Singular Perturbation-based Reinforcement Learning of Two-Point Boundary Optimal Control Systems

Vasanth Reddy, Hoda Eldardiry and Almuatazbellah Boker

***Abstract*— We solve the problem of two-point boundary optimal control of linear time-varying systems with unknown model dynamics using reinforcement learning. Leveraging singular perturbation theory techniques, we transform the time-varying optimal control problem into two time-invariant subproblems. This allows the utilization of an off-policy iteration method to learn the controller gains. We show that the performance of the learning-based controller approximates that of the model-based optimal controller and the approximation accuracy improves as the control problem's time horizon increases. We also provide a simulation example to verify the results.**

***Index Terms*— optimal control; singular perturbation; reinforcement learning**

I. Introduction

The control and analysis of time varying (dynamic) systems have been extensively explored over the last decade. In this work, we focus on finite-horizon time-varying systems. The difficulty level required to solve time-variant equations is significantly more complex when compared to the time-invariant equations. Previous work [1] illustrates two-value boundary problems exhibiting a two-time scale phenomenon, even when the original system is not singularly perturbed; in which case, their work makes it possible to approximate the original time-varying system into two time-invariant systems. The two systems are the initial boundary problem which stabilizes the original system in forward time and the final boundary system which stabilizes the terminal layer in reverse time. As the control problem time-horizon perturbation parameter decreases, the approximation accuracy increases. The approximation solution for the two-value boundary is provided if the dynamics of the model is known. However, in real-time applications, the dynamics may remain unknown or even suffer from modeling uncertainties. In these cases, it is difficult to solve the initial and final boundary problems. Previous research work introduced adaptive control laws through which we can learn the control of the model from the input-output data. Over the years, the gain in momentum of Reinforcement Learning has opened up the different possibilities of learning the controller. Adaptive Dynamic Programming is one of the first learning techniques in Reinforcement Learning. The latter uses an iteration method proposed in previous work [2] with a change in variables to find the optimal control gain. Learning methods such as Q-learning, Actor-Critic have been effective in learning the controller for continuous time systems when the model dynamics are unknown.

The main objective of this paper is to learn the controller for the finite horizon time varying system when the boundary conditions are given. Limited work has been done and proportionately less results are available for finite horizon time varying systems. In order to address the above problems, we have adopted the idea of solving the finite horizon time varying problems with given boundary conditions from seminal work [5]. The main concept is when the time period in which the cost function which needs to be optimized is large, the time-varying system will exhibit a two-time scale property, where the system dynamics will evolve at faster time scale relative to the time-scale of the control effort. Then [5] shows that the system can be reduced into two time-invariant problems. The final original state can thus be approximated by superimposing the solutions of initial and final layer problems. We make further use of the scheme of offline learning [4] to estimate the control gain of two boundary problems. Therefore, by reducing the complexity of the original system into two simple problems, we will learn the controllers of both problems. We both empirically and theoretically show that the learned controller (from the use of offline learning) is approximately the same as the original controller. We also show the accuracy of the state and controller which increases as the time interval to optimize the cost function increases. We can demonstrate the above point by taking up a linear time-variant system example.

In summary, the key contributions of our proposed model are twofold: 1.We provide a solution for the two-boundary optimal control problem for linear time-varying systems that does not require knowledge of system models.
2.We introduce a reinforcement learning framework that is based on an insight from the physical dynamics. This framework is simple to implement and results in a suboptimal solution, which converges to the optimal one as the control time interval gets large.

II. Problem Formulation

Consider the linear time-varying system expressed by the differential equation:

$$\frac{dx}{dt} = A(t)x(t) + B(t)u(t) \tag{1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $A(t) \in \mathbb{R}^{n\times n}$ and $B(t) \in \mathbb{R}^{n\times m}$ are the system states, control input, state matrix and

Vasanth Reddy and Hoda Eldardiry are with Department of Computer Science, Virginia Tech, Blacksburg, VA, USA. emails: vasanth2608@vt.edu and hdardiry@vt.edu

Almuatazbellah Boker is with Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA. email: boker@vt.edu

input matrix, respectively. The matrices $A(t)$ and $B(t)$ are smooth functions of time $\forall\, t \in [0, T]$ and can be unknown. The control objective is to design $u(t) \in \mathbb{R}^m$ to drive the system states $x(t)$ from the initial state $x(0) = x_0$ to the final state $x(T) = x_T$ within a time period T and meanwhile minimize the objective function

$$J = \int_0^T x^\top(t)Q(t)x(t) + u^\top(t)R(t)u(t)\ \ dt \tag{2}$$

where, $Q(t) = Q^\top(t) \succeq 0$ and $R(t) \succ 0\ \ \forall\, t \in [0, T]$. The matrices $Q(t)$ and $R(t)$ are assumed to be smooth functions of time. We further have the following assumptions:

Assumption 1. *The pair $(A(t), B(t))\quad \forall\ t \in [0, T]$ is controllable and the pair $(A(t), \sqrt{Q(t)})\quad \forall\ t \in [0, T]$ is observable.*

Assumption 2. *The time T needed to accomplish the control objective is long compared to the system dynamics.*

Assumption 1 is a standard assumption according to the theory of optimal control [6]. Assumption 2 implies that the system is slowly varying relative to the control objective. This will be explained further in the next section. To solve the optimal control problem, we define the Hamiltonaian function for the system (1) as [6]:

$$\begin{aligned}\mathcal{H} = x^\top(t)Q(t)x(t) + u^\top(t)R(t)u(t)+ \\ p^\top(t)(A(t)x(t) + B(t)u(t))\end{aligned} \tag{3}$$

where, $p(t) \in \mathbb{R}^n$ is the co-state equation that satisfies:

$$\dot{p}(t) = -\nabla_x \mathcal{H} = -Q(t)x(t) - A^\top(t)p(t). \tag{4}$$

The control law $u(t)$ for the dynamic system (1) that minimizes the objective function (2) is given by $\nabla_u \mathcal{H} = 0$:

$$u(t) = -R^{-1}(t)B^\top(t)p(t). \tag{5}$$

Our objective is to learn the optimal controller $u(t)$ given by (5) for the system (1) without the need to know the system matrices $A(t)$ and $B(t)$.

III. Singular Perturbation-based Design

It is known that for control problems that are formulated over a finite time interval $t \in [0, T]$, the closed-loop system dynamics behave in two time scales [7]. The separation of time scales is dependent on the length of the time interval T. In this work, we leverage this phenomenon by considering the case when the time needed to transfer the system state from one instant to another is small compared with T. This allows the system to be modeled in a singularly perturbed form. Accordingly, previous work [7] has shown that it is possible to achieve a sub-optimal solution to the control problem when T is sufficiently long and matrices $A(t)$ and $B(t)$ are known. In the next sections, we show that it is possible to achieve a similar result without requiring the system matrices to be known thanks to employing a reinforcement learning algorithm. Towards this goal, we start by setting up the singular perturbation model of the system. We first normalize the time period 0 to T to the interval $[0, 1]$ by introducing the scaled time and defining ε as

$$\tau = \frac{t}{T}, \quad \varepsilon = 1/T \tag{6}$$

and then reconstructing (1), (2), (4), and (5) to obtain

$$\varepsilon \begin{bmatrix} \frac{dx}{d\tau} \\ \frac{dp}{d\tau} \end{bmatrix} = \begin{bmatrix} A(\tau) & 0 \\ -Q(\tau) & -A^T(\tau) \end{bmatrix} \begin{bmatrix} x \\ p \end{bmatrix} + \begin{bmatrix} B(\tau) \\ 0 \end{bmatrix} u, \tag{7}$$

$$u(\tau) = -R^{-1}(\tau)B^\top(\tau)p(\tau), \tag{8}$$

$$J = T * \int_0^1 x^\top(\tau)Q(\tau)x(t) + u^\top(\tau)R(\tau)u(\tau)\ \ dt, \tag{9}$$

where $x(0) = x_0$ and $x(1) = x_T$. For the system described by (7) to be in standard singular perturbation form, we make the following assumption:

Assumption 3. *The eigenvalues of the Hamiltonian matrix*

$$M_H \triangleq \begin{bmatrix} A(\tau) & -B(\tau)R^{-1}(\tau)B^\top(\tau) \\ -Q(\tau) & -A^\top(\tau) \end{bmatrix} \tag{10}$$

lie off the imaginary axis $\forall\, t \in [0, 1]$.

This assumption implies that M_H, which is the closed-loop matrix after substituting (8) in (7), is nonsingular $\forall\, t \in [0, 1]$. We are going next to decouple the dynamics of (7). To accomplish this task, we use the transformation [7]

$$\begin{bmatrix} x \\ p \end{bmatrix} = \begin{bmatrix} I & I \\ P_a(\tau, \varepsilon) & P_b(\tau, \varepsilon) \end{bmatrix} \begin{bmatrix} x_a \\ x_b \end{bmatrix}. \tag{11}$$

It is shown in [Lemma 2.3, [7]] that, under the assumptions of this paper, the transformation (11) is nonsingular for sufficiently small ε. Accordingly, using (11), system (7) with (8) can be transformed into

$$\varepsilon \frac{d}{d\tau} x_a = A(\tau)x_a + B(\tau)u_a(\tau), \tag{12}$$

$$\varepsilon \frac{d}{d\tau} x_b = A(\tau)x_b + B(\tau)u_b(\tau), \tag{13}$$

where

$$u_a = -R^{-1}(\tau)B^\top(\tau)P_a(\tau, \varepsilon), \tag{14}$$

$$u_b = -R^{-1}(\tau)B^\top(\tau)P_b(\tau, \varepsilon), \tag{15}$$

and $P_a(\tau, \varepsilon) \geq 0$ and $P_b(\tau, \varepsilon) \leq 0$ are the roots of the differential Riccati equation

$$\begin{aligned}\varepsilon \dot{P} = &- A^\top(\tau)P - PA(\tau) \\ &+ PB(\tau)R^{-1}(\tau)B^\top(\tau)P - Q(\tau).\end{aligned} \tag{16}$$

Remark 1. *Let $\epsilon \to 0$, in* (16) *so that we have*

$$-A^\top(\tau)P - PA(\tau) + PB(\tau)R^{-1}(\tau)B^\top(\tau)P - Q(\tau) = 0. \tag{17}$$

It is shown in [7] that the solution of (17) *does exist and that the closed loop matrix $A(\tau) - B(\tau)R^{-1}(\tau)B^\top(t)P(\tau)$ is Hurwitz for each $\tau \in [0, 1]$. Furthermore, the solution has two roots of the form $P = P_a(\tau) \geq 0$ and $P = P_b(\tau) \leq 0$.*

We transform next the singular perturbation system (12)-(13) into the boundary layer system by considering the time-scale change

$$\gamma = \frac{\tau}{\varepsilon}, \quad \beta = \frac{1-\tau}{\varepsilon}. \tag{18}$$

In view of (18) and Remark 1, the boundary layer system is obtained by setting $\varepsilon \to 0$ in (12)-(16). This leads to the *initial regulator problem*

$$\frac{d}{d\gamma}x_a = A(0)x_a + B(0)u_a, \quad x_a(0) = x_0, \tag{19}$$

with the feedback controller u_a in the form

$$u_a(\gamma) \triangleq -K_a x_a(\gamma) = -R^{-1}(0)B^\top(0)P_a(0)x_a(\gamma), \tag{20}$$

which minimizes the cost function

$$J(x_a, u_a) = \int_0^\infty x_a^\top Q(0)x_a + u_a^\top R(0)u_a \, d\gamma, \tag{21}$$

and the *terminal regulator problem*

$$\frac{d}{d\beta}x_b = -A(1)x_b - B(1)u_b, \quad x_b(1) = x_T \tag{22}$$

with the feedback controller

$$u_b(\beta) \triangleq -K_b x_b(\beta) = -R^{-1}(1)B^\top(1)P_b(1)x_b(\beta), \tag{23}$$

which minimizes the cost function

$$J(x_b, u_b) = \int_0^\infty x_b^\top Q(1)x_b + u_b^\top R(1)u_b \, d\beta. \tag{24}$$

It is shown in Theorem 2.1 in [7] that if the initial and terminal regulator problems, which are now problems for linear time-invariant systems, are solved then the solution will approximate that of the original control problem (1)-(5) for sufficiently small ε provided that the dynamic model is known. In the next section, we develop a reinforcement learning algorithm to solve the initial and terminal regulator problems (19)-(24) without the need for $A(t)$ and $B(t)$ to be known.

IV. Main Results

A. Learning-based Design

In this section, we follow a reinforcement learning approach to learn the initial and terminal regulator problems (19)-(24). Guided by singular perturbation theory, we use the original state $x(t)$ for the learning procedure. It should be noted that, according to (6) and (18), the time-scales γ and β are the forward and reverse times of t, respectively. We will next solve the initial and terminal learning regulator problems separately.

1) Initial Regulator Learning Problem: The objective is to learn the feedback control gain $K_a \triangleq R^{-1}(0)B^\top(0)P_a(0)$ for the system described in (19) without knowing the system dynamics. Learning is done using the measurement data of system states x and control u_a such that control $u_a = -K_a x$ optimizes the cost function described in (21). Note that $P_a(0)$ is the solution to the algebraic Riccati equation:

$$A^\top(0)P_a(0) + P_a(0)A(0) - P_a(0)B(0)R^{-1}(0)B^\top(0)P_a(0) + Q(0) = 0. \tag{25}$$

where P_a^* is the solution of the Riccati equation (25). The optimal feedback gain in this case is given by $K_a^* = -R^{-1}(0)B^\top(0)P_a^*$. The optimal values are then found using the Kleimanns algorithm [3] as follows.

1) Solve for P^k of the Lyapunov equation

$$A_k^\top(0)P_a^k + P_a^k A_k(0) + Q(0) + P_a^k B(0)R^{-1}(0)B^\top(0)P_a^k = 0. \tag{26}$$

2) Update the feedback gain:

$$K_a^{k+1} = R^{-1}(0)B^\top(0)P_a^k, \tag{27}$$

where $A_k(0) = A(0) - B(0)K_a^k$. The matrix $P_a(0)^k$ and the gain K_a^{k+1} can be learned iteratively following the above steps but only when the model dynamics are known. As we assumed that the model dynamics are unknown, we follow a few steps to eliminate the use of model dynamics $A(0)$ and $B(0)$ in learning the controller.

Initialization: Consider an arbitrary control signal u_0 to excite the system (19). Accordingly, we define the control policy $u_a = u_0 - K_a^k x_a + K_a^k x_a$ with $K_a^k > 0$ and feed it back to (19) to get:

$$\begin{aligned} \dot{x}_a &= A(0)x_a + B(0)(u_0 - K_a^k x_a + K_a^k x_a) \\ &= A_k(0)x_a + B(0)(u_0 + K_a^k x_a). \end{aligned} \tag{28}$$

Towards eliminating the dependence of $A(0)$ and $B(0)$, we define the Lyapunov function $V^k(x_a) = x_a^\top P_a^k x_a$. Taking the derivative of this function along the system (28) leads to

$$\frac{d}{dt}(x_a^\top P_a^k x_a) = x_a^\top (A_k^\top(0)P_a^k + P_a^k A_k(0))x_a + 2(u_0 + K_a^k)^\top B(0)^\top P^k x_a. \tag{29}$$

Replacing $-Q_k(0) = (A_k^\top(0)P_a^k + P_a^k A_k(0)) = -Q(0) - K^\top R(0)K$ from (26) and $B^\top(0)P^k = R(0)K_a^{k+1}$ from (27) leads to

$$\frac{d}{dt}(x_a^\top P_a^k x_a) = -x_a^\top Q_k(0)x_a + 2(u_0 + K_a^k)^\top R(0)K_a^{k+1}x_a \tag{30}$$

Notice that (30) is independent of $A(0)$ and $B(0)$.
Integrating both sides of (30) on the interval $[t, t+\delta t]$, rearranging the terms and some math manipulation leads to

the offline policy iteration equation

$$\begin{aligned} x_a^\top(t+\delta t)P_a^k x_a(t+\delta t) - x_a^\top(t)P_a^k x_a(t) \\ - 2\int_t^{t+\delta t} \left(K_a^k x_a + u_0\right)^T R(0)K_a^{k+1}x_a\, dw \\ = -\int_t^{t+\delta t} x_a^\top Q_k(0)x_a\, dw \end{aligned} \tag{31}$$

By using some math manipulation we express the offline policy iteration in a compact form as:

$$\tilde{\psi}\begin{bmatrix} \text{vec}\left(P_a^k\right) \\ \text{vec}\left(K_a^{k+1}\right) \end{bmatrix} = \tilde{\Gamma} \tag{32}$$

where,
$\tilde{\psi} = \left[\tilde{\delta}_{xx}, -2\tilde{I}_{xx}\left(I_n \otimes \left(K^k\right)^\top R(0)\right) - 2\tilde{I}_{x\bar{u}_0}\left(I_n \otimes R\right)\right]$,
$\tilde{\delta}_{xx} = \left[x_a^\top \otimes x_a^\top\big|_t^{t+\delta t}\right]$,
$\tilde{\Gamma} = \tilde{\delta}_{x,x}\text{vec}(Q_k(0)$,
$\tilde{I}_{xx} = -2\left[\left(\int_{t_1}^{t_1+\tilde{t}} x_a^\top \otimes x_a^\top dw\right)\left(I_n \otimes (K^k)^\top R(0)\right)\right]$ and
$\tilde{I}_{xu_0} = -2\left[\left(\int_{t_1}^{t_1+\tilde{t}} x_a^\top \otimes u_0^\top dw\right)\left(I_n \otimes R(0)\right)\right]$. This way offline policy iteration is expressed in compact form (32). It should be emphasized that the learning is going to be done through the use of the state $x(t)$. In the next step, we collect this data to estimate the feedback control gain.

Data Collection: During learning, we collect data, including state space $x(t)$ and control policy u_0, for the time period $[t_i, t_j]$ with the sampling interval $t_{i+1} - t_i = \delta t = \tilde{t}$. This is followed by computing the matrices δ_{xx}, I_{xx} and I_{xu_0} as follows:

$$\delta_{xx} = \left[x^\top \otimes x^\top\big|_{t_1}^{t_1+\tilde{t}}, \quad \ldots, \quad x^\top \otimes x^\top\big|_{t_j}^{t_j+\tilde{t}}\right]^\top, \tag{33}$$

$$I_{xx} = \left[\int_{t_1}^{t_1+\tilde{t}} x^\top \otimes x^\top dw, \ldots, \quad \int_{t_l}^{t_l+\tilde{t}} x^\top \otimes x^\top dw\right]^\top, \tag{34}$$

$$I_{xu_0} = \left[\int_{t_1}^{t_1+\tilde{t}} x^\top \otimes u_0^\top dw, \ldots, \quad \int_{t_l}^{t_l+\tilde{t}} x^\top \otimes u_0^\top dw\right]^\top. \tag{35}$$

Assumption 4. *There exists a large number $j > 0$ such that rank($[I_{xx} \quad I_{xu_0}]$) = $\frac{n(n+1)}{2} + mn$.*

Assumption 4 ensures the collection of enough data for the learning process [4]. **Policy Iteration**: This step further involves two sub-steps. i) Policy evaluation; Evaluating the effectiveness of the current policy using the data from the present data step. and ii) Policy improvement; After we evaluate the current policy we then update the past policy with this new one. These two steps will be performed simultaneously until the threshold is met as shown in the Algorithm 1. At the end of convergence, the feedback gain can be obtained as $K_a = K_a^{k+1}$ and the controller for the system (19) is $u_a^l = -K_a x$.

2) Terminal Regulator Learning Problem: We follow the same steps as in the initial regulator problem except the initialization of the control gain should satisfy $K_b < 0$. Note that controller for the system (22) is $u_b^l = -K_b^* x$, where K_b^* is the feedback gain matrix. The pseudocode for the learning algorithm is given in Algorithm 1.

Algorithm 1: Offline policy iteration on two value boundary problems

while $rank([I_{xx} \quad I_{xu_0}]) < \frac{n(n+1)}{2} + mn$ **do**
  Collect the data $x(t)$ using the excitation control $u_a = u_0$;
  Construct the matrices δ_{xx}, I_{xx} and I_{xu_0}
end
Initialize $K_a > 0$;
while $|P_a^k - P_a^{k+1}| <$ *Threshold* **do**
  Estimate the values of P_a^k and K_a^{k+1} through (32)
end
$u_a^l = -K_a^{k+1}x$;
Initialize $K_b < 0$;
while $|P_b^k - P_b^{k+1}| <$ *Threshold* **do**
  Estimate the values of P_b^k and K_b^{k+1} through (32)
end
$u_b^l = -K_b^{k+1}x$

Following the learning procedure described in this section, the closed-loop performance of the learned-control system, comprised of (1) and $u_{\text{learned}} = u_a^l + u_b^l$, will approximate that of the original control system, comprised of (1) and (5), provided that ε is sufficiently small or T is sufficiently long. This will be verified in Theorem 1 given next and through the simulation example given in the Section V

B. Analysis of the closed-loop system performance

Consider the closed-loop system, based on the time-varying linear system (1) and the learning-based controllers K_a^* and K_b^* obtained through Algorithm 1, that can be written as

$$\frac{d}{d\gamma}x_a^l = (A(0) - B(0)K_a^*)x_a^l, \quad x_a^l(0) = x_0, \tag{36}$$

$$\frac{d}{d\beta}x_b^l = (A(1) - B(1)K_b^*)x_b^l, \quad x_b^l(1) = x_T. \tag{37}$$

Recall that γ and β are the forward and reverse times as defined in (18). Now we have the following theorem.

Theorem 1. *Under Assumptions 1-4, there exists $\varepsilon_1 > 0$ such that for all $\varepsilon \in (0, \varepsilon_1]$, the solution $x(\tau)$ of* (7)-(8) *satisfies*

$$x(\tau) = x_a^l(\gamma) + x_b^l(\beta) + \mathcal{O}(\varepsilon), \tag{38}$$

$$\begin{aligned} u(\tau) =& -R^{-1}(0)B^\top(0)P_a(0)x_a^l(\gamma) \\ & - R^{-1}(1)B^\top(1)P_b(1)x_b^l(\beta) + \mathcal{O}(\varepsilon), \\ \triangleq & \; u_a(\gamma) + u_b(\beta) + \mathcal{O}(\varepsilon) \end{aligned} \tag{39}$$

where x_a^l and x_b^l are the solutions of the closed-loop systems (36) *and* (37)*, respectively.* [1]

Remark 2. *Theorem 1 describes how the performance of the closed loop system* (36)-(37) *approximate the performance of the closed loop system* (7)-(8) *under optimal control. Moreover, equation* (39) *describes how the controllers of the initial and terminal learning regulator problems approximate the optimal control. All these results show that the performance of the learned controllers provide a sub-optimal performance and that the closed-loop performance gets closer to the optimal one as ϵ gets small (or $1/T$ gets large).*

Towards proving Theorem 1, we are going next to present a couple of results that show relative convergence and performance of the learned closed-loop system.

Lemma 2. *At the end of the learning process according to Algorithm 1, the matrices P_a^k and P_b^k and the feedback control gains K_a^k and K_b^k converge as follows:*

$$\lim_{k\to\infty} K_a^k = K_a^*, \quad \lim_{k\to\infty} P_a^k = P_a^*, \tag{40}$$

$$\lim_{k\to\infty} K_b^k = K_b^*, \quad \lim_{k\to\infty} P_b^k = P_b^*, \tag{41}$$

where K_a^* and P_a^* are the optimal controller gain and Riccati solution of the initial and terminal regulator problems (19)-(21) *and* (22)-(24)*, respectively.*

Proof. Kleinman's algorithm [2] is recalled below to show the convergence of K_a^* and P_a^*.
Let P_a^0 be the finite and positive definite solution of the Lyapunov equation (26) at $k = 0$, which is given as

$$P_a^0 = \int_0^\infty e^{(A_0^\top(0))t}\left(Q(0) + (K_a^0)^\top R(0) K_a^0\right) e^{(A_0(0))t} dt. \tag{42}$$

and for which the updated feedback control gain at $k = 1$ is given by $K_a^1 = R^{-1}B^\top P_a^0$. Similarly P_a^1 is the solution at $k = 1$, which is given by

$$P_a^1 = \int_0^\infty e^{(A_1^\top(0))t}\left(Q(0) + (K_a^1)^\top R(0) K_a^1\right) e^{(A_1(0))t} dt. \tag{43}$$

Subtracting equation (43) from (42) gives

$$P_a^0 - P_a^1 = \int_0^\infty e^{(A_1^\top(0))t}\left(K_a^0 - K_a^1\right)^\top R\left(K_a^0 - K_a^1\right) e^{(A_1(0))t} dt \geq 0. \tag{44}$$

[1] The symbol $\mathcal{O}$ denotes the "order of magnitude" and is defined as: $\delta_1(\varepsilon) = \mathcal{O}(\delta_2(\varepsilon))$ if there exist positive constants k and c such that $|\delta_1(\varepsilon)| \leq k\,|\delta_2(\varepsilon)|\,,\ \forall\ |\varepsilon| < c$.

From the above equation we see that $P_a^0 \geq P_a^1$. Similarly

$$P_a^1 - P_a^* = \int_0^\infty e^{(A_1^\top(0))t}\left(K_a^1 - K_a^*\right)^\top R\left(K_a^1 - K_a^*\right) e^{(A_1(0))t} dt \geq 0. \tag{45}$$

From the equations (44) and (45) we can see that $P_a^0 \geq P_a^1 \geq P_a^*$. We can observe that the sequence P_a^k is monotonically decreasing and is lower bounded by P_a^*. Thus the convergence value of P_a^k at $k = \infty$ is given by

$$\lim_{k\to\infty} P_a^k = P_a^*. \tag{46}$$

The feedback control gain converges as

$$\lim_{k\to\infty} K_a^k = \lim_{k\to\infty} R^{-1}(0)B^\top(0)P_a^k = R^{-1}(0)B^\top(0) \lim_{k\to\infty} P_a^k \tag{47}$$

Using equation (46) we can deduce that

$$\lim_{k\to\infty} K_a^k = R^{-1}(0)B^\top(0)P_a^* = K_a^*. \tag{48}$$

Similarly, we can show the convergence for the final regulator problem. □

Using the above lemmas, the proof for Theorem 1 is as follows.

Proof of Theorem 1. According to Theorem 6.1 in Chapter 5 of [7], there exists $\varepsilon_3 > 0$ such that for all $\varepsilon \in (0, \varepsilon_3]$ and for all $t \in [0, 1]$, the solutions of (12)-(13) with (14)-(15) and (19) with (20) and (22) with (23) are related by

$$x_a(\tau) = x_a(\gamma) + \mathcal{O}(\varepsilon), \quad x_b(\tau) = x_b(\beta) + \mathcal{O}(\varepsilon) \tag{49}$$

Now because of the convergence results stated in Lemma 2, we have

$$x_a(\gamma) = x_a^l(\gamma), \quad \text{and} \quad x_b(\beta) = x_b^l(\beta),$$

where $x_a^l(\gamma)$ and $x_b^l(\beta)$ are the solutions (36) and (37), respectively. This implies

$$x_a(\tau) = x_a^l(\gamma) + \mathcal{O}(\varepsilon), \quad x_b(\tau) = x_b^l(\beta) + \mathcal{O}(\varepsilon). \tag{50}$$

In the view of the transformation (11), we have

$$x(\tau) = x_a(\tau) + x_b(\tau). \tag{51}$$

Substituting the x_a and x_b from (50) into (51) gives (38). From the transformation (11), we have

$$p(\tau) = P_a(\tau, \varepsilon)x_a(\tau) + P_b(\tau, \varepsilon)x_b(\tau) \tag{52}$$

Substituting by (52) in (8), invoking Lemma 2 and using (50) leads to

$$u(\tau) = -R^{-1}(0)B^\top(0)P_a(0)x_a^l(\gamma) - R^{-1}(1)B^\top(1)P_b(1)x_b^l(\beta) + \mathcal{O}(\varepsilon) \tag{53}$$

By taking $\varepsilon_1 = \min(\varepsilon_2, \varepsilon_3)$, we conclude the proof. □

V. Simulation Example

Consider an electric circuit with the components: resistance (R) and inductor ($L(t)$) in series connection. The state space equation for the circuit is given as:

$$\frac{dx}{dt} = -\frac{R}{L(t)}x + \frac{1}{L(t)}u, \tag{54}$$

$$\dot{x_1} = x_2 \tag{55}$$

$$\dot{x_2} = -\frac{1}{1+0.2t}x_1 - x_2 + u \tag{56}$$

where $x(t)$ is the state (circuit current) and $u(t)$ is the control input (circuit voltage). In this case, $A(t) = -\frac{R}{L(t)}$ and $B(t) = \frac{1}{L(t)}$. We assume that the time-varying dissipating inductor L and resistor values are unknown. A controller $u(t)$ is sought to set the current $x(t)$ at desired values both at the initial and final times while keeping the current and voltage minimum in between. That is, the controller is needed to minimize the objective function while guaranteeing the initial and final conditions of the system as: $x_0 = 0.5$ and $x_T = 0.9$. The cost function parameters are specified as $q(t) = 1$, $r(t) = 1$. We are going to solve the problem for different final time T. This is to show that as T becomes large, the approximation result will be more accurate as indicated in Theorem 1. For simulation purposes, we assume that the inductor is given by 0.. During the offline data collection, we excite the system using the signal $u_0 = \sum_0^{100} sin(wt)$, where w is random value between 1 to 100, and then we collect the date at time interval of 0.01 seconds. The control gains K_a and K_b are found using the learning method described in Section IV-A and found to be 0.4 and -2.4, respectively.

To compare our results, we found K_a and K_b values assuming that $A(t)$ and $B(t)$ are known through the procedure outlined in Section IV by solving the problems (19)-(24).

The simulations of the learning- and singular perturbation-based controllers were done by solving the system differential equations forward and backward in time for the initial and terminal regulator problems, respectively, and then adding the responses as shown in Theorem 1. Note that we are normalizing the time interval to $[0, 1]$ instead of $[0, T]$ to make the comparison between the controllers clear. The optimal state trajectory and the optimal controller is found using the two value boundary problem solver from MATLAB command *bvp4c* using the knowledge of $A(t)$ and $B(t)$. We then compare the state space trajectory and the controller obtained using our learning method with that of the approximate method based on singular perturbation and the optimal controller with $A(t)$ and $B(t)$ known.

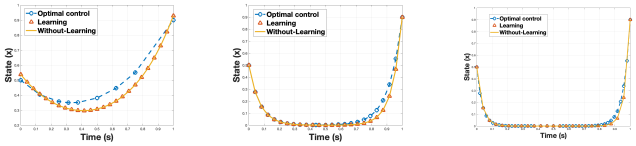

Fig. 1: Plots illustrate the state space trajectory for the state $x(t)$ for different values of $\varepsilon = 0.5, 0.1$ and 0.05 respectively.

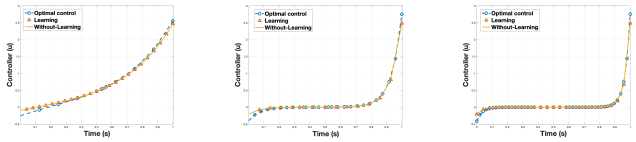

Fig. 2: Plots illustrates the control law $u(t)$ for different values of $\varepsilon = 0.5, 0.1$ and 0.05 respectively.

From Fig. 1 and Fig. 2, we can see that the learning-based controller accurately approximates the optimal controller as ε is decreased (or control time period T increases). This implies that the performance of the learning controller is a sub-optimal one and converges to the optimal performance as the time interval T gets large.

VI. Conclusions and Future Works

We proposed optimal controller design using reinforcement learning for time varying systems with two boundary conditions. The proposed design leverages the fast time scale occurring at the boundary conditions to reduce the time-varying problem into two simple time-invariant problems. Furthermore, we design a learning-based control strategy that does not need knowledge of the system model. We show that the accuracy of the controller performance improves as the problem time horizon increases. We presented simulation results to support our claims using an RL circuit. In the future, we plan to extend our work in learning the controllers of nonlinear systems.

References

[1] Wilde, R., and P. Kokotovic. "A dichotomy in linear control theory." IEEE Transactions on Automatic control 17.3 (1972): 382-383.

[2] Kleinman, David. "On an iterative technique for Riccati equation computations." IEEE Transactions on Automatic Control 13.1 (1968): 114-115.

[3] Kleinman, David. "On an iterative technique for Riccati equation computations." IEEE Transactions on Automatic Control 13, no. 1 (1968): 114-115.

[4] Jiang, Yu, and Zhong-Ping Jiang. Robust adaptive dynamic programming. John Wiley & Sons, 2017.

[5] Wilde, R., & Kokotovic, P. (1972). A dichotomy in linear control theory. IEEE Transactions on Automatic control, 17(3), 382-383.

[6] Athans, Michael, and Peter L. Falb. Optimal control: an introduction to the theory and its applications. Courier Corporation, 2013.

[7] Kokotović, Petar, Hassan K. Khalil, and John O'reilly. Singular perturbation methods in control: analysis and design. Society for Industrial and Applied Mathematics, 1999.