Optimal Robot-Environment Interaction Using Inverse Differential Riccati Equation

Hamed Rahimi Nohooji[1,*], Ian Howard[2], and Lei Cui[3]

[1] Faculty of Science and Engineering, School of Civil and Mechanical Engineering, Curtin University, Perth, Australia; hamed.rahiminohooji@postgrad.curtin.edu.au

[2] Faculty of Science and Engineering, School of Civil and Mechanical Engineering, Curtin University, Perth, Australia; i.howard@curtin.edu.au

[3] Faculty of Science and Engineering, School of Civil and Mechanical Engineering, Curtin University, Perth, Australia; lei.cui@curtin.edu.au

***** Correspondence: hamed.rahiminohooji@postgrad.curtin.edu.au; Tel.: +61 435 456 213

Abstract: An optimal robot-environment interaction is designed by transforming an environment model into an optimal control problem. In the optimal control, the inverse differential Riccati equation is introduced as a fixed-end-point closed-loop optimal control over a specific time interval. Then, the environment model, including interaction force is formulated in a state equation, and the optimal trajectory is determined by minimizing a cost function. Position control is proposed, and the stability of the closed-loop system is investigated using the Lyapunov direct method. Finally, theoretical developments are verified through numerical simulation.

Key Words: Inverse differential Riccati equation, Lyapunov direct method, Optimal control, Robot-environment interaction, Trajectory tracking.

1. INTRODUCTION

2.1. Related Works and Motivation

Robot-Environment Interaction (REI) has been theoretically studied over the last two decades and its development exhibits great popularity in recent robotic studies. Accordingly, demand for research in control of robots that interact with environments has increased. In many conventional interaction tasks, such as repetitive applications in construction or in industrial factories, the robot is expected to track a predefined task trajectory. However, in many of the recent applications, robots are likely working with initially undefined task trajectories. This brings along several challenges to control engineers.

In addition to its conventional industrial applications, REI control is becoming a challenging topic in social research issues. It can address emerging aspects of rehabilitation robotics, surgery robotic systems, haptic rendering, and several fields in human-robot interaction systems [1]. Control in REI systems has been studied to cope with different problems like impedance adaptation[2], impedance learning [3], collaborative manufacturing [4], or assistive human-robot interaction [5-7]. However, in most of the research work on REI control, desired trajectories in the task space are given, then tracking problems are addressed [8, 9] whereas in several applications of REI, like pick-and-place operations, two-end points are given and the path should be planned according to the desired objective.

Optimal control in robotics refers to control design that causes the state trajectories for a dynamic system, satisfying some physical constraints followed by extremizing a chosen performance criterion. On the other hand, development of an optimal control theory for a linear dynamic equation along with a performance index with quadratic functions of state and control has led to the emergence of the linear quadratic regulator (LQR). Such regulators typically abound in cylindrical robotic arms [10], mobile robots [11], UAVs [12], missiles [13], wind turbine [14], and multi-agent systems [15]. Over the last few decades, LQR has been widely employed for various robotic applications as in manufacturing, mining, aerospace and medical engineering [16]. Nevertheless, a considerable amount of LQR research is carried out using infinite-time regulators applied in robotic systems. However, most of the planning strategies in real robotic systems are applied in a fixed execution time. To increase the efficiency of such controllers, finite-time LQR has been developed based on the differential Riccati equation [17-19]. Although these controllers have given rise to far-reaching mathematical developments [20, 21], they are designed to find solutions for problems in the free-end-point state regulator systems. However, there are various practical examples of optimal planning in engineering for which two fixed and non-zero final boundary conditions are required. On the other hand, several efficient numerical methods have been proposed for solving differential Riccati equation. For example, in [22] an algorithm developed for nonnegative, stabilizing solutions for the periodic Riccati differential equation based on Fourier series expansion and the precise integration method. Also, [23] combined Fourier series expansion with recursive matrix formulas to propose an algorithm for computing solutions of the periodic Riccati and Lyapunov matrix differential equations.

Motivated by the above-mentioned considerations, this paper addresses optimal REI by developing a fixed-end-point differential Riccati equation. A closed-loop optimal control solution is developed to minimize a cost function combining system states, and control input. By that means, a finite-time fixed-end-point optimal controller is obtained based on the inverse Differential Riccati Equation (iDRE). Environment dynamic models are formed in a state equation and using the obtained iDRE method, optimal interaction force, and optimal trajectories are obtained. Then, the obtained optimal trajectory is considered as the desired trajectory, and position control is proposed for tracking purpose. The Lyapunov direct method is utilized for the stability analysis. The developed controller is examined through a numerical simulation study.

2.2. Contributions and Structure of the Paper

The contribution of this paper can be highlighted as follows:

- Different from conventional LQR based methods, the presented iDRE approach can tackle planning problems with fixed, and no-zero end-point states. Hence, the presented method can be useful for robotic systems with any fixed boundaries within the desired execution time. Also, it should be noted that the paper considers complete robot nonlinear dynamics, thus linearization is avoided.
- In addition, compared with the Pontryagin maximum principle, which is a canonical tool for dealing with optimal control of nonlinear systems, the paper avoids solving tedious two-point boundary value problem which involves both states and co-states [24-27]. Also, as the presented method leads to closed-loop optimal control, it enjoys advantages of simplification of controller's hardware implementation.
- Compared to previous works on REI control like [8, 9], in that tracking of the given desired trajectories occurs in the task space, in this paper a path between two end points is planned according to a desired task cost function, and then position tracking is handled. Also, in the presented paper, to cope with an optimal REI problem, only environment properties are required. By that means the optimal trajectory can be obtained according to the task-specific information without requiring knowledge of the robot dynamics.

The rest of the paper is organized as follows. Section 2 reviews the kinematics and dynamics of the system model, and the environment model dynamics. The iDRE method is formed in Section 3. First, states and performance index are formulated, and the optimizing process is developed that leads to open loop optimal control. Then, the resultant control is converted to closed loop optimal control. In Section 4 the optimal trajectory and the optimal interaction force are obtained using iDRE, then the position tracking controller is proposed and stability of the closed-loop system is studied using the Lyapunov direct method. Verification of theoretical developments is done by numerical simulation in Section 5. The discussion is provided in Section 6 and finally, the paper is concluded in Section 7.

2. System Overview

2.1. Dynamic Model

A system where a robotic arm physically interacts with an environment is studied in this paper. The kinematics of the robotic system can be given by,

$$\boldsymbol{x}(\boldsymbol{t})=\boldsymbol{\varphi}(\boldsymbol{q}(\boldsymbol{t})), \tag{1}$$

where $\boldsymbol{x}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}_C}$, and $\boldsymbol{q}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}}$ are vectors of the end-effector Cartesian position, and generalized joint coordinates, respectively with $\boldsymbol{n}_c$ being the dimension of the Cartesian space, and $\boldsymbol{n}$ is the number of joints. Time differentiating of (1) results in [28],

$$\dot{\boldsymbol{x}}(\boldsymbol{t})=\mathbf{J}(\boldsymbol{q}(\boldsymbol{t}))\dot{\boldsymbol{q}}(\boldsymbol{t}), \tag{2}$$

where $\mathbf{J}(\boldsymbol{q}(\boldsymbol{t}))\in\mathbf{R}^{\boldsymbol{n}_C\times\boldsymbol{n}}$ is the Jacobian matrix. We consider the dynamic model of the robot manipulator as [28, 29]:

$$\begin{aligned}&\mathbf{H}(\boldsymbol{q}(\boldsymbol{t}))\ddot{\boldsymbol{q}}(\boldsymbol{t})+\mathbf{C}(\boldsymbol{q}(\boldsymbol{t}),\dot{\boldsymbol{q}}(\boldsymbol{t}))\dot{\boldsymbol{q}}(\boldsymbol{t})+\mathbf{G}(\boldsymbol{q}(\boldsymbol{t}))\\&=\boldsymbol{\tau}(\boldsymbol{t})+\mathbf{J}^{T}(\boldsymbol{q}(\boldsymbol{t}))\boldsymbol{f}_{e}(\boldsymbol{t}),\end{aligned} \tag{3}$$

where $\mathbf{H}(\boldsymbol{q}(\boldsymbol{t}))\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{n}}$, $\mathbf{C}(\boldsymbol{q}(\boldsymbol{t}),\dot{\boldsymbol{q}}(\boldsymbol{t}))\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{n}}$, and $\mathbf{G}(\boldsymbol{q}(\boldsymbol{t}))\in\mathbf{R}^{\boldsymbol{n}}$ denote the inertia, centrifugal and Coriolis force matrices, and the vector of gravitational forces/torques, respectively, and $\boldsymbol{\tau}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}}$ is the vector of generalized joint inputs, and $\boldsymbol{f}_{e}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}_C}$ represent the interaction forces between the environment and robot.

Assumption 1. The Jacobian matrix $\mathbf{J}(\boldsymbol{q}(\boldsymbol{t}))$ is assumed to be known and nonsingular in a finite workspace.

Property 1 [30]: The matrix $\mathbf{H}(\boldsymbol{q}(\boldsymbol{t}))$ is symmetric and positive definite. Furthermore, then the matrix $2\mathbf{C}(\boldsymbol{q}(\boldsymbol{t}),\dot{\boldsymbol{q}}(\boldsymbol{t}))-\dot{\mathbf{H}}(\boldsymbol{q}(\boldsymbol{t}))$ is a skew-symmetric matrix.

2.2. Environment Model

The environment can be modeled by [3],

$$\mathbf{M}_{\mathbf{e}}(\boldsymbol{t})\ddot{\boldsymbol{x}}_{d}(\boldsymbol{t})+\mathbf{C}_{\mathbf{e}}(\boldsymbol{t})\dot{\boldsymbol{x}}_{d}(\boldsymbol{t})+\mathbf{G}_{\mathbf{e}}(\boldsymbol{t})\boldsymbol{x}_{d}(\boldsymbol{t})=-\boldsymbol{f}_{e}(\boldsymbol{t}), \tag{4}$$

where $\boldsymbol{x}_{d}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}}$ is the desired end-effector trajectory in Cartesian coordinates; and $\mathbf{M}_{\mathbf{e}}(\boldsymbol{t})$, $\mathbf{C}_{\mathbf{e}}(\boldsymbol{t})$, and $\mathbf{G}_{\mathbf{e}}(\boldsymbol{t})$ are the mass, damping and stiffness parameter matrices of the environment model, respectively.

Assumption 2. The mass, damping and stiffness parameter matrices of the environment model are assumed to be known and time-varying matrices in this paper.

Note that the environment model (4) can represent a large range of environments [31]. For example, it can model the human limb in the case of physical human-robot interaction where $\mathbf{M}_{\mathbf{e}}(\boldsymbol{t})$, $\mathbf{C}_{\mathbf{e}}(\boldsymbol{t})$, and $\mathbf{G}_{\mathbf{e}}(\boldsymbol{t})$ represent mass, damper and spring matrix of the human limb, respectively [6], or the dynamics of a viscoelastic object in robotic manipulation [3].

2.3. Problem Statement

In several studies of REI, the desired trajectory, $\boldsymbol{x_d}(\boldsymbol{t})$, is prescribed by the designer. In that case, this trajectory can be available for control design generally based on a basic understanding of a task. Nevertheless, this trajectory assignment typically cannot guarantee a good performance due to the lack of flexibility [32]. In REI research under study in this paper, the desired trajectory is obtained optimally which is unknown in the control design. As discussed in the Introduction, iDRE is developed to cope with this problem. Then, position tracking control is proposed, and stability analysis of the closed-loop system is provided.

3. Inverse Differential Riccati Equation

3.1. Background

This section presents an inverse Riccati equation to find the closed loop optimal control for a linear system.

The non-zero fixed boundary conditions are given as,

$$\boldsymbol{X}(\boldsymbol{t}=\boldsymbol{t}_0)=\boldsymbol{X_0}\,; \quad \boldsymbol{X}(\boldsymbol{t}=\boldsymbol{t_f})=\boldsymbol{X_f}\,, \tag{5}$$

and the performance index with mixed state-control quadratic functions is formed as,

$$\mathrm{E}=\frac{1}{2}\int_{t_0}^{t_f}\begin{bmatrix}\boldsymbol{X}^{\boldsymbol{T}}(\boldsymbol{t})\mathbf{Q}\boldsymbol{X}(\boldsymbol{t})+2\boldsymbol{X}^{\boldsymbol{T}}(\boldsymbol{t})\mathbf{S}\boldsymbol{U}(\boldsymbol{t})\\ +\boldsymbol{U}^{\boldsymbol{T}}(\boldsymbol{t})\mathbf{R}\boldsymbol{U}(\boldsymbol{t})\end{bmatrix}\boldsymbol{dt}\,, \tag{6}$$

and the state equation for the system is defined by

$$\dot{\boldsymbol{X}}(\boldsymbol{t})=\mathbf{A}(\boldsymbol{t})\boldsymbol{X}(\boldsymbol{t})+\mathbf{B}(\boldsymbol{t})\boldsymbol{U}(\boldsymbol{t})\,. \tag{7}$$

In (5) to (7), $\boldsymbol{t_f}$ is a fixed final time, $\boldsymbol{X}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}}$ and $\boldsymbol{U}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{m}}$ are state and control vectors, respectively; $\mathbf{A}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{n}}$ is the system matrix, $\mathbf{B}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{m}}$ is the input matrix, $0\le\mathbf{Q}\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{n}}$, $0\le\mathbf{S}\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{m}}$, and $0<\mathbf{R}\in\mathbf{R}^{\boldsymbol{m}\times\boldsymbol{m}}$. Note that the details background on optimal control of linear systems can be found in [19, 33].

3.2. Optimization Problem

The equations of the optimal control problem can be initiated by formation of the Hamiltonian equations as,

$$\mathrm{H}(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda},\boldsymbol{t})=\frac{1}{2}\boldsymbol{X}^{\boldsymbol{T}}(\boldsymbol{t})\mathbf{Q}\boldsymbol{X}(\boldsymbol{t})+\boldsymbol{X}^{\boldsymbol{T}}(\boldsymbol{t})\mathbf{S}\boldsymbol{U}(\boldsymbol{t}) \\ +\frac{1}{2}\boldsymbol{U}^{\boldsymbol{T}}(\boldsymbol{t})\mathbf{R}\boldsymbol{U}(\boldsymbol{t})+\boldsymbol{\lambda}^{\boldsymbol{T}}(\boldsymbol{t})\left[\mathbf{A}(\boldsymbol{t})\boldsymbol{X}(\boldsymbol{t})+\mathbf{B}(\boldsymbol{t})\boldsymbol{U}(\boldsymbol{t})\right]. \tag{8}$$

This is followed by verifying the state and co-state vector equations and defining the minimality conditions for the Hamiltonian as [33, 34],

$$\dot{\boldsymbol{X}}^*(\boldsymbol{t})=\left(\frac{\partial\mathrm{H}(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda},\boldsymbol{t})}{\partial\boldsymbol{\lambda}(\boldsymbol{t})}\right)_*, \tag{9}$$

$$\dot{\boldsymbol{\lambda}}^*(\boldsymbol{t})=-\left(\frac{\partial\mathrm{H}(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda},\boldsymbol{t})}{\partial\boldsymbol{X}(\boldsymbol{t})}\right)_*, \tag{10}$$

$$0\quad=\left(\frac{\partial\mathrm{H}(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda},\boldsymbol{t})}{\partial\boldsymbol{U}(\boldsymbol{t})}\right)_*, \tag{11}$$

where the symbol $(*)$ denotes the optimality conditions and $\boldsymbol{\lambda}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}}$ is known as the co-state vector. From (11), the optimal control $\boldsymbol{U}^*(\boldsymbol{t})$ can be obtained as,

$$\boldsymbol{U}^*(\boldsymbol{t})=-\mathbf{R}^{-1}\left(\mathbf{S}^{\boldsymbol{T}}\boldsymbol{X}^*(\boldsymbol{t})+\mathbf{B}(\boldsymbol{t})^{\boldsymbol{T}}\boldsymbol{\lambda}^*(\boldsymbol{t})\right). \tag{12}$$

Eliminating optimal control (12) from (9), and (10), gives the following equation

$$\dot{\mathbf{Y}}^*=\mathbf{G}\mathbf{Y}^*, \tag{13}$$

where $\mathbf{Y}=\left[\boldsymbol{X}(\boldsymbol{t}),\boldsymbol{\lambda}(\boldsymbol{t})\right]^{\boldsymbol{T}}$, and

$$\mathbf{G}=\begin{bmatrix}\mathbf{A}(\boldsymbol{t})-\mathbf{B}(\boldsymbol{t})\mathbf{R}^{-1}\mathbf{S}^{\boldsymbol{T}} & -\mathbf{B}(\boldsymbol{t})\mathbf{R}^{-1}\mathbf{B}(\boldsymbol{t})^{\boldsymbol{T}}\\ -\mathbf{Q}+\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^{\boldsymbol{T}} & -\mathbf{A}(\boldsymbol{t})^{\boldsymbol{T}}+\mathbf{S}\mathbf{R}^{-1}\mathbf{B}(\boldsymbol{t})^{\boldsymbol{T}}\end{bmatrix}.$$

The state and co-state system (13) along with the boundary conditions given by (5) construct a two-point boundary value problem. Substituting the solution into (12) gives an open-loop optimal control formulation for the system. However, open-loop optimal control has some disadvantages, such as the inability to compensate for system changes and difficulties with a hardware implementation. Accordingly, this work focuses on finding closed-loop optimal control realization for the fixed-end-point system.

3.3. Closed-Loop Optimal Control

The Riccati transformation between the state and co-state functions is formed as,

$$\boldsymbol{\lambda}^*(\boldsymbol{t})=\mathbf{P}(\boldsymbol{t})\boldsymbol{X}^*(\boldsymbol{t})\,, \tag{14}$$

where $0<\mathbf{P}(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{n}}$ is the matrix Riccati coefficient. The Riccati transformation (14) is employed to obtain the differential Riccati equation. This equation was widely used for path planning of the system with free final endpoints [12, 15, 35]. To find the optimal control for the two fixed end-point system, we adopted the inverse Riccati transformation as in [36, 37] between the state and costate variables. By that means, we arrived at the matrix inverse differential Riccati equation to handle the closed loop path planning of a system in a finite time horizon.

In the absence of knowledge on final conditions of a co-state function, the inverse Riccati transformation between the state $\boldsymbol{X}^*(\boldsymbol{t})$ and co-state $\boldsymbol{\lambda}^*(\boldsymbol{t})$ can be defined as

$$\boldsymbol{X}^*(\boldsymbol{t})=\Xi(\boldsymbol{t})\boldsymbol{\lambda}^*(\boldsymbol{t})+\Psi(\boldsymbol{t})\,, \tag{15}$$

where $\Xi(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}\times\boldsymbol{n}}$ and $\Psi(\boldsymbol{t})\in\mathbf{R}^{\boldsymbol{n}}$ are yet to be determined. Substituting (15) in (13) and eliminating $\boldsymbol{X}^*(\boldsymbol{t})$ yields,

$$\dot{\boldsymbol{X}}^*(\boldsymbol{t})=\dot{\Xi}(\boldsymbol{t})\boldsymbol{\lambda}^*(\boldsymbol{t})+\Xi(\boldsymbol{t})\dot{\boldsymbol{\lambda}}^*(\boldsymbol{t})+\dot{\Psi}(\boldsymbol{t})\,, \tag{16}$$

which leads to,

$$\begin{aligned}&\left(\mathbf{A}(t)-\mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^{T}\right)\left(\Xi(t)\lambda^{*}(t)+\Psi(t)\right)\\&-\mathbf{B}(t)\mathbf{R}^{-1}\mathbf{B}(t)^{T}\lambda^{*}(t)=\dot{\Xi}(t)\lambda^{*}(t)+\Xi(t)\\&\begin{pmatrix}\left(-\mathbf{Q}+\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^{T}\right)\left(\Xi(t)\lambda^{*}(t)+\Psi(t)\right)\\-\left(\mathbf{A}(t)^{T}-\mathbf{S}\mathbf{R}^{-1}\mathbf{B}(t)^{T}\right)\lambda^{*}(t)\end{pmatrix}+\dot{\Psi}(t).\end{aligned}\tag{17}$$

Rewriting (17), results in,

$$\begin{aligned}&\begin{pmatrix}\dot{\Xi}(t)-\mathbf{A}(t)\Xi(t)-\Xi(t)\mathbf{A}(t)^{T}-\Xi(t)\mathbf{Q}\Xi(t)\\+\left(\Xi(t)\mathbf{S}+\mathbf{B}(t)\right)\mathbf{R}^{-1}\left(\mathbf{S}^{T}\Xi(t)+\mathbf{B}(t)^{T}\right)\end{pmatrix}\lambda^{*}(t)\\&+\begin{pmatrix}\dot{\Psi}(t)+\mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^{T}\Psi(t)\\-\Xi(t)\mathbf{Q}\Psi(t)+\Xi(t)\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^{T}\Psi(t)-\mathbf{A}(t)\Psi(t)\end{pmatrix}=0.\end{aligned}\tag{18}$$

The above equation is valid for any arbitrary value of optimal co-state $\lambda^{*}(t)$. This gives the definition of the inverse matrix differential Riccati equation $\Xi(t)$ as in,

$$\begin{aligned}\dot{\Xi}(t)&=\mathbf{A}(t)\Xi(t)+\Xi(t)\mathbf{A}(t)^{T}+\Xi(t)\mathbf{Q}\Xi(t)\\&-\left(\Xi(t)\mathbf{S}+\mathbf{B}(t)\right)\mathbf{R}^{-1}\left(\mathbf{S}^{T}\Xi(t)+\mathbf{B}(t)^{T}\right).\end{aligned}\tag{19}$$

Moreover, the vector differential equation in $\Psi(t)$ is obtained as:

$$\dot{\Psi}(t)=\left(\mathbf{A}(t)-\mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^{T}+\Xi(t)\mathbf{Q}-\Xi(t)\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^{T}\right)\Psi(t).\tag{20}$$

The set of equations (19) and (20) can be solved either using the initial or final boundary conditions.

At a given fixed final point, (15) can be changed to

$$\begin{aligned}t=t_0:&\quad X^{*}(t_0)=\Xi(t_0)\lambda^{*}(t_0)+\Psi(t_0),\\t=t_f:&\quad X^{*}(t_f)=\Xi(t_f)\lambda^{*}(t_f)+\Psi(t_f).\end{aligned}\tag{21}$$

Since the values of optimal co-states are arbitrary, the final boundary conditions can be obtained as,

$$\begin{aligned}t=t_0:&\quad \Xi(t_0)=0,\quad \Psi(t_0)=X(t_0),\\t=t_f:&\quad \Xi(t_f)=0,\quad \Psi(t_f)=X(t_f).\end{aligned}\tag{22}$$

Finally, using the transformation (15) and the state equation in (13), the optimal control laws and optimal states are obtained as:

$$\begin{aligned}U^{*}(t)=&-\mathbf{R}^{-1}\left(\mathbf{S}^{T}+\mathbf{B}(t)^{T}\Xi^{-1}(t)\right)X^{*}(t)\\&+\mathbf{R}^{-1}\mathbf{B}(t)^{T}\Xi^{-1}(t)\Psi(t),\end{aligned}\tag{23}$$

$$\begin{aligned}\dot{X}^{*}(t)=&\left(\mathbf{A}(t)-\mathbf{B}(t)\mathbf{R}^{-1}\left(\mathbf{S}^{T}+\mathbf{B}^{T}\Xi^{-1}(t)\right)\right)X^{*}(t)\\&+\mathbf{B}(t)\mathbf{R}^{-1}\mathbf{B}(t)^{T}\Xi^{-1}(t)\Psi(t).\end{aligned}\tag{24}$$

The set of optimal controls in (23), and optimal trajectory in (24) with general boundary conditions can be used to solve the path planning problems of linear systems defined by (7).

4. Optimal Robot-Environment Control

In this section, first, the iDRE method developed in Section 3 is applied to the environment model (4) to find the optimal trajectory and optimal interaction force of the system (3). Then, the position tracking controller is proposed and employing the Lyapunov direct method the stability analysis of the system is performed.

4.1. Optimal Control using iDRE Method

The aim of this section is to find the optimal interaction force $f_e(t)$, and the desired Cartesian position trajectory $x_d(t)$ within the environment model (4). To do this, we first reform model dynamics (4) to be in the form with the state equation in (7). Then, obtain optimal values by employing the presented iDRE method.

Choose the system states as $X_1(t)=x_d(t)$, and $X_2(t)=\dot{x}_d(t)$, and form the system state to be as

$$X(t)=\left[x_d(t)^{T},\dot{x}_d(t)^{T}\right]^{T}.\tag{25}$$

Now, considering the model dynamics (4), the environment dynamics can be described in the state-space form as

$$\dot{X}(t)=\mathbf{A}(t)X(t)+\mathbf{B}(t)U(t),\tag{26}$$

where $\mathbf{A}=\begin{bmatrix}\mathbf{0} & \mathbf{I_n}\\ \mathbf{\text{-}M_e}(t)^{\mathbf{-1}}\mathbf{C_e}(t) & \mathbf{\text{-}M_e}(t)^{\mathbf{-1}}\mathbf{G_e}(t)\end{bmatrix}$,

$\mathbf{B}=\left[0,\mathbf{\text{-}M_e}(t)^{\mathbf{-1}}\right]^{T}$, and $U(t)=f_e(t)$.

Now, as the environment dynamic (26) is in the same format with state equation (7), we can find optimal interaction force $f_e(t)$, and the optimal desired trajectory $x_d(t)$ following the presented iDRE method. To do this, the cost function is defined as a trade-off between the desired trajectory and the interaction force as,

$$\mathrm{E}=\frac{1}{2}\int X(t)^{T}\mathbf{Q}X(t)+U(t)^{T}\mathbf{R}U(t).\tag{27}$$

Note that in this paper the optimal REI is utilized to minimize the cost function (27) which is formed in terms of the desired trajectory and the interaction force. By that means a trade-off between optimal path properties [i.e. minimum trajectory or velocity], and interaction force optimization can be achieved. As an example of physical interpretation, this can be utilized in an assistive human-robot interaction [6] by minimizing the applied force, in applications like robotic rehabilitation.

It is also noted that in (27), we assumed the value of the performance parameter $\mathbf{S}$, as in (6), to be zero. Also, it is worth noting that by forming the environment model according to the state system (7), the complete dynamic model of the robot can be obtained as in (3) without linearization of the model.

4.2. Position Control Design and Stability Analysis

4.2.1. Position Control Design

As the desired task space trajectory $x_d(t)$ has been obtained through the optimal control in the previous section, the joint space trajectory $q_d(t)$ can be obtained using robot inverse kinematics. This section develops position control to make the robot actual joint position $q(t)$ track the desired position $q_d(t)$.

To do this, the sliding mode error can be defined as,

$$\sigma(t) = \dot{e}(t) + \mu e(t), \tag{28}$$

with $e(t)$ being the trajectory error, defined by $e(t) = q_d(t) - q(t)$, where μ is a positive definite constant. According to the definition of error $\sigma(t)$, if $\lim_{t\to\infty} \dot{e}(t)$ exists, and $\lim_{t\to\infty} e(t) = 0$, then $\lim_{t\to\infty} \sigma(t) = 0$. Thus, our control objective can be achieved by making,

$$\lim_{t\to\infty} e(t) = 0. \tag{29}$$

The input control is proposed as,

$$\begin{aligned}\tau(t) &= \mathbf{H}(q(t))\ddot{q}_\gamma(t) + \mathbf{C}(q(t),\dot{q}(t))\dot{q}_\gamma(t) + \mathbf{G}(q(t)) \\ &+ \mathbf{J}^T(q(t)) f_e(t) + \kappa_p \sigma(t) + \kappa_i \int_0^t \sigma(\omega) d\omega,\end{aligned} \tag{30}$$

where $\ddot{q}_\gamma(t) = \dot{\sigma}(t) + \ddot{q}(t)$, $\dot{q}_\gamma(t) = \sigma(t) + \dot{q}(t)$, and κ_p, κ_i are positive definite matrices.

4.2.2. Stability Analysis

In this section, the original position tracking control task in the previous section is completed by a stability analysis of the designed control. For this case, we employed the position control input given in (30) for the robot dynamics (3) as follows.

Theorem 1: Consider the robot dynamics (3). By considering Property 1, if the control strategy governed by (30) is applied, then the following results are guaranteed:

1) the error $e(t)$ asymptotically converges to zero, as $t \to \infty$.

2) all the signals in the closed-loop system are bounded.

Proof: Consider the following integration-type Lyapunov function candidate,

$$\begin{aligned}L(t) &= \frac{1}{2}\sigma^T(t)\mathbf{H}(q(t))\sigma(t) \\ &+ \frac{1}{2}\left(\int_0^t \sigma(\omega)d\omega\right)^T \kappa_i \int_0^t \sigma(\omega)d\omega.\end{aligned} \tag{31}$$

The derivative of $L(t)$ with respect to time can be given by

$$\dot{L}(t) = \sigma^T(t)\begin{pmatrix}\mathbf{H}(q(t))\dot{\sigma}(t) + \frac{1}{2}\dot{\mathbf{H}}(q(t))\sigma(t) \\ +\kappa_i \int_0^t \sigma(\omega)d\omega\end{pmatrix}. \tag{32}$$

Considering

$$\begin{aligned}&\mathbf{H}(q(t))\dot{\sigma}(t) = \\ &\quad -\mathbf{H}(q(t))\ddot{q}(t) + \mathbf{H}(q(t))(\ddot{q}_d(t) + \mu\dot{e}(t)),\end{aligned} \tag{33}$$

and

$$\begin{aligned}\mathbf{H}(q(t))\ddot{q}(t) &= -\mathbf{C}(q(t),\dot{q}(t))\dot{q}(t) - \mathbf{G}(q(t)) \\ &+ \tau(t) + \mathbf{J}^T(q(t)) f_e(t),\end{aligned} \tag{34}$$

and substituting control (30) into (32)- (34) with employing Property 1, gives,

$$\dot{L}(t) = -\sigma^T(t)\kappa_p\sigma^T(t) \le 0. \tag{35}$$

Integrate $\dot{L}(t)$, and considering κ_p is positive definite, then,

$$\begin{aligned}&\lambda_{\min}(\kappa_p)\int_0^t \sigma^T(\omega)\sigma^T(\omega)d\omega \\ &\le \int_0^t \sigma^T(\omega)\kappa_p\sigma^T(\omega)d\omega \le L(0),\end{aligned} \tag{36}$$

where $\lambda_{\min}(\kappa_p)$ is the minimum eigenvalue of κ_p.

Considering $L(0)$, and $\lambda_{\min}(\kappa_p)$ are positive, it follows that $\sigma(t) \in L_2^n$. Then, according to the definition of $\sigma(t)$ in (28) , and considering $\dot{q}_d$, and $\ddot{q}_d \in L_\infty^n$, we have $\dot{q}_\gamma \in L_\infty^n$, and $\ddot{q}_\gamma \in L_\infty^n$. From $\sigma(t) \in L_2^n$, and further $q_d \in L_\infty^n$, we can conclude that $\dot{\sigma}(t) \in L_\infty^n$.

On the other hand, considering $\dot{L}(t) = -\sigma^T(t)\kappa_p\sigma^T(t) \le 0$, then $0 \le L(t) \le L(0)$, $\forall t \ge 0$, leading to $L(t) \in L_\infty^n$, and according to (36), $\int_0^t \sigma^T(\omega)\kappa_p\sigma^T(\omega)d\omega$ is bounded since $L(0)$ is bounded. Finally, According to Barbalat's Lemma, $\sigma(t) \in L_2^n$, $\dot{\sigma}(t) \in L_\infty^n$ lead to $\sigma \to 0$ as $t \to \infty$, which completes the proof.

The overall optimal REI scenario presented in this paper is summarized in the Algorithm 1.

Algorithm 1: Presented Optimal Robot-Environment Interaction

Input: Environment model matrices $\mathbf{M_e}(t)$, $\mathbf{C_e}(t)$, and $\mathbf{G_e}(t)$, robot dynamic matrices $\mathbf{H}(q(t))$, $\mathbf{C}(q(t),\dot{q}(t))$, and $\mathbf{G}(q(t))$, the control constant μ, and control matrices $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{S}$, κ_p, and κ_i.

Initialization: Form environment model dynamic equations (26), compute the performance index (27).

Optimal control: Find matrix $\Xi(t)$ from (19), and the vector $\Psi(t)$ from (20). Then, find optimal controls in (23) and optimal states in (24).

Tracking Control: Consider control (23) as the interaction force $f_e(t)$, and states (24) as the desired task space trajectory $x_d(t)$. Find the joint space trajectory $q_d(t)$ using robot inverse dynamics. Compute the control (30), and find the joint trajectory $q(t)$ from robot dynamics (3).

5. Numerical Simulation

In this section, theoretical considerations are verified by numerical simulation. The study will highlight the effectiveness of the proposed method to handle optimal REI tracking control with the dynamic environments. A simple 2D manipulator in the vertical plane is used for simulation. To model the robot dynamics we let $\boldsymbol{q}(\boldsymbol{t})=[\boldsymbol{q}_1(\boldsymbol{t}),\boldsymbol{q}_2(\boldsymbol{t})]$, and used the Lagrange's equation method, then the dynamics of the robot can be expressed as [38],

$$\mathbf{H}(\boldsymbol{q}(\boldsymbol{t}))=\begin{bmatrix} \mathrm{p}_1+\mathrm{p}_2+2\,\mathrm{p}_3\cos\boldsymbol{q}_2(\boldsymbol{t}) & \mathrm{p}_2+\mathrm{p}_3\cos\boldsymbol{q}_2(\boldsymbol{t}) \\ \mathrm{p}_2+2\,\mathrm{p}_3\cos\boldsymbol{q}_2(\boldsymbol{t}) & \mathrm{p}_2 \end{bmatrix},$$

$$\mathbf{C}(\boldsymbol{q}(\boldsymbol{t}))=\begin{bmatrix} -\mathrm{p}_3\,\dot{\boldsymbol{q}}_2\sin\boldsymbol{q}_2(\boldsymbol{t}) & -\mathrm{p}_3(\dot{\boldsymbol{q}}_1+\dot{\boldsymbol{q}}_2)\sin\boldsymbol{q}_2(\boldsymbol{t}) \\ \mathrm{p}_3\,\dot{\boldsymbol{q}}_1\sin\boldsymbol{q}_2(\boldsymbol{t}) & 0 \end{bmatrix},$$

$$\mathbf{G}(\boldsymbol{q}(\boldsymbol{t}))=\begin{bmatrix} \mathrm{p}_4\,\boldsymbol{g}\cos\boldsymbol{q}_1+\mathrm{p}_5\,\boldsymbol{g}\cos(\boldsymbol{q}_1+\boldsymbol{q}_2) \\ \mathrm{p}_5\,\boldsymbol{g}\cos(\boldsymbol{q}_1+\boldsymbol{q}_2) \end{bmatrix},$$

where,

$\mathrm{p}_1=\mathrm{m}_1(\mathrm{L}_1/2)^2+\mathrm{m}_2(\mathrm{L}_2/2)^2+\mathrm{I}_1$, $\mathrm{p}_2=\mathrm{m}_2(\mathrm{L}_2/2)^2+\mathrm{I}_2$, $\mathrm{p}_3=\mathrm{m}_2\mathrm{L}_1(\mathrm{L}_1/2)^2$, $\mathrm{p}_4=\mathrm{m}_1(\mathrm{L}_2/2)^2+\mathrm{m}_2\mathrm{L}_1$, and $\mathrm{p}_5=\mathrm{m}_2(\mathrm{L}_2/2)^2$.

Physical parameters are chosen as the mass of links $\mathrm{m}_1=\mathrm{m}_2=5$ kg, length of links $\mathrm{L}_1=\mathrm{L}_2=1.5\,\mathrm{m}$, Inertia of links $\mathrm{I}_1=\mathrm{I}_2=0.125\ \mathrm{kgm}^2$. The gravitational acceleration is $\boldsymbol{g}=9.81\,\mathrm{m/s}^2$. It is supposed that the robot departs from $\boldsymbol{q_d}=[30,60]^{\boldsymbol{T}}$ degree, and the initial and final desired conditions in the Cartesian space are defined by $\boldsymbol{X}_0=[0.5,1.2]^{\boldsymbol{T}}$ m, and $\boldsymbol{X_f}=[-0.5,0.866]^{\boldsymbol{T}}$ m; all the velocity boundary conditions are assumed to be zero.

We also choose time-varying environment dynamic parameters as,

$$\mathbf{M_e}=\begin{bmatrix} \sin(t) & -5 \\ -5 & 0.3\sin(t) \end{bmatrix},\quad \mathbf{C_e}=\begin{bmatrix} 0.5\sin(t) & -15 \\ -5.5 & -1.5\sin(t) \end{bmatrix},\quad \mathbf{G_e}=\begin{bmatrix} 0.5\sin(t) & -3.5 \\ -3.5 & -0.5\sin(t) \end{bmatrix}. \tag{37}$$

The performance parameters in (27) are chosen as $\mathbf{Q}=\mathbf{I}$, and $\mathbf{R}=10\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. The control gains are defined as $\boldsymbol{\mu}=100$, $\boldsymbol{\kappa_p}=10$, and $\boldsymbol{\kappa_i}=0.2$. Simulation results are shown in Figs 1-4.

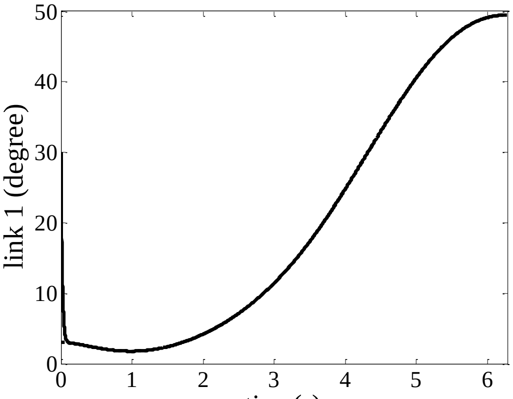

Figure 1 (a). Trajectory of joint positions: the desired signal (dotted line) versus the actual signal (solid line).

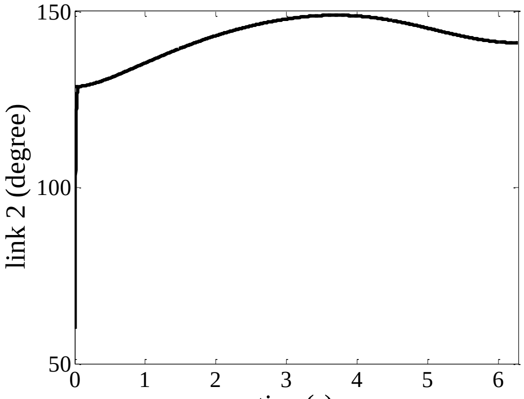

Figure 1 (b). Trajectory of joint positions: the desired signal (dotted line) versus the actual signal (solid line).

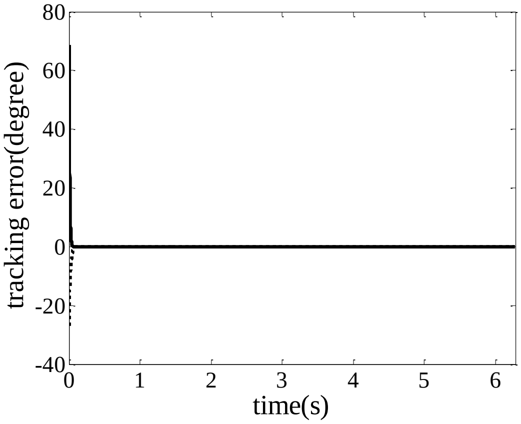

Figure 2. Tracking error of joint positions: joint 1 (solid line) versus joint 2 (dotted line).

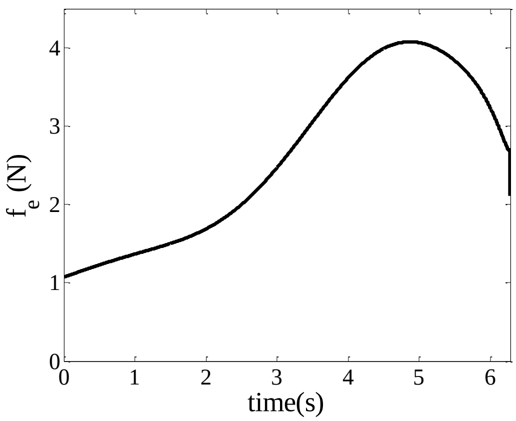

Figure 3. Required robot-environment interaction force.

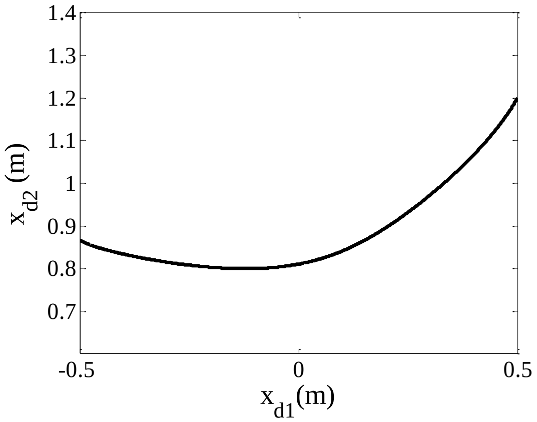

Figure 4. Trajectory of the end-effector in the Cartesian space.

The performance of the tracking controller is illustrated in Figs 1, 2. Figure 1 shows the desired (optimal) and actual values of joint positions. Tracking errors are shown in the Fig. 2. As shown in these figures, the position controller can track the obtained optimal positions asymptotically. Figure 3 shows the obtained optimal robot-environment force. Finally, the end-effector trajectory in the Cartesian space is depicted in Fig. 4. The results shown in the figures illustrate the ability of the presented iDRE method to plan the optimal path between two given endpoints. Also, the results illustrate that using the proposed REI method, an optimal interaction between the robot and environment can be achieved according to the environment characterizations, while stable tracking performance of the system can be accomplished.

6. Discussion

In this study, path planning is accomplished in the task-space and then the position tracking is handled. In addition, fixed and no-zero end-point states are considered for the planning problems. Also, to cope with an optimal REI problem, only environment properties are considered. By that means the optimal trajectory is achieved according to the task-specific information without needing the robot dynamics information. However, it is worth noting that the environment model parameters are assumed to be known. How to integrate the iDRE method with unknown environment models e.g. unknown $\mathbf{A}(t)$, and $\mathbf{B}(t)$ matrices within a unified framework requires further study. Also, in the presented method, the robot dynamics are supposed to be known. In this regard, the method may not be applicable to complex robots with challenging dynamics. Future research work will study techniques to cope with these issues. Finally, selecting a cost function is a nontrivial matter as different cost functions can change interaction performance [3]. A priori partial information from the environment can be helpful to tackle this issue in some cases, but solving this problem in a general case remains an open problem.

7. Conclusion

In this paper, optimal REI has been investigated using the iDRE method. An optimal closed-loop control has been developed for a linear system with two fixed endpoints over a specific time interval. The approach employs inverse Riccati transformation between state and co-state. Resulting equations have been subsequently used to find optimal trajectory and interaction force for robots interacting with the environment. The obtained optimal trajectory has been defined as the desired trajectory that integrated into the developing position controller. Trajectory following and stability of the closed-loop system have been analyzed using the Lyapunov direct method. Finally, numerical simulations have been performed to illustrate the effectiveness of the theoretical results.

Acknowledgment

Dr. Lei Cui is the recipient of a Discovery Early Career Award (project number DE170101062) funded by the Australian Government.

References

[1] M. Vukobratovic, *Dynamics and robust control of robot-environment interaction*. World Scientific, 2009.

[2] S. S. Ge, Y. Li, and C. Wang, "Impedance adaptation for optimal robot–environment interaction," *International Journal of Control*, vol. 87, no. 2, pp. 249-263, 2014.

[3] Y. Li and S. S. Ge, "Impedance learning for robots interacting with unknown environments," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1422-1432, 2014.

[4] A. Cherubinia, R. Passamaa, A. Crosniera, A. Lasnierb, and P. Fraisse, "Collaborative manufacturing with physical human–robot interaction," *Robotics and Computer-Integrated Manufacturing*, vol. 40, pp. 1-13, 2016.

[5] H. Modares, I. Ranatunga, F. L. Lewis, and D. O. Popa, "Optimized Assistive Human-Robot Interaction Using Reinforcement Learning," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 655 - 667, 2016.

[6] H. N. Rahimi, I. Howard, and L. Cui, "Neural Impedance Adaption for Assistive Human-Robot Interaction," *Neurocomputing*, vol. 290, pp. 50-59, 2018.

[7] Y. Li, G. Carboni, F. Gonzalez, D. Campolo, and E. Burdet, "Differential game theory for versatile physical human–robot interaction," *Nature Machine Intelligence*, vol. 1, pp. 36-43, 2019.

[8] B. Alqaudi, H. Modares, I. Ranatunga, S. M. Tousif, F. L. Lewis, and D. O. Popa, "Model reference adaptive impedance control for physical human-robot interaction," *Control Theory and Technology,* vol. 14, no. 1, pp. 68-82, 2016.

[9] M. Sharifi, S. Behzadipour, and G. Vossoughi, "Nonlinear model reference adaptive impedance control for human–robot interactions," *Control Engineering Practice,* vol. 32, pp. 9-27, 2014.

[10] C. Torres, J. de Jesús Rubio, C. F. Aguilar-Ibáñez, and J. H. Pérez-Cruz, "Stable optimal control applied to a cylindrical robotic arm," *Neural Computing and Applications,* vol. 24, no. 3-4, pp. 937-944, 2014.

[11] M. S. Miah and W. Gueaieb, "RFID-Based Mobile Robot Trajectory Tracking and Point Stabilization Through On-line Neighboring Optimal Control," *Journal of Intelligent & Robotic Systems,* pp. 1-23, 2014.

[12] O. Santos, H. Romero, S. Salazar, and R. Lozano, "Discrete optimal control for a quadrotor UAV: Experimental approach," presented at the International Conference on Unmanned Aircraft Systems, 2014.

[13] C. Wei, J. Guo, S.-Y. Park, J. Xu, and X. Ma, "IFF Optimal Control for Missile Formation Reconfiguration in Cooperative Engagement," *Journal of Aerospace Engineering,* 2013.

[14] J. M. Adánez, B. M. Al- Hadithi, and A. Jiménez, "Wind turbine multivariable optimal control based on incremental state model," *Asian Journal of Control,* vol. 20, no. 6, pp. 2075-2087, 2018.

[15] M. Li and Q. Li, "Admissible Consensus of Multi- Agent Singular Systems," *Asian Journal of Control,* 2013.

[16] Y. Yang, "Analytic LQR design for spacecraft control system based on quaternion model," *Journal of Aerospace Engineering,* vol. 25, no. 3, pp. 448-453, 2011.

[17] J. Nazarzadeh, M. Razzaghi, and K. Nikravesh, "Solution of the matrix Riccati equation for the linear quadratic control problems," *Mathematical and computer modelling,* vol. 27, no. 7, pp. 51-55, 1998.

[18] M. Razzaghi, "Solution of the matrix Riccati equation in optimal control," *Information Sciences,* vol. 16, no. 1, pp. 61-73, 1978.

[19] D. S. Naidu, *Optimal control systems*. CRC press, 2002.

[20] A. Ferrante and L. Ntogramatzidis, "The generalized continuous algebraic Riccati equation and impulse-free continuous-time LQ optimal control," *Automatica,* vol. 50, no. 4, pp. 1176-1180, 2014.

[21] A. Ferrante and L. Ntogramatzidis, "Continuous-time singular linear–quadratic control: Necessary and sufficient conditions for the existence of regular solutions," *Systems & Control Letters,* vol. 93, pp. 30-34, 2016.

[22] S. J. Tan, W. Y. Zhou, H. J. Peng, and Z. G. Wu, "A novel extended precise integration method based on Fourier series expansion for periodic Riccati differential equations," *Optimal Control Applications and Methods,* vol. 38, no. 6, pp. 896-907, 2017.

[23] H.-J. Peng, Z.-G. Wu, and W.-X. Zhong, "Fourier expansion based recursive algorithms for periodic Riccati and Lyapunov matrix differential equations," *Journal of computational and applied mathematics,* vol. 235, no. 12, pp. 3571-3588, 2011.

[24] M. Korayem and A. Nikoobin, "Maximum payload path planning for redundant manipulator using indirect solution of optimal control problem," *The International Journal of Advanced Manufacturing Technology,* vol. 44, no. 7-8, pp. 725-736, 2009.

[25] D. A. Anisi, J. Hamberg, and X. Hu, "Nearly time-optimal paths for a ground vehicle," *Journal of Control Theory and Applications,* vol. 1, no. 1, pp. 2-8, 2003.

[26] S. Effati and H. S. Nik, "Solving a class of linear and non-linear optimal control problems by homotopy perturbation method," *IMA journal of mathematical control and information,* vol. 28, no. 4, pp. 539-553, 2011.

[27] M. H. Korayem, H. N. Rahimi, and A. Nikoobin, "Mathematical modeling and trajectory planning of mobile manipulators with flexible links and joints," *Applied Mathematical Modelling,* vol. 36, no. 7, pp. 3229-3244, 2012.

[28] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot manipulator control: theory and practice*. CRC Press, 2003.

[29] H. N. Rahimi, I. Howard, and L. Cui, "Neural Network Adaptive Control Design for Robot Manipulators under Velocity Constraints," *Journal of the Franklin Institute,* vol. 355, no. 2, pp. 693-713, 2018.

[30] T. H. Lee and C. J. Harris, *Adaptive neural network control of robotic manipulators*. World Scientific, 1998.

[31] K. Dupree, C.-H. Liang, G. Hu, and W. E. Dixon, "Adaptive Lyapunov-based control of a robot and mass–spring system undergoing an impact collision," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* vol. 38, no. 4, pp. 1050-1061, 2008.

[32] Y. Li and S. S. Ge, "Human–robot collaboration based on motion intention estimation," *IEEE/ASME Transactions on Mechatronics,* vol. 19, no. 3, pp. 1007-1014, 2014.

[33] D. E. Kirk, *Optimal control theory: an introduction*. Courier Corporation, 2012.

[34] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control*. John Wiley & Sons, 2012.

[35] P. Bader, S. Blanes, and E. Ponsoda, "Structure preserving integrators for solving (non-) linear quadratic optimal control problems with applications to describe the flight of a quadrotor," *Journal of Computational and Applied Mathematics,* vol. 262, pp. 223-233, 2014.

[36] I. Mufti, C. Chow, and F. Stock, "Solution of ill-conditioned linear two-point boundary value problems by the Riccati transformation," *SIAM Review,* vol. 11, no. 4, pp. 616-619, 1969.

[37] W. T. Reid, *Riccati differential equations*. Elsevier, 1972.

[38] Z.-L. Tang, S. S. Ge, K. P. Tee, and W. He, "Adaptive neural control for an uncertain robotic manipulator with joint space constraints," *International Journal of Control,* vol. 89, no. 7, pp. 1428-1446, 2016.