) restringe a quantidade mínima para o reabastecimento. Já (9) garante que a variáveis u_{ij}, y_j, z_{ij} são binárias e (11) que as demais variáveis são não negativas. Por fim, na restrição de (11), o conjunto V representa qualquer grupo de restrições que eliminem a formação de sub-rotas.

3.3 Instâncias

Neste trabalho são utilizadas quatro instâncias: Bahia30D, Minas24D, Minas30D e Minas57D. Cada uma delas envolve um conjunto de cidades dos estados de Minas Gerais e Bahia. As características das instâncias envolvem as distâncias entre as localidades, calculadas a partir das coordenadas (latitude e longitude) e também o preço médio de diesel (D) em cada cidade.

Os problemas Minas24 e Minas30 foram propostos inicialmente por Ottoni et al. (2016) levando em consideração o valor da gasolina em Dezembro/2015. Já neste trabalho, foi pesquisado no site da ANP (em Dezembro/2018) o preço médio do diesel em cada uma das localidades das instâncias. Em seguida, as cidades que compõem cada uma das instâncias, sendo a apresentação no formato cidade (preço médio do diesel em reais):

- Bahia30D: Alagoinhas (3,307), Barreiras (3,654), Brumado (3,646), Caetite (3,688), Camacari (3,358), Eunápolis (3,526), Feira de Santana (3,346), Guanambi (3,620), Ilhéus (3,761), Ipirá (3,304), Irecê (3,650), Itabuna (3,599), Itamaraju (3,520), Jacobina (3,592), Jaguaquara (3,336), Jequié (3,597), Juazeiro (3,668), Lauro de Freitas (3,270), Livramento de Nossa Senhora (3,721), Paulo Afonso (3,683), Poções (3,380), Porto Seguro (4,067), Salvador (3,399), Santo Antônio de Jesus (3,340), Senhor do Bonfim (3,481), Serrinha (3,443), Simões Filho (3,367), Teixeira de Freitas (3,545), Valença (3,532) e Vitória da Conquista (3,291).
- Minas24D: Araguari (3,321), Belo Horizonte (3,471), Betim (3,408), Campo Belo (3,433), Contagem (3,393), Formiga (3,418), Governador Valadares (3,366), Guaxupé (3,446), Itabira (3,476), Ituiutaba (3,437), Juiz de Fora (3,307), Monte Carmelo (3,428), Montes Claros (3,458), Oliveira (3,361), Patos de Minas (3,526), Poços de Caldas (3,613), Pouso Alegre (3,453), Sete Lagoas (3,238), Teófilo Otoni (3,443), Três Corações (3,735), Uberaba (3,51), Uberlândia (3,476), Unaí (3,486) e Varginha (3,511).
- Minas30D: Localidades de Minas24 mais Araxá (3,399), Barbacena (3,475), Divinópolis (3,507), Ipatinga (3,483), Lavras (3,774) e Passos (3,657).
- Minas57D: Localidades de Minas30 mais Alfenas (3,624), Bom Despacho (3,249), Caratinga (3,429), Congonhas (3,557), Conselheiro Lafaeite (3,629), Coronel Fabriciano (3,668), Curvelo (3,288), Frutal (3,583), Itajubá (3,456), Itaúna (3,444), Janaúba (3,586), Januária (3,726), João Monlevade (3,421), João Pinheiro (3,533), Leopoldina (3,287), Manhuaçu (3,422), Muriaé (3,458), Nova Lima (3,724), Ouro Preto (3,72), Pará de Minas (3,526), Paracatu (3,656), Patrocínio (3,608), Sabará (3,532), São João del-Rei (3,712), São Sebastião do Paraíso (3,529), Timóteo (3,459) e Ubá (3,545).

Em todas as instâncias, o custo pela utilização do guincho foi fixado em R$ 200,00 ($g_{ij} = 200$).

3.4 Características do Veículo

Neste trabalho, buscou-se representar o planejamento de rotas para um veículo de carga autônomo de porte pe-

DOI: 10.17648/sbai-2019-111113

queno. O veículo simulado possui características de um caminhão do tipo 3/4 (três quartos): capacidade do tanque de 150 litros ($L_{max} = 150$) e consumo médio de diesel de 7 km/litro. Alguns exemplos de caminhões neste estilo são: Volkswagem 8150 e Ford Cargo 816.

Além disso, foi definido que o veículo apenas seria reabastecido quando o nível do tanque estivesse com menos de 25% da capacidade, ou seja, a quantidade mínima de reabastecimento equivale à 75% da capacidade do reservatório ($L_{min} = 0,75 \times L_{max}$).

4. METODOLOGIA

4.1 Sistema de Aprendizado por Reforço

O modelo de AR definido para a resolução do problema de planejamento de rotas com reabastecimento é composto por um conjunto de estados (S), ações (A) e reforços (R). A formulação adotada é baseada em trabalhos anteriores: (Bianchi et al., 2009; Lima Júnior et al., 2010; Ottoni et al., 2018). Após analisar a lógica do problema abordado, é proposta a seguinte estrutura:

- Estados: são as localidades que o agente (caixeiro viajante) deve visitar para realizar a rota.
- Ações: cada ação representa a intenção de movimentação para outra localidade (estado) do problema. Além disso, a ação de reabastecimento é realizada sempre que o veículo chegue em uma localidade com o nível do tanque menor do que 25% da capacidade total ($0,25 \times N_{max}$).
- Reforços: a função de reforço foi estabelecida de forma a associar o custo com a movimentação entre duas localidades e também o gasto com reabastecimento em uma determinada cidade, conforme (12):

$$R = -(d_{ij} + c_j), \tag{12}$$

em que, d_{ij} é a distância entre as localidades de partida (i) e de chegada (j), c_j é o custo com reabastecimento na cidade de chegada (j). Dessa forma, quanto maior o custo total de movimentação e reabastecimento, mais negativo será a penalidade pela formação da aresta (i, j).

Assim, o modelo tem como objetivo permitir ao agente aprender a planejar rotas que minimizem o deslocamento e o gasto com reabastecimento.

4.2 Experimentos Realizados

A metodologia experimental foi baseada em trabalhos recentes: (Ottoni et al., 2018, 2019). As simulações foram realizadas no *software* $MATLAB^{®}$ e compreenderam 16 grupos de experimentos (2 algoritmos × 4 instâncias × 2 tipos de problemas):

- Instâncias: Bahia30D, Minas24D, Minas30D e Minas57D.
- Algoritmos: *Q-learning* e SARSA.
- Tipos de problemas: não-uniforme e uniforme.

Além disso, para cada grupo de experimentos foram realizadas simulações envolvendo 64 combinações dos parâmetros taxa de aprendizado (α) e fator de desconto (γ), sendo os valores definidos conforme Ottoni et al. (2018):

- α: [0,01; 0,15; 0,30; 0,45; 0,60; 0,75; 0,90; 0,99].
- γ: [0,01; 0,15; 0,30; 0,45; 0,60; 0,75; 0,90; 0,99].

Cada combinação de parâmetros foi simulada em 3 épocas (repetições) com 1000 episódios. Uma época é uma repetição independente, ou seja, o aprendizado é acumulado ao longo dos mil episódios e zerado sempre ao inicializar uma época. Vale ressaltar também que, um episódio é composto por iterações, responsáveis por iniciar, desenvolver e finalizar um rota. As medidas de desempenho de um episódio são a distância total, gasto com reabastecimento e com guincho na rota.

4.3 Modelagem Matemática via RSM

A Metodologia de Superfície de Resposta (RSM) envolve um conjunto de técnicas para análise de problemas de otimização (Myers et al., 2009). A Equação (13) apresenta a estrutura e um modelo RSM de 2ª ordem:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2 + e, \tag{13}$$

em que, y é a variável resposta, x_1 e x_2 são as variáveis independentes, β_n são os coeficientes e o erro aleatório (resíduo) é representado por e.

Ottoni et al. (2018) apresentam a modelagem matemática via RSM para a estimação dos parâmetros α e γ do AR. A estrutura proposta por Ottoni et al. (2018) é dada por (14):

$$\hat{y} = \beta_0 + \beta_1 \alpha + \beta_2 \gamma + \beta_3 \alpha^2 + \beta_4 \gamma^2 + \beta_5 \alpha\gamma, \tag{14}$$

em que, os parâmetros α e γ são as variáveis independentes do modelo e $\hat{y}$ é a resposta predita.

Neste trabalho, foram ajustados 16 modelos RSM de 2ª ordem utilizando o *software* R (R Core Team, 2018; Lenth, 2009), conforme Tabela 1. Esses modelos têm como objetivo estimar os parâmetros α e γ na tentativa de minimizar o gasto em uma rota. Para isso, foram utilizados os dados referentes ao menor gasto na rota (reabastecimento + guincho) em uma época por combinação de α e γ.

Tabela 1. Modelos RSM ajustados.

Modelo	Instância	Algoritmo	Problema
1	Bahia30D	*Q-learning*	Não-uniforme
2		*Q-learning*	Uniforme
3		SARSA	Não-uniforme
4		SARSA	Uniforme
5	Minas24D	*Q-learning*	Não-uniforme
6		*Q-learning*	Uniforme
7		SARSA	Não-uniforme
8		SARSA	Uniforme
9	Minas30D	*Q-learning*	Não-uniforme
10		*Q-learning*	Uniforme
11		SARSA	Não-uniforme
12		SARSA	Uniforme
13	Minas57D	*Q-learning*	Não-uniforme
14		*Q-learning*	Uniforme
15		SARSA	Não-uniforme
16		SARSA	Uniforme

DOI: 10.17648/sbai-2019-111113

Tabela 2. Coeficientes ajustados.

Modelo	β_0	β_1	β_2	β_3	β_4	β_5
1	2710,10	-2319,30	-438,10	1646,70	1195,20	480,70
2	2798,20	-2726,90	-516,90	1901,00	1173,70	644,40
3	2821,98	-2839,97	-483,02	1997,31	1058,81	924,06
4	2709,63	-2701,70	-264,93	1951,85	1024,47	637,31
5	2189,96	-1785,36	-978,54	1321,80	1466,82	433,15
6	2143,82	-1596,51	-937,44	1193,88	1512,57	320,33
7	2229,19	-1918,10	-1123,14	1430,11	1650,32	523,74
8	2175,42	-1796,58	-896,94	1377,56	1428,38	389,12
9	2545,87	-2795,74	-1011,82	2029,03	1812,24	752,80
10	2559,03	-2834,34	-1100,92	2068,00	1955,39	715,21
11	2555,97	-2912,14	-1093,35	2157,97	1939,76	762,75
12	2516,03	-2674,95	-963,81	1950,91	1790,69	785,88
13	4869,70	-6280,20	-2276,40	4355,80	3995,60	1941,60
14	4903,00	-6148,30	-2756,00	4251,90	4501,30	1978,10
15	4900,40	-6262,40	-2208,90	4336,20	3916,80	1886,30
16	4888,40	-6231,90	-2455,10	4293,60	4219,60	2058,50

5. RESULTADOS

Os resultados para o ajuste dos modelos RSM ajustados são descritos em seguida. A metodologia de análise é baseada no trabalho de Ottoni et al. (2018).

5.1 Modelos ajustados

A análise dos modelos ajustados deve observar algumas medidas de adequação, como: normalidade dos resíduos, coeficiente de determinação múltipla (R^2), coeficiente de determinação múltipla ajustado (R^2_a) e significância dos coeficientes individuais (Myers et al., 2009; Ottoni et al., 2018).

O primeiro teste determina se os resíduos dos modelos seguem uma distribuição normal. Adotando o teste de Kolmogorov-Smirnov (Lopes, 2011), observou-se que para os 16 modelos, foi aceita a hipótese de normalidade dos resíduos ($p_{KS} > 0,05$), conforme Tabela 3. Em seguida, foram analisados os valores de R^2 e R^2_a. Quanto mais esses coeficientes se aproximam de 1, é evidenciado um bom ajuste do modelo à amostra. A Tabela 3 também apresenta os valores calculados para R^2 e R^2_a.

Tabela 3. Medidas de adequação dos modelos.

Modelo	p_{KS}	R^2	R^2_a
1	0,7992	0,7636	0,7573
2	0,7966	0,7806	0,7747
3	0,6472	0,7929	0,7873
4	0,8622	0,7867	0,7809
5	0,3864	0,7817	0,7817
6	0,7555	0,8145	0,8095
7	0,8889	0,8373	0,8329
8	0,7294	0,8062	0,8010
9	0,5763	0,8321	0,8276
10	0,3175	0,8506	0,8466
11	0,5569	0,8584	0,8546
12	0,2780	0,8352	0,8308
13	0,9460	0,8553	0,8515
14	0,3722	0,8618	0,8581
15	0,8394	0,8557	0,8518
16	0,7492	0,8738	0,8704

A Tabela 2 apresenta os coeficientes ajustados para cada um dos modelos. Nesse sentido, o teste de significância dos coeficientes individuais, aponta que para todos os modelos, os coeficientes são altamente significantes ($p < 0,001$).

5.2 Pontos Estacionários

A análise de pontos estacionários permite verificar os valores que otimizam a resposta predita nos modelos RSM ajustados. Nesse aspecto, a estimação dos parâmetros α e γ refere-se a um segundo problema de otimização com o objetivo de minimizar a resposta predita $\hat{y}$ (gasto na rota) em cada um dos modelos ajustados. A formulação deste problema é dada por (15) (Ottoni et al., 2018):

$$\begin{aligned} &\underset{\alpha,\gamma}{\text{minimizar}} \quad \hat{y} \\ &\text{sujeito à} \quad 0 \leq \alpha \leq 1, \quad 0 \leq \gamma \leq 1. \end{aligned} \tag{15}$$

A Tabela 4 apresenta os pontos estacionários obtidos utilizando o *software* R (R Core Team, 2018; Lenth, 2009):

Tabela 4. Pontos estacionários.

Modelo	α	γ
1	0,6980	0,0429
2	0,7131	0,0244
3	0,7321	0,0000
4	0,7069	0,0000
5	0,6361	0,2396
6	0,6361	0,2425
7	0,6265	0,2409
8	0,6197	0,2296
9	0,6627	0,1415
10	0,6574	0,1613
11	0,6474	0,1545
12	0,6605	0,1242
13	0,6951	0,1160
14	0,6870	0,1552
15	0,6973	0,1141
16	0,6967	0,1210

Em seguida, foram realizados novos experimentos adotando os valores estimados (pontos estacionários) para os parâmetros α e γ. Além disso, também foram realizadas simulações com parâmetros definidos em outros trabalhos que abordaram a resolução de problemas de otimização combinatória com o AR: $\alpha = 0,1$ e $\gamma = 0,3$ (Gambardella and Dorigo, 1995; Bianchi et al., 2009), $\alpha = 0,8$ e $\gamma = 0,9$ (Sun et al., 2001), $\alpha = 0,1$ e $\gamma = 0,9$ (Liu and Zeng, 2009) e $\alpha = 0,9$ and $\gamma = 1$ (Lima Júnior et al., 2010). Essas combinações de parâmetros foram simuladas em três épocas (repetições) com 10 mil episódios para cada um dos

DOI: 10.17648/sbai-2019-111113

Tabela 5. Melhores soluções encontradas (gasto em reais) adotando os valores dos pontos estacionários e parâmetros definidos em outros trabalhos por grupos de experimentos.

Instância	Algoritmo	Problema	D95	S01	Z09	L10	PE
Bahia30D	*Q-learning*	Não-uniforme	1.771,87	2.187,80	1.762,87	3.364,72	**1.718,25**
	Q-learning	Uniforme	1.730,21	2.241,33	2.006,29	3.109,58	**1.682,05**
	SARSA	Não-uniforme	1.741,65	2.448,15	2.046,19	3.477,01	**1.631,92**
	SARSA	Uniforme	1.743,67	2.247,96	1.790,75	3.114,29	**1.644,14**
Minas24D	*Q-learning*	Não-uniforme	1.610,36	1.957,65	1.630,88	2.650,50	**1.595,50**
	Q-learning	Uniforme	1.615,75	1.665,03	1.678,26	2.468,04	**1.607,46**
	SARSA	Não-uniforme	1.653,52	1.871,15	1.631,37	2.577,96	**1.574,90**
	SARSA	Uniforme	1.628,47	2.082,31	1.724,16	2.540,71	**1.622,49**
Minas30D	*Q-learning*	Não-uniforme	1.643,83	2.110,45	1.831,30	3.259,00	**1.582,92**
	Q-learning	Uniforme	1.673,59	2.105,02	1.845,03	3.083,11	**1.604,29**
	SARSA	Não-uniforme	1.639,86	2.363,15	1.819,63	2.974,17	**1.604,21**
	SARSA	Uniforme	1.642,56	2.513,51	1.775,26	2.954,95	**1.608,64**
Minas57D	*Q-learning*	Não-uniforme	**2.412,82**	4.176,65	3.015,38	5.670,47	2.530,59
	Q-learning	Uniforme	**2.499,96**	4.018,88	2.904,66	5.960,89	2.567,18
	SARSA	Não-uniforme	**2.435,77**	4.722,92	3.077,39	6.208,59	2.453,25
	SARSA	Uniforme	**2.479,90**	4.583,15	2.951,04	5.987,73	2.556,11

Soluções com parâmetros descritos em **D95**: (Gambardella and Dorigo, 1995), **S01**: (Sun et al., 2001), **Z09**: (Liu and Zeng, 2009), **L10**: (Lima Júnior et al., 2010). **PE**: Soluções com pontos estacionários definidos na Tabela 4.

grupos de experimentos. Os resultados (menor gasto na rota) são apresentados na Tabela 5.

Os parâmetros estimados pela RSM alcançaram as melhores soluções nas simulações de três instâncias (Bahia30D, Minas24D e Minas30D) e o segundo desempenho nos experimentos com a instância Minas57D. As Figuras 1 à 3 exemplificam graficamente a capacidade dos parâmetros estimados pela RSM gerar bons resultados. Nesses gráficos é possível visualizar rotas geradas para a instância Minas30D/não-uniforme pelo *Q-learning* com pontos estacionários (Figura 1) e parâmetros adotados em outros trabalhos (Figuras 2 e 3).

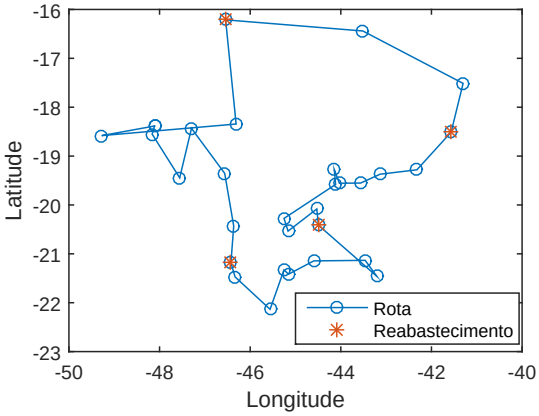

Figura 1. Rota gerada para a instância Minas30D (não-uniforme) pelo algoritmo *Q-learning* com os parâmetros $\alpha = 0,6627$ e $\gamma = 0,1415$ (pontos estacionários). Gasto de R$ 1606,64 e distância de 3957,64 km.

6. CONCLUSÃO

O objetivo deste trabalho foi estimar os parâmetros do AR (α e γ) para o problema de planejamento de rotas com reabastecimento. Para isso, foi proposto um ambiente de simulação baseado em dados reais de preços de combustível da ANP e localização de cidades brasileiras. Além disso, foi adotada a modelagem matemática para a estimação de parâmetros proposta por Ottoni et al. (2018).

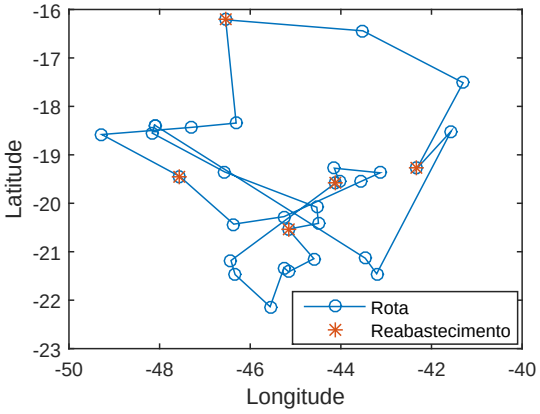

Figura 2. Rota gerada para a instância Minas30D (não-uniforme) pelo algoritmo *Q-learning* com $\alpha = 0,8$ e $\gamma = 0,9$ (parâmetros descritos em (Sun et al., 2001)). Gasto de R$ 2178,35 e distância de 5581,43 km.

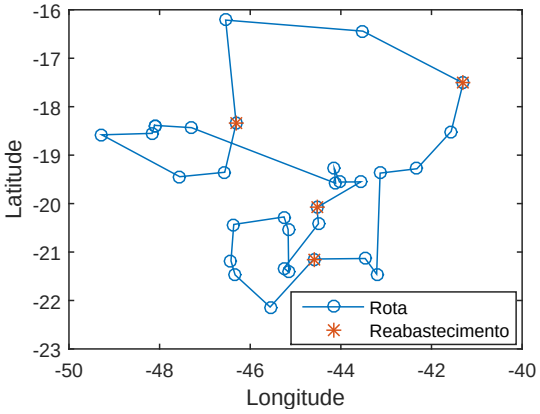

Figura 3. Rota gerada para a instância Minas30D (não-uniforme) pelo algoritmo *Q-learning* com $\alpha = 0,1$ e $\gamma = 0,9$ (parâmetros descritos em (Liu and Zeng, 2009)). Gasto de R$ 1864,52 e distância de 4318,25 km.

DOI: 10.17648/sbai-2019-111113

Os parâmetros estimados alcançaram as melhores soluções nos experimentos envolvendo as instâncias Bahia30D, Minas24D e Minas30D. Esses resultados são válidos para ambos os algortimos (*Q-learning* e SARSA) analisados e também para as simulações com preços de combustível uniforme e não-uniforme em cada localidade. Já para o problema Minas57D, os pontos estacionários conseguiram o segundo desempenho, com resultados não distantes dos valores obtidos pelos parâmetros adotados no trabalho de Gambardella and Dorigo (1995).

Em trabalhos futuros, espera-se analisar outros fatores do problema de planejamento de rotas com reabastescimento, como a influência do porte do veículo (pequeno, médio e grande). Além disso, avaliar outras funções de recompensa e também os efeitos do parâmetro ϵ (política de seleção de ações) no aprendizado.

AGRADECIMENTOS

Agradecemos à CAPES, CNPq/INERGE, FAPEMIG, UFRB e UFSJ (Edital nº 001/2019/Reitoria).

REFERÊNCIAS

Adorno, B.V. and Borges, G.A. (2014). *Robótica Móvel*, chapter Planejamento de Rotas, 84–110. LTC.

Almeida, J.P.S., Arruda, L.V.R., and Neves-Jr, F.A. (2017). Planejamento de rota por meio de algoritmo genético para um enxame de robôs. In *Anais do XIII Simpósio Brasileiro de Automação Inteligente*.

Bianchi, R.A.C., Ribeiro, C.H.C., and Costa, A.H.R. (2009). On the relation between Ant Colony Optimization and Heuristically Accelerated Reinforcement Learning. *1st International Workshop on Hybrid Control of Autonomous System*, 49–55.

Bodin, L., Golden, B., Assad, A., and Ball, M. (1983). Routing and Scheduling of Vehicles and Crews – The State of the Art. *Computers and Operations Research*, 10(2), 63–211.

Even-Dar, E. and Mansour, Y. (2003). Learning Rates for Q-learning. *Journal of Machine Learning Research*, 5, 1–25.

Gambardella, L.M. and Dorigo, M. (1995). Ant-Q: A reinforcement learning approach to the traveling salesman problem. *Proceedings of the 12th International Conference on Machine Learning*, 252–260.

Giardini, G. and Kalmár-Nagy, T. (2011). Genetic algorithm for combinatorial path planning: the subtour problem. *Mathematical Problems in Engineering*, 2011.

Khuller, S., Malekian, A., and Mestre, J. (2007). To fill or not to fill: the gas station problem. In *European Symposium on Algorithms*, 534–545. Springer.

Konar, A., Chakraborty, I.G., Singh, S.J., Jain, L.C., and Nagar, A.K. (2013). A deterministic improved q-learning for path planning of a mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(5), 1141–1153.

Lenth, R.V. (2009). Response-Surface Methods in R, using rsm. *Journal of Statistical Software*, 32(7), 1–17.

Levy, D., Sundar, K., and Rathinam, S. (2014). Heuristics for routing heterogeneous unmanned vehicles with fuel constraints. *Mathematical Problems in Engineering*, 2014.

Li, S., Xu, X., and Zuo, L. (2015). Dynamic path planning of a mobile robot with improved q-learning algorithm. In *Information and Automation, 2015 IEEE International Conference on*, 409–414. IEEE.

Lima Júnior, F.C., Neto, A.D.D., and Melo, J.D. (2010). *Traveling Salesman Problem, Theory and Applications*, chapter Hybrid Metaheuristics Using Reinforcement Learning Applied to Salesman Traveling Problem, 213–236. InTech.

Liu, F. and Zeng, G. (2009). Study of genetic algorithm with reinforcement learning to solve the TSP. *Expert Systems with Applications*, 36(3), 6995 – 7001.

Lopes, R.H.C. (2011). *Kolmogorov-Smirnov Test*, 718–720. Springer Berlin Heidelberg, Berlin, Heidelberg.

Macharet, D.G. and Campos, M.F.M. (2018). A survey on routing problems and robotic systems. *Robotica*, 36(12), 1781–1803.

Myers, R.H., Montgomery, D.C., and Anderson-Cook, C.M. (2009). *Response surface methodology: process and product optimization using designed experiments*. John Wiley & Sons, 3 ed.

Ottoni, A.L.C., Nepomuceno, E.G., Oliveira, M.S., and Felix, B.L. (2016). Modelo híbrido de aprendizado por reforço e lógica fuzzy aplicado ao problema do caixeiro viajante com reabastecimento. In *Anais do XXI Congresso Brasileiro de Automática (CBA 2016)*, 1–6.

Ottoni, A.L.C., Nepomuceno, E.G., and de Oliveira, M.S. (2018). A response surface model approach to parameter estimation of reinforcement learning for the travelling salesman problem. *Journal of Control, Automation and Electrical Systems*, 29(3), 350–359.

Ottoni, A.L.C., Nepomuceno, E.G., de Oliveira, M.S., and de Oliveira, D.C.R. (2019). Tuning of reinforcement learning parameters applied to sop using the scott–knott method. *Soft Computing*, 1–13.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rakshit, P., Konar, A., Bhowmik, P., Goswami, I., Das, S., Jain, L.C., and Nagar, A.K. (2013). Realization of an adaptive memetic algorithm using differential evolution and q-learning: a case study in multirobot path planning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(4), 814–831.

Rodrigues Junior, A.D. (2011). *Um Modelo de Otimização da Política de Reabastecimento para Transportadores Rodoviários de Carga*. Master's thesis, Programa de Pós-Graduação em Engenharia Civil da UFES.

Rodrigues Junior, A.D. and Cruz, M.M.C. (2013). A generic decision model of refueling policies: a case study of a brazilian motor carrier. *Journal of Transport Literature*, 7(4), 8–22.

Russell, S.J. and Norving, P. (2013). *Artificial Intelligence*. Campus, 3st ed.

Schweighofer, N. and Doya, K. (2003). Meta-learning in reinforcement learning. *Neural Networks*, 16(1), 5–9.

Sipahioglu, A., Yazici, A., Parlaktuna, O., and Gurel, U. (2008). Real-time tour construction for a mobile robot in a dynamic environment. *Robotics and Autonomous Systems*, 56(4), 289–295.

Sun, R., Tatsumi, S., and Zhao, G. (2001). Multiagent reinforcement learning method with an improved ant colony system. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 3, 1612–1617.

Sutton, R. and Barto, A. (2018). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 2nd edition.

Suzuki, Y. (2012). A decision support system of vehicle routing and refueling for motor carriers with time-sensitive demands. *Decision Support Systems*, 54(1), 758–767.

Suzuki, Y. (2009). A decision support system of dynamic vehicle refueling. *Decision Support Systems*, 46(2), 522–531.

Watkins, C.J. and Dayan, P. (1992). Technical note Q-learning. *Machine Learning*, 8(3), 279–292.

Yoo, C., Fitch, R., and Sukkarieh, S. (2016). Online task planning and control for fuel-constrained aerial robots in wind fields. *The International Journal of Robotics Research*, 35(5), 438–453.

Yu, Z., Jinhai, L., Guochang, G., Rubo, Z., and Haiyan, Y. (2002). An implementation of evolutionary computation for path planning of cooperative mobile robots. In *Intelligent Control and Automation, 2002. Proceedings of the 4th World Congress on*, volume 3, 1798–1802. IEEE.

DOI: 10.17648/sbai-2019-111113