Journal of Control, Automation and Electrical Systems (2018) 29:350–359
https://doi.org/10.1007/s40313-018-0374-y

A Response Surface Model Approach to Parameter Estimation of Reinforcement Learning for the Travelling Salesman Problem

André L. C. Ottoni[1] · Erivelton G. Nepomuceno[1] · Marcos S. de Oliveira[2]

Received: 13 February 2017 / Revised: 11 July 2017 / Accepted: 22 February 2018 / Published online: 2 March 2018

Abstract
This paper reports the use of response surface model (RSM) and reinforcement learning (RL) to solve the travelling salesman problem (TSP). In contrast to heuristically approaches to estimate the parameters of RL, the method proposed here allows a systematic estimation of the learning rate and the discount factor parameters.The Q-learning and SARSA algorithms were applied to standard problems from the TSPLIB library. Computational results demonstrate that the use of RSM is capable of producing better solutions to both symmetric and asymmetric tests of TSP.

Keywords Reinforcement learning · Travelling salesman problem · Response surface model

1 Introduction

The reinforcement learning (RL) is a technique based on Markov decision processes, in which learning is conducted by success and failure (Sutton and Barto 1998). In a common structure of RL, the agent uses sensors to identify the current state of environment in order to decide the next action. In RL, for every action, an agent receives a reward. This piece of information is stored and used in the choice of following actions. RL has been widely applied in many areas of science and engineering, such as robotics, multi-agent systems, optimal control and optimization (Sutton and Barto 1998).

One of the most important aspects of RL is to estimate the parameters, such as the learning rate and discount factor. Many works, such as Sutton and Barto 1998; Schweighofer and Doya 2003; Even-Dar and Mansour 2003; Gatti 2015; Ottoni et al. 2016, have shown that the RL performance is influenced by the setting of parameters such as the learning rate (α), discount factor (γ) and ϵ-greedy method. For instance, Even-Dar and Mansour (2003) show that the convergence of Q-learning is sensitive to the values of learning rate and discount factor, while Gosavi (2008) presents an empirical study on the effect of learning rate in the convergence of RL algorithms. In order to overcome this problem, Schweighofer and Doya (2003) introduce a concept of meta-parameters for the RL and propose an algorithm to set RL parameters dynamically. Murakoshi and Mizuno (2004), based on the work of Schweighofer and Doya (2003), suggest a parameter control method that responds more quickly to sudden changes in the environment.

In the work of Kobayashi et al. (2009), a parameter adjustment method based on the temporal difference error is proposed. Yoshida et al. (2013) develop a framework to optimize the discount factor, adopting evolutionary algorithms. The method is based on adaptation function of the γ according to the current state of the environment. An additional work can be seen in Tokic et al. (2013), where the authors investigate the determination of the parameters adopting the framework "Reinforce Exploitation Control" and note that the constant defining eligibility traces (λ) is connected to the learning rate value. Other works implement algorithms for adaptive learning rates in dynamic environments (Noda 2010; Dabney 2014; Ryzhov et al. 2015).

One of the most important and challenging application of RL is the travelling salesman problem (TSP), which is combinatorial optimization problem (Gambardella and Dorigo

✉ Erivelton G. Nepomuceno
nepomuceno@ufsj.edu.br

André L. C. Ottoni
andreottoni@ymail.com

Marcos S. de Oliveira
mso@ufsj.edu.br

[1] Control and Modelling Group (GCOM) - Department of Electrical Engineering, Federal University of São João del-Rei, São João del-Rei, Brazil

[2] Department of Mathematics and Statistics, Federal University of São João del-Rei, São João del-Rei, Brazil

1995; Reimann et al. 2001; Sun et al. 2001; Liu and Zeng 2009; Santos et al. 2009; Lima Júnior et al. 2010; Santos et al. 2014; Alipour and Razavi 2015; Ottoni et al. 2015). In this application, the estimation of the parameters is still one of the key aspects to be investigated. In order to overcome this problem, an attempt using a statistical methodology to analyse the effects of RL parameters applied to the TSP has been reported in Ottoni et al. (2015). Although the stochastic nature of RL and the combinatorial feature of TSP seems to at first glance not being possible to identify any relationship between parameters and results, this paper shows that response surface model (RSM) (Myers et al. 2009) is usually able to identify such relationship. The RSM is a statistical technique used in the study process optimization (Myers et al. 2009). Recent studies have addressed the RSM in conjunction with intelligent techniques such as neural networks (Gonçalves Júnior et al. 2014) and genetic algorithms (Mendes et al. 2014). Already in Gatti (2015), the RSM is adopted in the analysis of the influence of RL parameters in the convergence of TD(λ) algorithm into two problems: mountain car problem and truck backer-upper problem. To sum up, RSM can be seen as a box of statistical methods to create typically polynomial functions to represent the answer or the result of an experiment in terms of several independent variables. These functions help to reduce the complexity in finding solution (Gonçalves Júnior et al. 2014).

In this paper, we apply the RSM to calculate optimal values for the learning rate and the discount factor. We show that the learning rate and discount factor can be related to the response of TSP by a RSM. The stationary points of RSM are, then, used as the parameters for RL. Computational results demonstrate that the use of RSM is capable of producing better solutions to both symmetric and asymmetric tests of TSP.

The remainder of this paper is organized as follows. Section 2 presents basic theoretical concepts of the TSP, RL and RSM. Then, Sect. 3 describes a general overview of RL. The experiments carried out and the structure of the proposed mathematical modelling are presented in Sects. 3.1 and 3.2, respectively. The results are given in Sect. 4, and concluding remarks are delivered in Sect. 5.

2 Theoretical Foundation

2.1 Travelling Salesman Problem

The TSP is designed to determine the shortest route among a set of cities, $C = (c_1, c_2, c_3, \ldots, c_n)$ (Applegate et al. 2007; Lima Júnior et al. 2010). A distance (or cost) associated to each pair of city is given by c_{ij}. As a restriction, each location must be visited once and the agent must start and finish the route in the same city. Generally, the TSP is formulated on a graph $G = (N, A)$, where N is the set of nodes (vertices) and A is the set of arcs (i, j) of problem (Goldbarg and Luna 2005).

In this work, the TSP is addressed using two paradigms: symmetric (TSP) and asymmetric (ATSP). In the TSP, the cost associated with the displacement of a city i for j locale is equivalent to the cost of going to j for i. On the other hand, in ATSP, the sense of accomplishment of the route can change the value of the total distance. The experiments were performed adopting problems from travelling salesman problem library (TSPLIB)[1] (Reinelt 1991). The TSPLIB is an open data repository which includes options of case studies of TSP. The TSPLIB repository presents problems (instances) for both symmetric and asymmetric TSP. In addition, the database provides the known optimal value for each instance of the library.

2.2 Reinforcement Learning

Reinforcement learning can be seen as an interaction between agent and environment a sequence of discrete time steps ($t = 0, 1, 2, \ldots$). At each instant t, an agent receives an environmental representation, by means a state, $s_t \in S$, and selects an action $a_t \in A$ (Sutton and Barto 1998). In the next instant $t + 1$, it receives a reinforcement, $r_{t+1} \in R$, and notes the new state s_{t+1}. S is the set of all states, A is the set of actions, and R is the reward function (Sutton and Barto 1998).

The learning rate (α) and discount factor (γ) are used in most of the RL methods. These parameters are generally set in the range between 0 and 1. The learning rate is responsible for controlling the new updates effects on learning matrix. In a different perspective, the discount factor enables the agent to select the actions in an attempt to maximize the sum of rewards in the future. The function R_t in Eq. (1) is the sequence of the time discounted returns, such as

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}, \qquad (1)$$

where $\gamma \in [0, 1]$ (Sutton and Barto 1998).

Next, a brief description of RL algorithms adopted in this work is presented.

2.2.1 Q-learning

The Q-learning (Watkins and Dayan 1992) is based on updating a Q matrix from Eq. (2)

$$Q_{t+1} = Q_t(s, a) + \alpha \left[r(s, a) + \gamma \max_{a'} Q(s', a') - Q_t(s, a) \right], \qquad (2)$$

[1] http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/.

where $Q_t(s, a)$ is value at time t in the Q matrix for the pair state (s) × action (a); $Q_{t+1}(s, a)$ is updating the Q matrix at instant $t + 1$ for the implementation action a in the state s; $r(s, a)$ is the immediate reward for the take of action a in the state s; $\max_{a'} Q(s', a')$ is the use of s', the maximum value in the Q matrix in line with the new state s'. Algorithm 1 shows the Q-learning.

```
Set the parameters: $\alpha$, $\gamma$ and $\epsilon$
For each pair s,a to initialize the matrix Q(s,a)=0
Observe the state s
repeat
    Select the action a using $\epsilon$-greedy method
    Take the action a
    Receive immediate reward r(s, a)
    Observe the new state s'
    Update Q (s, a) with Eq. (2)
    s = s'
until the stopping criterion is satisfied;
```

Algorithm 1: Q-learning

2.2.2 SARSA

The SARSA (Sutton and Barto 1998) is a traditional RL algorithm adaptation from Q-learning. The SARSA (see Algorithm 2) received this name because it involves in its updated terms: s_t (state at time t), a_t (action at time t), $r(s_t, a_t)$ (return to the pair $s_t \times a_t$) s_{t+1} (state at time $t + 1$) and a_{t+1} (action at time $t + 1$). Equation (3) describes the update of the Q matrix by SARSA with the execution of action a in the state s:

$$Q_{t+1} = Q_t(s, a) + \alpha[r(s, a) + \gamma Q_t(s', a') - Q_t(s, a)]. \tag{3}$$

```
Set the parameters: $\alpha$, $\gamma$ and $\epsilon$
For each pair s,a to initialize the matrix Q(s,a)=0
Observe the state s
Select the action a using $\epsilon$-greedy method
repeat
    Take the action a
    Receive immediate reward r(s, a)
    Observe the new state s'
    Select the new action a using $\epsilon$-greedy method
    Update Q (s, a) with Eq. (3)
    s = s'
    a = a'
until the stopping criterion is satisfied;
```

Algorithm 2: SARSA

In Algorithms 1 and 2, the ϵ-greedy method is responsible to control between greedy and randomness in the selection of actions (Sutton and Barto 1998).

2.3 Response Surface Models

The response surface model (RSM) is a set of statistical techniques for the optimization of processes (Myers et al. 2009). The performance measure is called the response, and input variables are called independent variables (IV) (Myers et al. 2009). The response surface model features the same structure of the multiple linear regression models (Myers et al. 2009). Thus, Eqs. (4) and (5) have the structure of RSM models of first and second order, with two IV (x_1 and x_2):

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + e, \tag{4}$$

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2 + e. \tag{5}$$

The effect of the error in the response is represented by e. For the estimation of the model coefficients (β), it is usually adopted the method of least squares, assuming normal distribution with zero mean and constant variance (Myers et al. 2009). According to Myers et al. (2009), the models of second order are more suitable to real surface problems.

3 Methodology

Application of the RL to solve the TSP requires the definition of a model with a set of states (S), actions (A) and rewards (R). Here the method adopted for the development of learning strategy is divided into four steps:

1. Definition of finite set of environmental states: in this case, states are locations where the travelling salesman (agent) must access. This definition ensures the resolution of the TSP as a problem of sequential decision (Lima Júnior et al. 2010). Moreover, the set S always has the same size of the problem instance (Lima Júnior et al. 2010).
2. Definition of finite set of actions that the agent is able to perform: each action intends to go to another location (state) of the problem. It is noteworthy that, to avoid repetition of locations on the route, actions that lead to states already visited should not be available (Lima Júnior et al. 2010).
3. Definition of reward values for each pair state (s) versus action (a): reinforcements were defined as distances between locations multiplied by − 1, according to Eq. (6):

   $$r_{ij} = -d_{ij}, \tag{6}$$

   where i and j are the locations, d_{ij} is the distance between the cities i and j and r_{ij} is the reinforcement received by from i to j. Thus, the greater the distance, the more negative is strengthening. Thus, it is expected that an agent finds the shortest distance between two locations

Table 1 TSPLIB instances

Type	Instance	n	Optimal solution
TSP	berlin52	52	7542
	brazil58	58	25,395
	kroA100	100	21,282
	kroA200	200	29,368
ATSP	br17	17	39
	ftv33	34	1286
	ftv44	45	1613
	ftv64	65	1839

We select 4 symmetric and 4 asymmetric problems. n is the number of cities. The fourth columns indicated the best known solution

to reduce the penalty. This approach is the same adopted in Bianchi et al. (2009).

4. Tests are performed using Q-learning and SARSA algorithms in a MATLAB platform.

3.1 Experiments

The experiments were performed in the *MATLAB* and included tests on instances of TSPLIB as indicated in Table 1. Simulations were performed involving a group of 64 combinations of the learning rate (α) and discount factor (γ) for each TSP problem. The values for α and γ are:

$$\alpha = [0.01\ 0.15\ 0.30\ 0.45\ 0.60\ 0.75\ 0.90\ 0.99].$$

and

$$\gamma = [0.01\ 0.15\ 0.30\ 0.45\ 0.60\ 0.75\ 0.90\ 0.99].$$

Moreover, each combination was simulated five times with 1000 episodes. The response of an episode is the total distance (cost) travelled by the agent on the route (Ottoni et al. 2015).

The value for ϵ was defined based in Ottoni et al. (2015). Ottoni et al. (2015) analyse the effects of ϵ in three TSP instances: berlin52, brazil58 and kroA100. They found good results with $\epsilon = 0.01$.

3.2 Mathematical Modelling

In this work, mathematical models of second order for each instance described in Table 1 using Q-learning and SARSA have been fitted, in a total of 16 models. These models aims at representing sensitivity to parameters (α and γ). The structure of the proposed models is composed of three variables: y, α and γ. The response variable (y) is the average distance travelled by the salesman on the route. For each of the five repetitions of each instance, an average was calculated for each combination of α and γ. In addition, the independent variables are $IV_1 = \alpha$ and $IV_2 = \gamma$. Thus, the models have the form of Eq. (7):

$$y = \beta_0 + \beta_1\alpha + \beta_2\gamma + \beta_3\alpha^2 + \beta_4\gamma^2 + \beta_5\alpha\gamma + e. \tag{7}$$

Equation (8) represents the structure of adjusted models having as output the predicted response $\hat{y}_i$. In this case, the error does not appear in Eq. (8) as e_i is the difference between an observation (y_i) and its predicted response ($\hat{y}_i$), or $e_i = y_i - \hat{y}_i$.

$$\hat{y} = \beta_0 + \beta_1\alpha + \beta_2\gamma + \beta_3\alpha^2 + \beta_4\gamma^2 + \beta_5\alpha\gamma. \tag{8}$$

To facilitate the identification of models, each structure received a code, given by the union of the first letter of the algorithm and the instance name. For example, for the model representing the simulations of the Q-learning algorithm in berlin52 problem, the code is Qberlin52.

For the adjustment of the models was adopted RSM package of statistical software R (Lenth 2009; Core Team 2013).

4 Results

The results for the adjustment of the response surface model are described below. The analysis of results with RSM comprises three stages:

1. Analysis of model adjustment measures: the objective is to verify if the models meet some statistical requirements, such as normality of the residues and significance of the coefficients.
2. Analysis of contours and surfaces graphics: the goal is to visualize graphically as the response variable is influenced by the levels of parameters α and γ.
3. Analysis and simulations with stationary points in order to check the optimality of the response in a model. In addition, the values of the stationary points are adopted in new experiments to analyse the performance of the parameters set by the models.

4.1 Adjustment Measures

4.1.1 Residual Analysis

One of these tests is to determine whether the residues of the models are normally distributed (Hines et al. 2006). Let $e_i = y_i - \widehat{y}_i$, $i = 1, 2, \ldots, n$, where y_i is a note and $\widehat{y}_i$ is the corresponding value estimated from the regression model (Hines et al. 2006). We used the Kolmogorov–Smirnov test (KS test) (Lopes 2011) to check the assumption. The corresponding p values of the KS test for the 16 second-order

Table 2 Adjustment measures: p values of the KS test (p_{KS}), R^2 and adjusted R^2

Model	p_{KS}	R^2	R_a^2
Qberlin52	0.6801	0.8914	0.8896
Sberlin52	0.2491	0.9037	0.9022
Qbrazil58	0.2769	0.9008	0.8992
Sbrazil58	0.3649	0.9080	0.9065
QkroA100	0.1068	0.8959	0.8942
SkroA100	0.0579	0.9025	0.9010
QkroA200	0.0000012	0.7897	0.7864
SkroA200	0.1943	0.8965	0.8948
Qbr17	0.0564	0.7159	0.7114
Sbr17	0.3025	0.8377	0.8352
Qftv33	0.4205	0.8712	0.8691
Sftv33	0.3692	0.8872	0.8854
Qftv44	0.4019	0.8817	0.8798
Sftv44	0.1803	0.8932	0.8915
Qftv64	0.0536	0.9015	0.8999
Sftv64	0.0025	0.9091	0.9076

models are shown in Table 2 denoted by p_{KS}. In the KS test, the initial hypothesis (H_0) is that the residues follow a normal distribution ($p_{\text{KS}} > 0.05$), and the alternative hypothesis (H_1) otherwise ($p_{\text{KS}} < 0.05$) (Lopes 2011).

Applying the RL to estimate the parameters of Q-learning and SARSA allows to find at least one suitable model for all instances. Only the models QkroA200 ($p_{\text{KS}} = 1.19 \times 10^{-6}$) and Sftv64 ($p_{\text{KS}} = 0.0025$) did not confirm the residues normality hypothesis, because $p_{\text{KS}} < 0.05$. Nevertheless, the results of these models QkroA200 and Sftv64 are presented in the next sections. Although these models does not present the normality of residues confirmed, their stationary points present competitive responses when compared to other literature results (see Table 6).

4.1.2 R^2 and Adjusted R^2

Other components analysis of the adequacy of a response surface model is: coefficient of multiple determination (R^2) and adjusted coefficient of multiple determination (R_a^2) (Myers et al. 2009). These coefficients set between 0 and 1 indicating how much variability is explained by the model. When R^2 and R_a^2 approach 1, it is an evidence of a good model (Hines et al. 2006). Table 2 presents the adjusted values for R^2 and R_a^2.

The importance to use more than one index to evaluate the quality of the models can be illustrated with the models QkroA200 and Sftv64. Although the KS test does not confirm the normality of the residuals, the values of R^2 and R_a^2 are significant, which means a good explanation of the variability of data. This evidence is confirmed when the stationary point of these models are used, as shown in Tables 5 and 6.

4.1.3 Tests on Individual Regression Coefficients

The tests on individual regression coefficients check the hypothesis for the significance of each variable in the model. The hypotheses for testing the significance of any individual regression coefficient, say β_j are $H_0\colon \beta_j = 0$ and $H_1\colon \beta_j \neq 0$ (Myers et al. 2009). If $H_1\colon \beta_j \neq 0$ is accept ($p < 0.05$), then this indicates that corresponding variable x_j is significant from the model.

Tables 3 and 4 present the adjusted coefficients for each model under study, adopting the Q-learning and SARSA, respectively. The significance testing of individual coefficients for this work showed that for the 16 models, the coefficients are significant ($p < 0.05$). Only for the Qbr17 model the terms γ ($p = 0.606$) and $\alpha\gamma$ ($p = 0.84998$) are not significant. The term intercept refers to the linear coefficient (β_0) of the proposed model, as shown in Eq. (7).

4.2 Surface and Contours Graphics

The RSM provides graphical tools for analysis contour plot and response surface (Myers et al. 2009). The contour plot gives a two-dimensional view between the IVs (α and γ) and the response variable (y) of the model. Thus, in this type of graph, the IVs are given in the scales x and y and the response values are represented by the contour lines. In this sense, a contour plot is similar to a topographic map, where the latitude values are represented (axis x), longitude (axis y) and elevation (contours). Thus, from contour lines it is possible to identify regions that approach the minimum or maximum of the adjusted response.

In this work, the contours plot shows in two dimensions as the learning rate (α) and the discount factor (γ) influence the distance (contours–response variable) by travelling salesman on the route. Figure 1 shows the contour plot for Sberlin52 model, referring to experiments with berlin52 proceedings and adoption of SARSA algorithm. The red region indicates the set of points, given by the relationship between α and γ, which produce lower values for the response variable.

As for the analysis in three dimensions, the tool adopted is the response surface graph. Figure 2 presents this display surface for Sberlin52 model. Similar to the contour plot, it is possible to identify regions of α and γ approaching the minimum of the response variable (distance). The IVs (α and γ) are displayed on the scales x and y in the 3D graphic. The response variable (distance–axis z) is represented by the surface. In these 3D graphs, the red region also indicates where the response variable (distance) tends to be minimized.

Table 3 Coefficients for models with Q-learning, where p states for the level of significance are given in italics

Model	Coef.	β	p
Qberlin52	Intercept	19,051	$< 2 \times 10^{-16}$
	α	$-23,477$	$< 2 \times 10^{-16}$
	γ	-9839	$< 2 \times 10^{-16}$
	α^2	15,476	$< 2 \times 10^{-16}$
	γ^2	17,169	$< 2 \times 10^{-16}$
	$\alpha\gamma$	6262	9.6×10^{-14}
Qbrazil58	Intercept	70,114	$< 2 \times 10^{-16}$
	α	$-89,081$	$< 2 \times 10^{-16}$
	γ	$-39,545$	$< 2 \times 10^{-16}$
	α^2	56,615	$< 2 \times 10^{-16}$
	γ^2	67,766	$< 2 \times 10^{-16}$
	$\alpha\gamma$	29,422	$< 2 \times 10^{-16}$
QkroA100	Intercept	92,844	$< 2 \times 10^{-16}$
	α	$-134,710$	$< 2 \times 10^{-16}$
	γ	$-50,035$	1.74×10^{-14}
	α^2	85,159	$< 2 \times 10^{-16}$
	γ^2	92,922	$< 2 \times 10^{-16}$
	$\alpha\gamma$	32,659	3.67×10^{-12}
QkroA200	Intercept	195,160	$< 2 \times 10^{-16}$
	α	$-230,422$	$< 2 \times 10^{-16}$
	γ	$-119,200$	3.1×10^{-13}
	α^2	161,552	$< 2 \times 10^{-16}$
	γ^2	197,427	$< 2 \times 10^{-16}$
	$\alpha\gamma$	25,827	0.0238
Qbr17	Intercept	118.8969	$< 2 \times 10^{-16}$
	α	-95.6770	$< 2 \times 10^{-16}$
	γ	3.5651	*0.606*
	α^2	77.0405	$< 2 \times 10^{-16}$
	γ^2	32.8138	$< 2 \times 10^{-16}$
	$\alpha\gamma$	-0.3029	*0.952*
Qftv33	Intercept	3157	$< 2 \times 10^{-16}$
	α	-3927	$< 2 \times 10^{-16}$
	γ	1639	$< 2 \times 10^{-16}$
	α^2	2682	$< 2 \times 10^{-16}$
	γ^2	2648	$< 2 \times 10^{-16}$
	$\alpha\gamma$	728	3.24×10^{-8}
Qftv44	Intercept	4380	$< 2 \times 10^{-16}$
	α	-5600	$< 2 \times 10^{-16}$
	γ	-2087	1.48×10^{-15}
	α^2	3684	$< 2 \times 10^{-16}$
	γ^2	3564	$< 2 \times 10^{-16}$
	$\alpha\gamma$	1243	3.09×10^{-11}
Qftv64	Intercept	6097	$< 2 \times 10^{-16}$
	α	-8162	$< 2 \times 10^{-16}$
	γ	-2986	3.41×10^{-16}
	α^2	5100	$< 2 \times 10^{-16}$
	γ^2	5266	$< 2 \times 10^{-16}$
	$\alpha\gamma$	2072	4.97×10^{-15}

Table 4 Coefficients for models with SARSA

Model	Coef.	β	p
Sberlin52	Intercept	18,656	$< 2 \times 10^{-16}$
	α	$-22,730$	$< 2 \times 10^{-16}$
	γ	-7097	1.9×10^{-11}
	α^2	14,985	$< 2 \times 10^{-16}$
	γ^2	14,487	$< 2 \times 10^{-16}$
	$\alpha\gamma$	6718	$< 2 \times 10^{-16}$
Sbrazil58	Intercept	68,807	$< 2 \times 10^{-16}$
	α	$-85,423$	$< 2 \times 10^{-16}$
	γ	$-30,211$	2.16×10^{-13}
	α^2	54,619	$< 2 \times 10^{-16}$
	γ^2	58,197	$< 2 \times 10^{-16}$
	$\alpha\gamma$	30,530	$< 2 \times 10^{-16}$
SkroA100	Intercept	91,178	$< 2 \times 10^{-16}$
	α	$-129,538$	$< 2 \times 10^{-16}$
	γ	$-37,187$	6.76×10^{-10}
	α^2	81,825	$< 2 \times 10^{-16}$
	γ^2	79,626	$< 2 \times 10^{-16}$
	$\alpha\gamma$	34,971	4.60×10^{-15}
SkroA200	Intercept	191,276	$< 2 \times 10^{-16}$
	α	$-236,975$	$< 2 \times 10^{-16}$
	γ	$-75,721$	1.07×10^{-11}
	α^2	147,295	$< 2 \times 10^{-16}$
	γ^2	14,7568	$< 2 \times 10^{-16}$
	$\alpha\gamma$	62,817	1.59×10^{-14}
Sbr17	Intercept	116,829	$< 2 \times 10^{-16}$
	α	$-86,174$	$< 2 \times 10^{-16}$
	γ	12.367	0.01779
	α^2	67.467	$< 2 \times 10^{-16}$
	γ^2	26.247	2.26×10^{-8}
	$\alpha\gamma$	10.321	0.00656
Sftv33	Intercept	3110	$< 2 \times 10^{-16}$
	α	-3888	$< 2 \times 10^{-16}$
	γ	-1281	4.64×10^{-14}
	α^2	2658	$< 2 \times 10^{-16}$
	γ^2	2278	$< 2 \times 10^{-16}$
	$\alpha\gamma$	877	9.28×10^{-13}
Sftv44	Intercept	4298	$< 2 \times 10^{-16}$
	α	-5442	$< 2 \times 10^{-16}$

Table 4 continued

Model	Coef.	β	p
	γ	− 1593	3.93×10^{-11}
	α^2	3550	$< 2 \times 10^{-16}$
	γ^2	3079	$< 2 \times 10^{-16}$
	$\alpha\gamma$	1401	3.29×10^{-15}
Sftv64	Intercept	6006.3	$< 2 \times 10^{-16}$
	α	− 7976.9	$< 2 \times 10^{-16}$
	γ	− 2320.6	7.81×10^{-12}
	α^2	4975.0	$< 2 \times 10^{-16}$
	γ^2	4590.5	$< 2 \times 10^{-16}$
	$\alpha\gamma$	2228.6	$< 2 \times 10^{-16}$

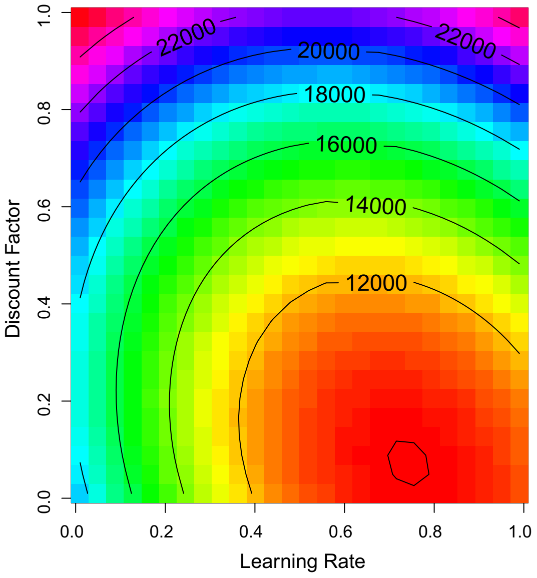

Fig. 1 Contours graphics for Sberlin52 model

4.3 Stationary Points

The identification of stationary points (minimum or maximum) is interesting to check the values that optimize the predicted response in RSM models (Myers et al. 2009). In the TSP, the goal is to minimize the distance travelled on the route. Thus, the desired stationary points on the modelled surfaces are the minimum RSM functions.

In the TSP, the goal is to minimize the distance travelled on the route. Thus, the desired stationary points are the minimum modelled surfaces. It is noteworthy that the definition of stationary points refers to a second optimization problem in this work, that is, find the values of the parameters α and γ that minimize the predicted response $\hat{y}$ in each the adjusted models. The optimization problem is formulated as

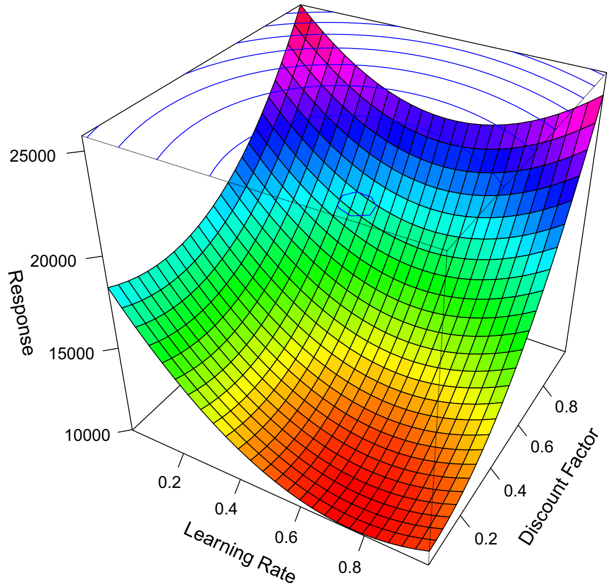

Fig. 2 Response Surface for Sberlin52 model

Table 5 Stationary points for Q-learning and SARSA

	Q-learning		SARSA	
Problem	α	γ	α	γ
berlin52	0.7273	0.1539	0.7421	0.0729
brazil58	0.7534	0.1282	0.7656	0.0587
kroA100	0.7651	0.1348	0.7782	0.0626
kroA200	0.6927	0.2566	0.7854	0.0894
br17	0.6208	0	0.6667	0
ftv33	0.7032	0.2128	0.7074	0.1450
ftv44	0.7321	0.1652	0.7491	0.0882
ftv64	0.7735	0.1313	0.7879	0.0615

$$\begin{aligned} &\underset{\alpha,\gamma}{\text{minimize}} \ \hat{y} \\ &\text{subject to } 0 \leq \alpha \leq 1 \\ &\qquad\qquad\quad 0 \leq \gamma \leq 1 \end{aligned}$$

Table 5 shows the stationary points obtained from the canonical analysis in software R (Myers et al. 2009; Lenth 2009).

The next step is to verify the RL performance using the stationary points for the parameters α and γ. Thus, the combinations have been simulated with five replicates with 10,000 (ten thousand) episodes. Table 6 shows the best results found for each instance with the adjusted values of learning rate and discount factor (stationary points) for Q-learning and SARSA algorithms, respectively.

In addition, experiments were conducted by adopting the parameters used in other works: $\alpha = 0.1$ and $\gamma = 0.3$ (Gambardella and Dorigo 1995; Bianchi et al. 2009), $\alpha = 0.8$ and $\gamma = 0.9$ (Sun et al. 2001), $\alpha = 0.1$ and $\gamma = 0.9$ (Liu and

Table 6 Best solution found with the Q-learning and SARSA for each problem by adopting the values of the stationary points (SP) and parameters as other works

TSPLIB		Q-learning					SARSA				
Problem	Optimal	D95	S01	Z09	L10	SP	D95	S01	Z09	L10	SP
berlin52	7542	8871	10,920	8710	15,126	8619	9169	12,206	9115	16,441	8048
brazil58	25,395	27,895	41,816	34,881	54,371	27,487	28,284	44,791	34,798	49,156	26,685
kroA100	21,282	25,363	47,580	37,749	64,022	24,925	26,764	55,873	42,301	67246	24263
kroA200	29,368	37,704	83,341	76,833	121,336	38,254	38,468	98,795	82,088	121879	35311
br17	39	39	39	40	40	39	39	39	39	42	39
ftv33	1286	1525	1650	1517	2245	1464	1533	1810	1501	2245	1381
ftv44	1613	1980	2372	2057	2631	1873	2033	2692	2091	2692	1812
ftv64	1839	2411	3281	2631	4682	2279	2432	3605	2570	4638	2139

Optimal: optimal solution of TSPLIB. Solutions with parameters described in D95: Gambardella and Dorigo (1995), S01: Sun et al. (2001), Z09: Liu and Zeng (2009), L10: Lima Júnior et al. (2010). In the column SP, the solutions are obtained with parameters of stationary points in Table 5

Zeng 2009) and $\alpha = 0.9$ and $\gamma = 1$ (Lima Júnior et al. 2010; Santos et al. 2014). These combinations were also simulated in five replicates with 10,000 episodes, and the best results are presented in Table 6.

For SARSA algorithm, stationary points achieved the best results in all instances. The results show that for the Q-learning, the parameters set by RSM reached the best performance in seven instances and a second place in the kroA100 problem, in which parameters used by Gambardella and Dorigo (1995) achieved the best result. Here it is important to stress the ability of RSM to indicate good parameters. Taking as example the parameters used by Gambardella and Dorigo (1995) for the ftv44 problem, it is clear to see that the pair of parameters $\alpha = 0.1$ and $\gamma = 0.3$ is located in a green contour in the Fig. 3, quite far away from the red area, in which it is possible to find the stationary point ($\alpha = 0.7320604$ e $\gamma = 0.1652470$). In this same aspect, still analysing Fig. 3, the point defined by the parameters of Lima Júnior et al. (2010) ($\alpha = 0.9$ and $\gamma = 1$) is in the blue region of the graph. In this respect, the contour plot offers a visual aspect that helps to define the are with the best values of α and γ.

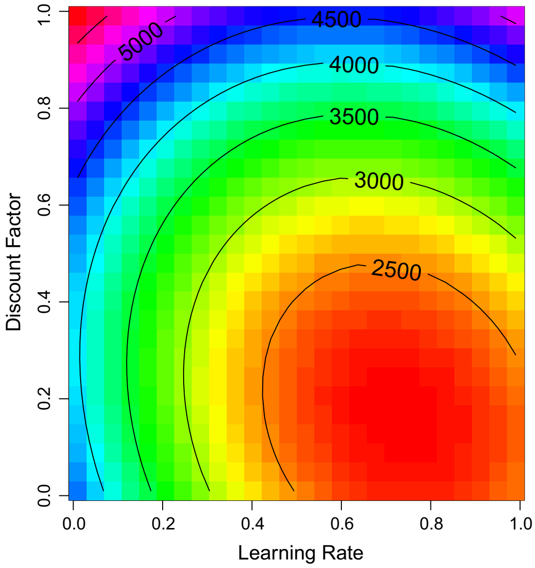

Fig. 3 Contours graphics for Qftv44 model. The stationary point $\alpha = 0.7320604$ and $\gamma = 0.1652470$ estimated in this paper is in the area (red) that approaches the minimum of the adjusted response. On the other hand, the set of parameters (0.1; 0.3) and (0.9; 1) heuristically adopted by Gambardella and Dorigo (1995) and Lima Júnior et al. (2010), respectively, is clearly distant from the red area (Color figure online)

5 Conclusion

This paper has addressed the problem of estimating the parameters of RL for the TSP. We applied the RSM to relate the response variable (average distance travelled) to the learning rate α and discount factor γ. Our main contribution is to present a systematic approach to overcome the problem of ad hoc estimations of these parameters. As a consequence, our results present better performance to those present in the literature without a systematic approach.

The TSP is frequently discussed in the literature, and many methods have been investigated, such as ant colony algorithm (Dorigo and Gambardella 1997), memetic algorithms (Buriol et al. 2004), genetic algorithms (Deng et al. 2015), tabu search (Fiechter 1994), particle swarm optimization (Marinakis and Marinaki 2010), neural networks (Siqueira et al. 2007), simulated annealing (Chen and Chien 2011) and others (Applegate et al. 2007; Cook 2011; Marinakis et al. 2011; Ouaarab et al. 2014). Important contributions were also conducted to works that applied the RL in solving the TSP. Gambardella and Dorigo (1995) perform a connection between the ant system (AS) and RL.

Another recurrent approach in solving combinatorial optimization problems is the development of hybrid solutions between genetic algorithms (GAs) and reinforcement learning (Liu and Zeng 2009; Santos et al. 2009; Lima Júnior et al. 2010). Lima Júnior et al. (2010) propose the adoption of Q-learning algorithm as a strategy of exploration/ exploitation for GRASP metaheuristics and GAs. Similarly, Santos et al. (2009) also discuss resolutions with RL, GRASP and GAs, but adopting a parallel implementation. Liu and Zeng (2009), in turn, proposes the RMGA algorithm, which also includes GAs and RL. Following the same line of combining techniques for the resolution of the TSP, the work Bianchi et al. (2009) applies acceleration heuristics in RL.

In this paper, we show that RSM allows to identify how the performance of the RL is influenced by the levels of learning rate and discount factor. This is made possible to the set of tools available for RSM. The contours and response surface graphics provide an important visual aspect as the sensitivity of the RL values of α and γ. The analysis of stationary points, in turn, allows inferring for each model which parameter values tend to optimize the response.

The parameters set by RSM achieved the best overall performance in SARSA simulations, among the combinations of α and γ analysed. As for the Q-learning, stationary points obtained the best performance in seven instances and a second position in one instance The analysis of surface and contours graphics identifies regions close to the stationary point that optimize the response of TSP. It is a valuable help to estimate the parameters without a trial and error approach. Additionally, as it has been noticed to the parameters suggested by Gambardella and Dorigo (1995), the RSM the contour plots is very practical tool to avoid low performance parameters.

In future works, we intend to investigate the sensitivity of the RL parameters in other combinatorial optimization problems and also other traditional areas of the RL application, such as mobile robotics and multi-agent systems. Also, we plan to improve the mathematical modelling by RSM for a three-parameter function: α, γ and also ϵ, thus adding the effects of the policy ϵ-greedy in settings of response surface model. Additionally, it will be worth to compare the results with other techniques, such as artificial neural network, as it has been done by Erzurumlu and Oktem (2007); Desai et al. (2008) or polynomial NARMAX (Billings 2013). Other possible future works are to evaluate dynamic methods of parameter definition for the TSP. As done by Schweighofer and Doya (2003), for adaptive algorithms it is necessary to define the initial values for the parameters. In this respect, an important point would be to adopt RSM to model mathematically the influence of the initial conditions for α, γ and ϵ.

Acknowledgements The authors acknowledge the financial support of CAPES, CNPq, FAPEMIG and UFSJ.

References

Alipour, M. M., & Razavi, S. N. (2015). A new multiagent reinforcement learning algorithm to solve the symmetric traveling salesman problem. *Multiagent and Grid Systems*, *11*(2), 107–119.

Applegate, D., Bixby, R . E., Chvtal, V., & Cook, W. (2007). *The traveling salesman problem: A computational study*. Princeton: Princeton University Press.

Bianchi, R. A. C., Ribeiro, C. H. C., & Costa, A. H. R. (2009). On the relation between Ant Colony Optimization and Heuristically Accelerated Reinforcement Learning. In *1st international workshop on hybrid control of autonomous system* (pp. 49–55).

Billings, S. A. (2013). *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. West Sussex: Wiley.

Buriol, L., Frana, P., & Moscato, P. (2004). A new memetic algorithm for the asymmetric traveling salesman problem. *Journal of Heuristics*, *10*(5), 483–506.

Chen, S.-M., & Chien, C.-Y. (2011). Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques. *Expert Systems with Applications*, *38*(12), 14439–14450.

Cook, W . J. (2011). *In pursuit of the traveling salesman: Mathematics at the limits of computation*. Princeton: Princeton University Press.

Dabney, W. (2014). *Adaptive step-sizes for reinforcement learning*. Ph.D. Thesis, Amherst, MA: University of Massachusetts Amherst.

Deng, Y., Liu, Y., & Zhou, D. (2015). An improved genetic algorithm with initial population strategy for symmetric TSP. *Mathematical Problems in Engineering*, *2015*, 212794. https://doi.org/10.1155/2015/212794.

Desai, K. M., Survase, S. A., Saudagar, P. S., Lele, S., & Singhal, R. S. (2008). Comparison of artificial neural network (ann) and response surface methodology (RSM) in fermentation media optimization: Case study of fermentative production of scleroglucan. *Biochemical Engineering Journal*, *41*(3), 266–273.

Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, *1*(1), 53–66.

Erzurumlu, T., & Oktem, H. (2007). Comparison of response surface model with neural network in determining the surface quality of moulded parts. *Materials & Design*, *28*(2), 459–465.

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

Fiechter, C.-N. (1994). A parallel tabu search algorithm for large traveling salesman problems. *Discrete Applied Mathematics*, *51*(3), 243–267.

Gambardella, L. M. & Dorigo, M. (1995). Ant-Q: A reinforcement learning approach to the traveling salesman problem. In *Proceedings of the 12th international conference on machine learning* (pp. 252–260).

Gatti, C. (2015). *Design of experiments for reinforcement learning*. Berlin: Springer.

Goldbarg, M . C., & Luna, H. P . L. (2005). *Otimização Combinatória e Programação Linear*. Amsterdam: Elsevier/Campus.

Gonçalves Júnior, A. M., Rocha, V. V. R., Baccarini, L. M. R., & Reis, M. L. F. (2014). Three-phase induction motors faults recognition and classification using neural networks and response surface models. *Journal of Control, Automation and Electrical Systems*, *25*(3), 330–338.

Gosavi, A. (2008). On step sizes, stochastic shortest paths, and survival probabilities in reinforcement learning. In *Proceedings of the 40th conference on winter simulation* (pp. 525–531).

Hines, W. W., Montgomery, D. C., Goldsman, D. M., & Borror, C. M. (2006). *Probabilidade e Estatística na Engenharia*. Groupo Gen-LTC. (In Portuguese).

Kobayashi, K., Mizoue, H., Kuremoto, T., & Obayashi, M. (2009). *A meta-learning method based on temporal difference error* (pp. 530–537). Berlin: Springer.

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

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

Liu, F., & 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* (pp. 718–720). Berlin: Springer.

Marinakis, Y., & Marinaki, M. (2010). A hybrid multi-swarm particle swarm optimization algorithm for the probabilistic traveling salesman problem. *Computers and Operations Research*, *37*(3), 432–442.

Marinakis, Y., Marinaki, M., & Dounias, G. (2011). Honey bees mating optimization algorithm for the euclidean traveling salesman problem. *Information Sciences*, *181*(20), 4684–4698.

Mendes, L. F. S., Baccarini, L. M. R., & Abreu Júnior, L. (2014). Diagnóstico de Falhas em Motores de Indução Utilizando Superfície de Resposta e Algoritmos Genéticos. *XX CBA - Congresso Brasileiro de Automática* (pp. 2946–2953).

Murakoshi, K., & Mizuno, J. (2004). A parameter control method in reinforcement learning to rapidly follow unexpected environmental changes. *Biosystems*, *77*(1–3), 109–117.

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

Noda, I. (2010). Recursive adaptation of stepsize parameter for non-stationary environments. *Lecture Notes in Computer Science*, *5924*, 74–90.

Ottoni, A. L. C., Nepomuceno, E. G., Cordeiro, L. T., Lamperti, R. D., & Oliveira, M. S. (2015). Análise do Desempenho do Aprendizado por Reforço na Solução do Problema do Caixeiro Viajante. *XII SBAI - Simpósio Brasileiro de Automação Inteligente* (pp. 43–48). (In Portuguese).

Ottoni, A. L. C., Nepomuceno, E. G., Oliveira, M. S., Cordeiro, L. T., & Lamperti, R. D. (2016). Análise da influência da taxa de aprendizado e do fator de desconto sobre o desempenho dos algoritmos Q-learning e SARSA: aplicação do aprendizado por reforço na navegação autônoma. *Revista Brasileira de Computação Aplicada*, *8*(2), 44–59. (In Portuguese).

Ouaarab, A., Ahiod, B., & Yang, X.-S. (2014). Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Computing and Applications*, *24*(7–8), 1659–1669.

R Core Team. (2013). *R: A language and environment for statistical computing*. Vienna: R Foundation for Statistical Computing.

Reimann, M., Shtovba, S., & Nepomuceno, E. (2001). A hybrid ant colony optimization and genetic algorithm approach for vehicle routing problems solving. *Student papers of the complex systems summer school* (pp. 134–141). Santa Fe Institute: Budapest.

Reinelt, G. (1991). TSPLIB–A traveling salesman problem library. *ORSA Journal on Computing*, *3*(4), 376–384.

Ryzhov, I. O., Frazier, P. I., & Powel, W. B. (2015). A new optimal stepsize for approximate dynamic programming. *IEEE Transactions on Automatic Control*, *60*, 743–757.

Santos, J. P. Q., Melo, J. D., Duarte Neto, A. D., & Aloise, D. (2014). Reactive search strategies using reinforcement learning, local search algorithms and variable neighborhood search. *Expert Systems with Applications*, *41*(10), 4939–4949.

Santos, J. Q., Lima Júnior, F., Magalhaes, R., de Melo, J., & Neto, A. (2009). A parallel hybrid implementation using genetic algorithm, GRASP and reinforcement learning. In *International Joint Conference on Neural networks, 2009. IJCNN 2009* (pp. 2798–2803).

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

Siqueira, P., Steiner, M., & Scheer, S. (2007). A new approach to solve the traveling salesman problem. *Neurocomputing*, *70*(4–6), 1013–1021.

Sun, R., Tatsumi, S., & 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* (Vol. 3, pp. 1612–1617).

Sutton, R., & Barto, A. (1998). *Reinforcement learning: An introduction* (1st ed.). Cambridge, MA: MIT Press.

Tokic, M., Schwenker, F., & Palm, G. (2013). Meta-learning of exploration and exploitation parameters with replacing eligibility traces. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8183 LNAI, pp. 68–79).

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

Yoshida, N., Uchibe, E., & Doya, K. (2013). Reinforcement learning with state-dependent discount factor. In *2013 IEEE third joint international conference on development and learning and epigenetic robotics (ICDL)* (pp. 1–6).