Optimization in Dynamical Systems: Theory and Application

Masoud Roudneshin

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy (Electrical Engineering) at

Concordia University

Montréal, Québec, Canada

September 2022

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Masoud Roudneshin**

Entitled: **Optimization in Dynamical Systems:**

Theory and Application

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Electrical Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Wei-Ping Zhu

_____ External Examiner
Dr. Lacramioara Pavel

_____ Examiner
Dr. Luis Rodrigues

_____ Examiner
Dr. Walter Lucia

_____ Supervisor
Dr. Amir G. Aghdam

Approved by _____
Martin D. Pugh, Chair
Department of Electrical and Computer Engineering

_____ 2022

_____
Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

**Optimization in Dynamical Systems:
Theory and Application**

Masoud Roudneshin, Ph.D.

Concordia University, 2022

In this dissertation, we study optimization methods in interconnected systems and investigate their applications in robotics, energy harvesting, and mean-field linear quadratic multi-agent systems. We first focus on parallel robots. Parallel Robots have numerous applications in motion simulation systems and high-precision instruments. Specifically, we investigate the forward kinematics (FK) of parallel robots and formulate it as an error minimization problem. Following this formulation, we develop an optimization algorithm to solve FK and provide a theoretical analysis of the convergence of the proposed algorithm. Then, we investigate the energy optimization (maximization) in a specific class of micro-energy harvesters (MEH). These types of energy harvesters are known to extract the largest amount of power from the kinetic energy of the human body, making them an appropriate choice for wearable technology in healthcare applications. Employing machine learning tools and using the existing models for the MEH's kinematics, we propose three methods for energy maximization. Next, we study optimal control in a mean-field linear quadratic system. Mean-field systems have critical applications in approximating very large-scale systems' behavior. Specifically, we establish results on the convergence of policy gradient (PG) methods to the optimal solution in a mean-field linear quadratic game. We finally consider the risk-constrained control of agents in a mean-field linear quadratic setting. Simulations validate the theoretical findings and their effectiveness.

Acknowledgments

First and foremost, I must genuinely thank my supervisor, Prof. Amir G. Aghdam for being a wonderful mentor. I was fortunate to learn the great power of theory in different applications under his supervision. He guided me in a research path and Ph.D. experience that I am proud of. I am thankful for his great support and passion for exposing me to a variety of projects ranging from energy harvesting to robotics and theory of machine learning. I would also like to extend my sincere gratitude to Dr. Kamran Sayrafian from National Institute of Standards and Technology for his huge support and passion in the energy harvesting project. In addition, I am immensely grateful for working with Dr. Kamran Ghaffari from Touche Technologies to whom I am indebted for better critical thinking abilities and an amazing internship opportunity at Touche Technologies. I should also express my warmest regards to Dr. Jalal Arabneydi for his amazing mentorship and providing me the opportunity to have a better understanding about the theory of reinforcement learning and optimization. Next, I would like to express my warmest appreciation to my examining committee: Prof. Luis Rodrigues, Prof. Wei-Ping Zhu, Prof. Walter Lucia, and Prof. Lacramioara Pavel. I am truly indebted for their time and consideration to attend my dissertation defense, despite their busy schedule.

Lastly, I would like to thank my beloved family, my mother, Fatemeh, my father, Soleiman, and my amazing brother, Mostafa who all gave me their unconditional love and kindness.

To my beloved family

and to the the people of Iran fighting for freedom and a better future

Contribution of Authors

- Chapter 2: This Chapter was published in *IEEE Control Systems Letters* (2022) with Dr. Kamran Ghaffari and Dr. Amir G. Aghdam. Dr. Ghaffari initiated the idea and helped with the edition of the manuscript. Dr. Aghdam led the research, checked the theory, and edited the manuscript.

- Chapters 3: The result of this chapter was published in the *American Control Conference 2022* with Dr. Kamran Sayrafian and Dr. Amir G. Aghdam, who helped with writing the manuscript. Also, some parts of this chapter are under final revisions to be submitted to a journal by the same authors.

- Chapter 4: This chapter was published in the *IEEE Sensors 2022* with Dr. Kamran Sayrafian and Dr. Amir G. Aghdam who checked the theory and edited the manuscript.

- Chapter 5: This chapter was published in the *IEEE Conference on Decision and Control 2020* with Dr. Jalal Arabneydi and Dr. Amir G. Aghdam. Dr. Arabneydi and Dr. Aghdam checked the theory and edited the manuscript.

- Chapter 6: This chapter will be submitted as a conference paper, co-authored by Dr. Amir G. Aghdam who checked the theory and edited the manuscript.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this dissertation, we study some optimization problems in dynamical systems. The main theme in all the following chapters is the optimization (maximization or minimization) of a cost function.

In Part I (Chapter 2), we focus on the application of optimization in robotics. We study the forward kinematics (FK) of a parallel robot where the objective is to find the end-effector configuration given the joint space values (either joint length or joint angles). Unlike the inverse kinematics of parallel robots which usually find a closed-form solution, FK may be solved through solution of a series of nonlinear equations. In practice, FK is a trade-off between precision and computational effort. Existing approaches can be categorized as analytical and numerical methods. In analytical methods, the problem is reduced to solving a polynomial function involving multiple sines and cosines products and seeking closed-form solutions [1,3–6]. However, solving a high-order polynomial may be impractical and inefficient for real-time applications. On the other hand, the Newton-Raphson method is often employed in numerical approaches to solve the problem [7–19]. This class of algorithms involves computing the Jacobian matrix of the manipulator and its inverse at each iteration. However, such methods require a significant computational resource which may be infeasible for real-time applications. As an alternative approach, we formulate FK as an error minimization problem and establish analytical results on the convergence of the solution to the sub-optimal values for a special class of parallel robots.

In addition, our results indicate that the required computational effort of our proposed method is even lower in comparison to the state-of-the-art methods.

In Part II (Chapters 3-5), we investigate energy harvesting (EH) in a micro energy harvester known as Coulomb force parametric generator (CFPG) [33, 34]. CFPGs can be attached to some parts of the human body to harvest energy from the kinetic motions during daily life activities. To maximize the harvested energy, we are interested to adjust a parameter (electrostatic force F) inside CFPG. Adjustment of the electrostatic force is dependent on the intensity of the external acceleration of the CFPG's frame. Therefore, this problem may be categorized as an online optimization problem [23, 24]. Despite the rich literature in online optimization, the existing methods are mostly tailored for problems enjoying special structures (convex loss functions) and relatively smooth dynamics. However, a CFPG architecture has a nonlinear dynamics and non-convex cost function, which makes the application of the existing theories almost impossible. In addition, one factor in choosing algorithms in EH is their relatively low amounts of computational complexity. In Chapters 3-5, we study different methods using tools form optimization and machine learning to find a sub-optimal solution for the EH problem in CFPG micro energy harvesters.

In part III (Chapter 6), we study optimization in nonzero-sum deep structured linear quadratic systems. We consider players having linear dynamics interacting with each other through a set of weighted averages of all players. Specifically, we are interested in finding the optimal policy of the players minimizing the cumulative infinite horizon cost of each player using model-based and model-free policy gradient (PG) methods. This problem finds significance in the theory of machine learning in explaining the learning process of an agent interacting with a dynamical system. Speaking of sample-based (model-free) methods, we are interested to find bounds on the number of required samples to attain a guaranteed level of performance. Although the work in [51] proves the non-convexity of the optimization in policy space and the fact that policy optimization does not generally converge in game setting [53], we prove that the proposed model-based and model-free policy gradient descent and natural policy gradient descent algorithms globally converge

to the sub-game perfect Nash equilibrium.

There are three main features that unify the problems studied in this dissertation:

(1) We consider dynamical systems whose state can be modeled as a differential or algebraic equation.

(2) The objective of all the problems is to minimize (or maximize) a non-negative cost function as a performance measure for these systems.

(3) We use analogous mathematical tools from optimization and control theory for the analysis of these systems.

Finally, some real-world applications were the main thrust of studying the three problems in this dissertation.

- Problem 1 was motivated by a project carried out at Touche Technologies to address some of the existing challenges in robotic motion simulation systems.
- Problem 2 was part of a project supported by the National Institute of Standards and Technology (NIST), which aimed at improving the energy efficiency of wearable medical devices.
- Problem 3 is studied to better understand the behavior of learning algorithms in multi-agent systems. This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Part I

Robotics

Chapter 2

On Forward Kinematics of a 3SPR Parallel Manipulator

This chapter presents a new numerical method to solve the forward kinematics (FK) of a parallel manipulator with a three-limb spherical-prismatic-revolute (3SPR) structure. Unlike the existing numerical approaches that require the manipulator's Jacobian matrix and its inverse at each iteration, the proposed algorithm requires much less complex computations to estimate the FK parameters. A cost function is introduced that measures the difference between actual FK values and its estimates. At each iteration, the problem is handled in two steps. First, the estimates of the platform orientation from the heave estimates are obtained. Then, the heave estimates are updated along the gradient direction of the proposed cost function. To evaluate the performance of the proposed algorithm, we compare results with those of a Jacobian-based (JB) approach for a 3SPR parallel manipulator.

2.1 Introduction

The forward kinematics (FK) problem has attracted researchers in various engineering fields and has fundamental applications in robotics. The problem is in finding the manipulator's workspace configuration with a given set of joint lengths or angles. The requirement to solve, in real-time, a set of nonlinear equations that contain products of trigonometric

functions makes the FK problem a challenging one for parallel manipulators [1–3, 6–8].

There is a vast body of recent literature on methods to solve FK for parallel manipulators [1–19]. These approaches can be categorized as analytical and numerical methods. In analytical methods, the problem is reduced to solving a polynomial function involving multiple sine and cosine products and seeking closed-form solutions [1, 3–6]. In most cases, a post-processing step is required to identify the feasible answer among various solutions of the polynomial. However, solving a high-order polynomial may be impractical and inefficient for real-time applications.

On the other hand, the Newton-Raphson method is often employed in numerical approaches to solve the problem [7–19]. This class of algorithms involves computing the Jacobian matrix of the manipulator and its inverse at each iteration. However, such methods require a significant computational resource which may be infeasible for real-time applications. Furthermore, the algorithm is sensitive to the initial value that may harm the convergence to the actual solution.

For a 3SPR architecture, passive degrees-of-freedom (DoFs) are a function of active DoFs and possess very small amplitudes compared to the dimensions of the manipulator. The corresponding equations include complex algebraic relations that need to be solved, adding to the problem's computational complexity. The present chapter focuses on a 3SPR parallel manipulator with a three DoF architecture to develop a new numerical approach for solving the FK problem. The algorithm is computationally efficient as it neglects the passive DoFs at the cost of some estimation error. In the proposed method, instead of formulating the problem as a non-convex problem, FK is solved in three steps. First, it is shown that the additional error introduced by neglecting the passive DoFs is upper bounded by the maximum amplitude of the passive DoFs. Then, it is demonstrated that the orientation of the manipulator can be estimated using the translational movements of the manipulator along the z-axis, called heave. It is also shown that estimation errors in the orientation are functions of the heave estimation error and the maximum amplitude of the passive DoFs. Next, using some results from the inverse kinematics (IK) problem, which is much easier to solve, a cost function is proposed that indirectly measures the distance

between the actual and estimated FK values. Finally, an algorithm is developed to solve for FK, and its convergence to the actual value of heave is analytically investigated.

The rest of the chapter is organized as follows. In Section II, the kinematics of a 3SPR parallel manipulator is reviewed. The proposed methodology and main results are presented in Section III. In Section IV, the theoretical results are validated by simulation. The chapter is concluded in Section V.

2.2 Kinematics Review of 3SPR Manipulator

In this section, the architecture of the parallel manipulator is elaborated. Then, inverse and forward kinematics problems are discussed.

2.2.1 Manipulator Architecture

Throughout the chapter, $\mathbb{R}$, $\mathbb{R}^+$ and $\mathbb{N}$ refer to the sets of real, positive real, and natural numbers, respectively. Given any $n \in \mathbb{N}$, $\mathbb{N}_n$ denotes the finite set $\{1, \ldots, n\}$. For any vector $v \in \mathbb{R}^{3\times 1}$, the expression of the vector in frame $\{F\}$ is denoted by ${}^F v$. Let also ${}^F v^x$, ${}^F v^y$, and ${}^F v^z$ denote the elements of v in x, y and z directions of frame $\{F\}$. Fig. 2.1 depicts the generic architecture of a 3SPR parallel manipulator. As seen in the figure, the manipulator's base is a triangle with vertices A_1, A_2 and A_3. An inertial frame $\{I\}$ is attached to the manipulator's base. The moving platform is defined by a triangle $B_1B_2B_3$, which is the same size as triangle $A_1A_2A_3$. A moving coordinate frame $\{M\}$ is also attached to the moving platform.

The three vertices A_1, A_2 and A_3 are fixed at the point of attachment of spherical joints to the ground. Similarly, vertices B_1, B_2 and B_3 are the revolute joints attached to the moving platform. The origin O' of the moving platform is obtained by finding the intersection of the orthogonal lines to the axes of the revolute joints. The inertial frame origin O is obtained analogously with respect to the triangle $A_1A_2A_3$. Each spherical joint on the inertial frame is attached to the revolute joints on the moving frame by a prismatic joint. Let ${}^I a_i \in \mathbb{R}^{3\times 1}$ and ${}^I b_i \in \mathbb{R}^{3\times 1}$ denote the position vector of A_i and B_i, $i \in \mathbb{N}_3$, with respect to the inertial

frame, respectively. Also, let $a_1 = [d_1, 0, 0]^\intercal$, $a_2 = [-d_2, d_3, 0]^\intercal$ and $a_3 = [-d_2, -d_3, 0]^\intercal$, where $d_1, d_2, d_3 \in \mathbb{R}^+$. Denote l_1, l_2 and l_3 as the length of each prismatic actuator and $l = [l_1, l_2, l_3]^\intercal \in \mathbb{R}^{3\times 1}$ as the vector of prismatic actuators' lengths.

The parallel 3SPR architecture provides three degrees of freedom (DoF) for the manipulator. Denote Z, α and β as the translational motion along the z-axis (heave), rotational motion around the x-axis (roll), and the rotational motion around the y-axis (pitch), respectively. Also, denote X, Y and γ as the small translation of the moving platform along the x and y axes and the small rotation around the z-axis (yaw), respectively. The set of small motions (X, Y, γ) is generally known as parasitic motions [7].

2.2.2 Inverse Kinematics

Denote $R = R_z(\gamma)R_y(\beta)R_x(\alpha)$ as the X-Y-Z rotation matrix of the moving platform with respect to the fixed frame, and $P = [X, Y, Z]^\intercal$ as the coordinate of the moving platform with respect to the fixed frame. The position of each revolute joint in $\{I\}$ can be expressed as

$$^I b_i = P + R \times {}^M b_i \qquad i \in \mathbb{N}_3 \tag{1}$$

or equivalently

$$^I b_i = {}^I a_i + l_i \hat{s}_i \qquad i \in \mathbb{N}_3 \tag{2}$$

where $\hat{s}_i$ denotes the unit vector of the ith prismatic joint expressed in $\{I\}$. It results from (1) and (2) that

$$l_i = |P + R \times {}^M b_i - {}^I a_i| \qquad i \in \mathbb{N}_3 \tag{3}$$

In other words, if the space of prismatic values is denoted by $\theta = (l_1, l_2, l_3)$ and the feasible workspace values by $\Theta = (Z, \alpha, \beta)$, the inverse kinematics equation (3) defines a mapping Φ from the workspace to joint space represented by

$$\Phi : \mathbb{R}^3 \to \mathbb{R}^3, \qquad \theta = \Phi(\Theta) \tag{4}$$

Remark 1. *In a 3SPR manipulator, to solve the IK problem for exact joint lengths, the parasitic*

motions must be computed as a function of the configuration Θ.

2.2.3 General Form of Parasitic Motions

Let T denote the transformation matrix from the inertial frame to the moving frame such that $T = \left[\begin{array}{c|c} R & P \\ \hline [0,0,0] & 1 \end{array}\right]$. Recall that the coordinates of the spherical joints with respect to $\{M\}$ can be expressed by using T^{-1} that leads to

$$^{M}a_i = R^{\intercal} \times (-P + {}^{I}a_i) \tag{5}$$

Since the revolute joints constrain the relative motion of spherical joints with respect to $\{M\}$, it always holds that

$$^{M}a_1^y = 0, \quad {}^{M}a_2^y = m(^{M}a_2^x), \quad {}^{M}a_3^y = -m(^{M}a_3^x) \tag{6}$$

where $m = \frac{^{M}b_2^y}{^{M}b_2^x} = -\frac{^{M}b_3^y}{^{M}b_3^x}$. The closed-form solutions for these motions can be obtained from (5) and (6). For the yaw angle $\gamma = \frac{(d_1 d_2 + d_2^2)\sin\alpha \sin\beta}{(d_1 d_2 + d_2^2)\cos\alpha + d_3^2 \cos\beta}$. The expression for the X and Y are page-long algebraic equations which are provided in Appendix.

Remark 2. *One must account for the required extra computations imposed by considering parasitic motions to obtain exact solutions for both the inverse and forward kinematics.*

Consider $d_1 = 1150$ mm, $d_2 = 500$ mm and $d_3 = 390$ mm that exemplify typical dimensions of a 3SPR parallel manipulator as a motion platform. Fig. 2.2 illustrates the distribution of the ratios of $\frac{X}{Z}$ and $\frac{Y}{Z}$ for this manipulator. It is observed that these amplitude ratios are relatively small in comparison to the heave for different configurations. Hence, one way to solve the FK may be to consider a simplified form of the kinematic equation at the cost of some error. Therefore, the following problem is defined.

Problem 1 (Forward kinematics of a 3SPR manipulator)**.** *For a given length of prismatic actuators* l_1, l_2 *and* l_3*, develop an algorithm to estimate the manipulator workspace configuration* (Z, α, β).

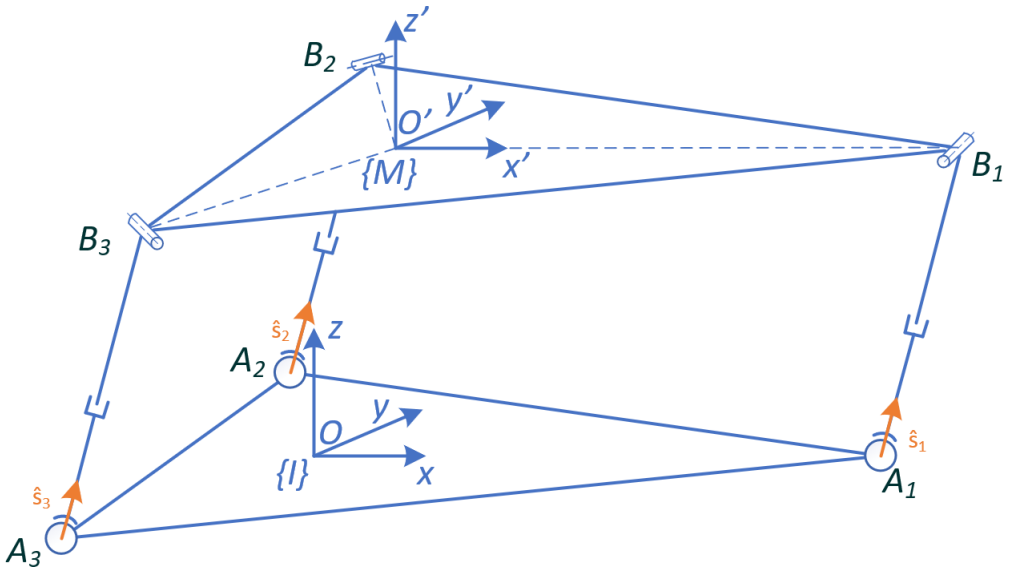

Figure 2.1: Architecture of a 3SPR manipulator

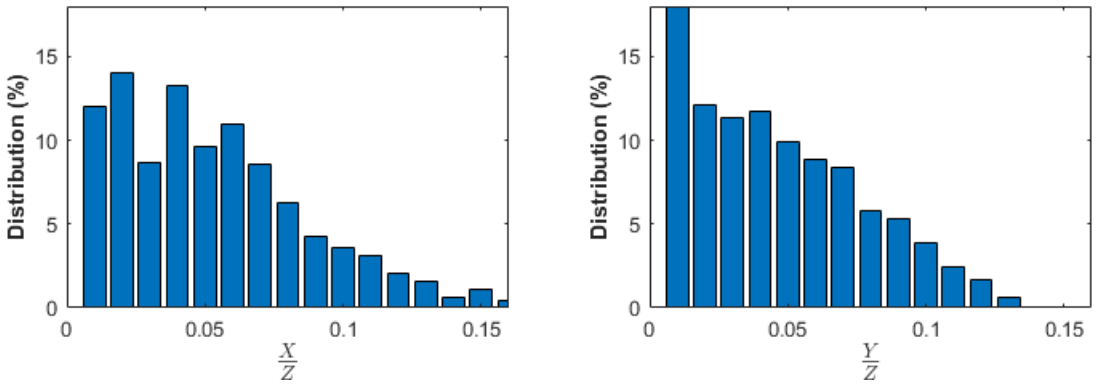

Figure 2.2: Distribution of the parasitic motion to heave ratios for a typical motion platform

2.3 Main Results

In this section, we tackle Problem 1 using a novel technique.

2.3.1 A Simplified IK Formulation

Consider the general form of the IK equation in (3) again. By neglecting the X and Y components of the motion, a simplified form of the IK can be introduced as $\tilde{\theta} = \widetilde{\Phi}(\Theta)$ which has the extended form $\tilde{l}_i = |\tilde{P} + R \times {}^{M}b_i - {}^{I}a_i|, i \in \mathbb{N}_3$

where $\tilde{l}_i$ denotes an approximate value of prismatic joint length for the ith limb, and $\tilde{P} = [0, 0, Z]^\intercal$ is the simplified position of the manipulator. The following lemma establishes an upper bound on the error between the actual value l_i and the approximate value of the joint length $\tilde{l}_i$.

Lemma 1. *Let* $\mu = \max\{|X|, |Y|\}$*; then* $|l_i - \tilde{l}_i| \leq \sqrt{2}\mu$.

Proof. By expanding (3) and using the triangle inequality, one has

$$|l_i - \tilde{l}_i| \leq |P - \tilde{P}| = \sqrt{X^2 + Y^2} \leq \sqrt{2\mu^2} = \sqrt{2}\mu. \tag{7}$$

□

2.3.2 Roll & Pitch Estimation from the Heave Estimates

In this step, the objective is to estimate the roll and the pitch angles from the geometry of the manipulator based on a given heave value. From (3), for $i = 1$, it holds that

$$l_1^2 = (X - d_1 + d_1 \cos(\beta))^2 + (Z - d_1 \sin(\beta))^2 + Y^2,$$

and with some simplifications, one has

$$(2d_1X + 2d_1^2)\cos(\beta) + 2d_1Z\sin(\beta) = 2d_1^2 + P^\intercal P - l_1^2.$$

Solving for β yields $\beta = \lambda \pm \omega$, where

$$\begin{aligned}\lambda &= \arcsin\Big(\frac{2d_1 Z}{\sqrt{(2d_1 X + 2d_1^2)^2 + (2d_1 Z)^2}}\Big),\\ \omega &= \arccos\Big(\frac{2d_1^2 + P^\intercal P - l_1^2}{\sqrt{(2d_1 X + 2d_1^2)^2 + (2d_1 Z)^2}}\Big).\end{aligned} \tag{8}$$

By considering the zero pitch condition ($\beta = 0$) for a zero heave ($Z = 0$), the solution with the plus sign is ruled out and therefore $\beta = \lambda - \omega$. Neglecting the effects of movement in the X and Y directions, the pitch angle can be estimated as

$$\hat{\beta} = \hat{\lambda} - \hat{\omega} \tag{9}$$

in which $\hat{\lambda} = \arcsin\Big(\frac{2d_1\hat{Z}}{\sqrt{(2d_1^2)^2 + (2d_1\hat{Z})^2}}\Big)$ and $\hat{\omega} = \arccos\Big(\frac{2d_1^2 + (\hat{Z}^2) - l_1^2}{\sqrt{(2d_1^2)^2 + (2d_1\hat{Z})^2}}\Big)$ where $\hat{Z}$ denotes the estimated heave.

To solve for the roll angle, consider the IK equation for the two rear limbs and subtract (3) with $i = 3$ from (3) with $i = 2$ to obtain

$$\begin{aligned}l_2^2 - l_3^2 = -4d_3\Big(&Y\cos(\alpha) - Y + Z\cos(\beta)\sin(\alpha)\\ &+ X\sin(\alpha)\sin(\beta) + d_2\sin(\alpha)\sin(\beta)\Big),\end{aligned}$$

which can be rewritten as

$$Y\cos(\alpha) + \Big(Z\cos(\beta) + (X + d_2)\sin(\beta)\Big)\sin(\alpha) = \frac{l_3^2 - l_2^2}{4d_3} + Y.$$

As a result $\alpha = \gamma \pm \kappa$, where

$$\begin{aligned}\gamma &= \arcsin\left(\frac{\Big(Z\cos(\beta) + (X + d_2)\sin(\beta)\Big)}{\sqrt{Y^2 + \Big(Z\cos(\hat{\beta}) + (X + d_2)\sin(\beta)\Big)^2}}\right)\\ \kappa &= \arccos\left(\frac{\frac{l_3^2 - l_2^2}{4d_3} + Y}{\sqrt{Y^2 + \Big(Z\cos(\beta) + (X + d_2)\sin(\beta)\Big)^2}}\right).\end{aligned} \tag{10}$$

Given the zero roll condition ($\alpha = 0$) for a zero heave ($Z = 0$), the solution with the plus sign is ruled out and $\alpha = \gamma - \kappa$. Let the translational displacement along the X and Y axes be neglected. Then, the roll angle can also be estimated as

$$\hat{\alpha} = \hat{\gamma} - \hat{\kappa} = \frac{\pi}{2} - \hat{\kappa} \tag{11}$$

where $\hat{\kappa} = \arccos\left(\frac{\frac{l_3^2 - l_2^2}{4d_3}}{\sqrt{\left(\hat{Z}\cos(\hat{\beta}) + (d_2)\sin(\hat{\beta})\right)^2}}\right)$.

Definition 1. *Define $\hat{R}$ as the estimated rotation matrix from (9) and (11). Let $e_z = Z - \hat{Z}$ denote the heave estimation error. Let also Ψ_1 and Ψ_2 represent the pitch and roll angles as functions of the heave and translational motion along the X and Y axes. Furthermore, define $\rho = \max\|P - \tilde{P}\|$ and the region $\mathcal{D} = \{P | \|P - \tilde{P}\| \leq \rho\}$.*

Lemma 2. *Let $P \in \mathcal{D}$; it holds that*

$$\begin{aligned} |\alpha - \hat{\alpha}| &\leq \mathcal{L}_1(\sqrt{2}\mu + e_z), \\ |\beta - \hat{\beta}| &\leq \mathcal{L}_2(\sqrt{2}\mu + e_z), \end{aligned} \tag{12}$$

where $\mathcal{L}_1 = \max|\nabla\Psi_1|$ and $\mathcal{L}_2 = \max|\nabla\Psi_2|$.

Proof. Since $\Psi_1(X, Y, Z)$ is continuously differentiable in the bounded region $\mathcal{D}$, it has a bounded Lipschitz constant within that region, denoted by $\mathcal{L}_1$, which yields

$$\begin{aligned} |\alpha - \hat{\alpha}| &\leq \mathcal{L}_1 |(X, Y, Z) - (0, 0, \hat{Z})| \\ &= \mathcal{L}_1 \sqrt{X^2 + Y^2 + (Z - \hat{Z})^2} \leq \mathcal{L}_1(\sqrt{2}\mu + e_z) \end{aligned} \tag{13}$$

where the above relation results from the triangle inequality. Using the same line of argument, an upper bound for the roll estimate error can also be established. □

2.3.3 Optimal Estimation of Heave

Consider the IK mapping in (4). Let Θ and $\hat{\Theta}$ denote the actual and estimated workspace configurations, respectively. It follows that if $\hat{\Theta} = \Theta$, then $\hat{\theta} = \theta$. One way to formulate this

observation is the following distance function

$$\Lambda_1(\hat{\Theta}) = (\theta - \Phi(\hat{\Theta}))^\intercal(\theta - \Phi(\hat{\Theta})),$$

where $\Lambda_1(\hat{\Theta}) = 0$ if and only if $\hat{\Theta} = \Theta$. Considering the computational effort for obtaining Φ, if one employs $\widetilde{\Phi}$ from $\widetilde{\Phi}$ instead, the following cost function is constructed

$$\Lambda_2(\hat{\Theta}) = (\theta - \widetilde{\Phi}(\hat{\Theta}))^\intercal(\theta - \widetilde{\Phi}(\hat{\Theta})). \tag{14}$$

Remark 3. *Assume that for a given set of joint lengths l, the roll and pitch angles are estimated by employing (9) and (11). Then, Λ_2 can be parametrized as a single variable function of heave, denoted as $\Lambda_3(Z)$. A sub-optimal value of the heave can be obtained along the gradient of this function with respect to Z.*

Remark 4. *By estimating the values of the roll and pitch angles, an approximate direction of the gradient can be used to update the heave estimates as*

$$\hat{Z}_{k+1} = \hat{Z}_k - \eta\widehat{\left(\frac{\partial \Lambda}{\partial Z}\right)}, \tag{15}$$

where η is a proper update step size defined later.

Considering Remarks 3 and 4, Algorithm 1 is proposed to solve for the FK problem.

Lemma 3. *It always holds that*

$$\left\|\frac{\partial \tilde{l}_i}{\partial Z}\right\| \leq 1, \quad i \in \mathbb{N}_3. \tag{16}$$

Proof. It follows from (3) that

$$\tilde{l}_i = \sqrt{(\tilde{P} + (R - I)a_i)^\intercal(\tilde{P} + (R - I)a_i)} \tag{17}$$

and it holds that $\frac{\partial \tilde{l}_i}{\partial Z} = \frac{[0 \quad 0 \quad 1](\tilde{P}+(R-I)a_i)}{\sqrt{(\tilde{P}+(R-I)a_i)^\intercal(\tilde{P}+(R-I)a_i)}}$. The proof follows on noting that for any vector v, $\left\|\frac{[0 \quad 0 \quad 1]v}{\sqrt{v^\intercal v}}\right\| \leq 1$. □

Lemma 4. *It holds that* $\left|\frac{\partial \Lambda_3}{\partial Z}\right| \leq \|\tilde{l} - l\|_1$.

Algorithm 1 Forward kinematics estimation

Input: $\theta = (l_1, l_2, l_3)$, step size η and N iterations
Output: Forward kinematics estimate $\hat{\Theta} = (\hat{Z}, \hat{\alpha}, \hat{\beta})$

Initialize iteration counter $k \leftarrow 0$
Initialize $\hat{Z}_k \leftarrow \hat{Z}_0$
Calculate $\hat{\beta}_k$ from (9)
Calculate $\hat{\alpha}_k$ from (11)
Calculate $\hat{Z}_{k+1}$ from (15)
while $k < N$ **do**
    $\hat{Z}_{\mathrm{k}} \leftarrow \hat{Z}_{k+1}$
    Calculate $\hat{\beta}_k$ from (9)
    Calculate $\hat{\alpha}_k$ from (11)
    Calculate $\hat{Z}_{k+1}$ from (15)
    $k \leftarrow k + 1$
end while

Proof. From (14) and Remark 3 the cost function is described as $\Lambda_3(Z) = \frac{1}{2}\sum_{i=1}^{3}(l_i - \tilde{l}_i)^2$. Therefore

$$\frac{\partial \Lambda_3}{\partial Z} = \sum_{i=1}^{3}(l_i - \tilde{l}_i)\frac{\partial \tilde{l}_i}{\partial Z}. \tag{18}$$

The proof follows from Lemma 3 and the definition of L_1 norm. □

Remark 5. *Recall that the discrepancy between $\tilde{l}_i$ and l_i, $i \in \mathbb{N}_3$, is due to the parasitic motions X and Y. In addition, parasitic motions affect the discrepancy between the estimated and exact rotation matrices $\hat{R} - R$. Denote by ϵ_{1i}, ϵ_{2i}, ϵ_{3i} and ϵ_{4i} the scaled compound effect of these errors as follows*

$$\begin{aligned}
\epsilon_{1i} &= \frac{[0 \;\; 0 \;\; 1]\Big([X \;\; Y \;\; 0] + (\hat{R} - R)a_i\Big)}{\tilde{l}_i + l_i}, \\
\epsilon_{2i} &= \frac{a_i^{\intercal}(\hat{R} - I)^{\intercal}(\hat{R} - I)a_i - a_i^{\intercal}(R - I)^{\intercal}(R - I)a_i}{\tilde{l}_i + l_i}, \\
\epsilon_{3i} &= \frac{2[0 \;\; 0 \;\; Z](\hat{R} - R)a_i}{\tilde{l}_i + l_i}, \\
\epsilon_{4i} &= \frac{X^2 + Y^2 + 2[X \;\; Y \;\; 0](\hat{R} - R)a_i}{\tilde{l}_i + l_i}.
\end{aligned}$$

Next, the general form of the partial derivative of the cost function with respect to the optimal heave is presented.

Lemma 5. *The partial derivative of the cost function with respect to the heave can be expressed as*

$$\frac{\partial \Lambda_3}{\partial Z} = \sum_{i=1}^{3}((1+\epsilon_{1i})(\hat{Z}-Z)+\epsilon_{2i}+\epsilon_{3i}+\epsilon_{4i})\frac{\partial \tilde{l}_i}{\partial Z}. \tag{19}$$

Proof. From (18), one has $\frac{\partial \Lambda_3}{\partial Z} = \sum_{i=1}^{3}(\tilde{l}_i^2 - {l_i}^2)\frac{1}{l_i+\tilde{l}_i}\frac{\partial \tilde{l}_i}{\partial Z}$. The proof follows now from (3) and (17). □

Definition 2. *For a given step size η, define*

$$\delta = (1-\eta\sum_{i=1}^{3}(1+\epsilon_{1i})\frac{\partial \tilde{l}_i}{\partial Z}).$$

Also, define

$$c_1 = \sum_{i=1}^{3}|1+\epsilon_{1i}| \quad c_2 = \sum_{i=1}^{3}|\epsilon_{2i}|+|\epsilon_{3i}|+|\epsilon_{4i}|.$$

Theorem 1. *Let Z and $\hat{Z}_0$ denote the actual heave and its initial estimate, respectively. Using Algorithm 1 with the step size $\eta < \frac{1}{c_1}$, the heave estimate error after N iterations is upper-bounded as described below*

$$\left|\hat{Z}_N - Z\right| \leq \max\left\{\left|\frac{c_2}{c_1}\right|\right\} + \delta^{N-1}\left|\hat{Z}_0 - Z\right|.$$

Proof. From (15) and (19), it holds that

$$\begin{aligned}\hat{Z}_N - Z &= (\hat{Z}_{N-1} - Z) \\ &\quad - \eta\sum_{i=1}^{3}((1+\epsilon_{1i})(\hat{Z}_{N-1}-Z)+\epsilon_{2i}+\epsilon_{3i}+\epsilon_{4i})\frac{\partial \tilde{l}_i}{\partial Z} \\ &\leq \left|\delta(\hat{Z}_{N-1}-Z)\right| + \eta c_2,\end{aligned} \tag{20}$$

where the last inequality results from (16) and the triangle inequality. If $\eta < \frac{1}{c_1}$, then $\delta < 1$.

Let (20) be expanded as

$$\begin{aligned}
\left|\hat{Z}_N - Z\right| &\leq \delta\left|\hat{Z}_{N-1} - Z\right| + \left|\eta c_2\right| \leq \delta^2\left|\hat{Z}_{N-2} - Z\right| + (1+\delta)\left|\eta c_2\right| \\
&\vdots \\
&\leq \delta^{N-1}\left|\hat{Z}_0 - Z\right| + \sum_{i=0}^{\infty}\delta^i\left|\eta c_2\right| = \delta^{N-1}\left|\hat{Z}_0 - Z\right| + \frac{\left|\eta c_2\right|}{1-\delta} \\
&= \delta^{N-1}\left|\hat{Z}_0 - Z\right| + \left|\frac{c_2}{c_1}\right|.
\end{aligned} \tag{21}$$

This completes the proof. □

Remark 6. *If the manipulator's motion is sufficiently smooth such that* $|\epsilon_{1i}| \ll 1$, $|\epsilon_{2i}| \ll 1$, $|\epsilon_{3i}| \ll 1$ *and* $|\epsilon_{4i}| \ll 1$, *then Theorem 1 implies that the heave can be estimated sufficiently accurately with the maximum residual error of* $\max\left\{\left|\frac{c_2}{c_1}\right|\right\}$.

Remark 7. *The proposed method may be generalized to other parallel manipulators if two necessary conditions are met: (i) The ratio of the parasitic motion to the independent DoFs is small enough to simplify the kinematic equations with negligible error, and (ii) for a given set of joint lengths (angles), all independent DoFs can be formulated as a function of only one DoF. Our preliminary findings indicate that the proposed method can be extended to other commercial parallel manipulators like H4 [21] and Stewart-Gough platform [22].*

2.3.4 Computational Complexity of the Forward Kinematics

In this part, we compare the computational complexity of the JB method with that of the proposed method. For the IK mapping in (4), a first-order approximation at the nth time instant may be written as

$$\theta_n = \Phi(\Theta_n) = \Phi(\Theta_{n-1} + \Delta\Theta) \approx \Phi(\Theta_{n-1}) + \frac{\partial\Phi}{\partial\Theta}\Delta\Theta,$$

where $J = \frac{\partial\Phi}{\partial\Theta}$ is known as the Jacobian in IK problem. Also, recall that

$$\Phi(\Theta_{n-1}) = \theta_{n-1},$$

therefore

$$\theta_n - \theta_{n-1} = \Delta\theta \approx J\Delta\Theta. \tag{22}$$

Hence, to solve FK, starting from an initial configuration, the workspace variables can be approximated by first computing the elements of the Jacobian J and then solving (22) for $\Delta\Theta$.

For solving FK with the proposed method, the estimates of the pitch (9) and the roll angle (11) should be computed at every iteration. Then, the heave estimate must be updated using (15). For the current 3SPR manipulator, Table I reports the required number of elementary arithmetic operations for both methods (these values are obtained using MATLAB®'s built-in function named socFunctionAnalyzer). It is observed that the required arithmetic operations for the JB method are approximately thirty times larger than those for one iteration of the proposed method. The relatively large algebraic expression of the parasitic motions explains the observed discrepancy between the required computation of the JB and the proposed method.

Remark 8. *There is no guarantee of obtaining a solution for the FK of parallel manipulators using an analytical approach. We could not find an analytical solution using standard symbolic software packages for the manipulator we experimented with. Therefore, we only focus on comparing the proposed approach with the Jacobian-based method.*

2.4 Simulations

We run simulations for a 3SPR parallel manipulator with dimensions described in 2.2.3, where the allowable ranges for DoF parameters are $Z \in [0, 100]$ mm, $\alpha \in [-3.5^\circ, 3.5^\circ]$ and $\beta \in [-1.5^\circ, 1.5^\circ]$. To assess the performance of the proposed algorithm under different motion conditions, we consider a combined parabolic, ramp and sinusoidal trajectory. Let $u(t-\tau)$ denote the unit step function delayed by τ and assume that the heave, pitch and roll angles are time-varying functions described by $Z = (50 + 1.6t^2)(u(t) - u(t-2.5)) + (85 - 10t)(u(t-2.5) - u(t-5)) + 15\sin(\frac{\pi}{2}t - 3\pi)$, $\alpha = 0.4t(u(t) - u(t-5)) + 2\cos(0.4\pi t)u(t-5)$

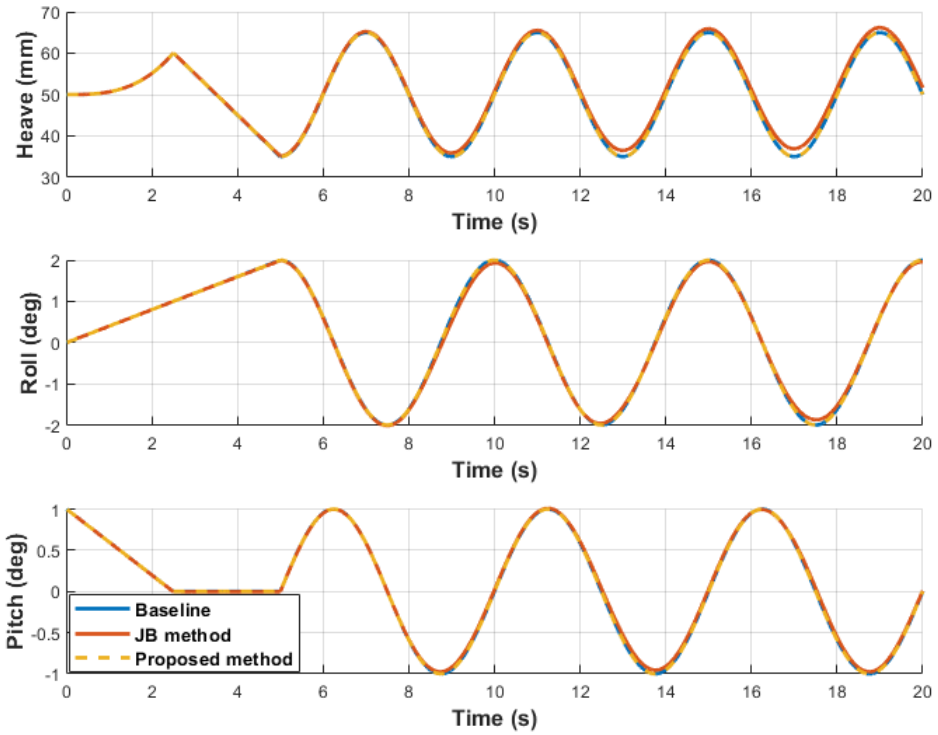

Figure 2.3: Estimation of the FK variables by employing the proposed algorithm with six iterations and the JB method with $f_{\text{pitch}} = 0.2$ Hz

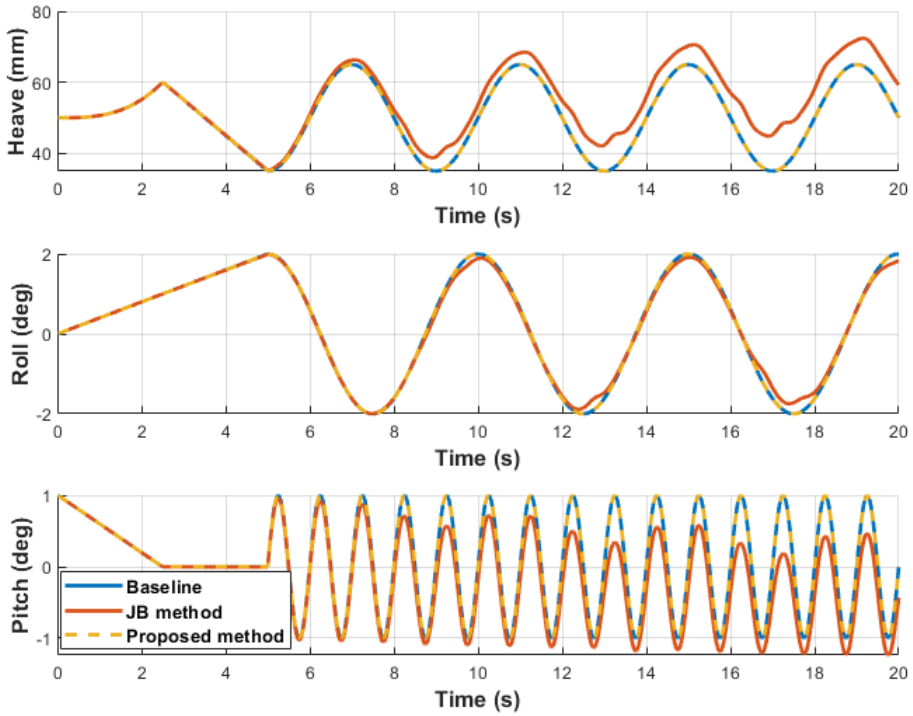

Figure 2.4: Estimation of the FK variables by employing the proposed algorithm with six iterations and the JB method with f_{pitch} = 1 Hz

and $\beta = (-0.4t + 1)(1 - u(t - 2.5)) + \sin(2\pi f_{\text{pitch}} t)u(t - 5)$ for 20 seconds. At each time instant, the estimated heave of each method from the previous instant is used as the initial value for the current instant. For the JB method, we run simulations by taking into account the full information of the parasitic motions. For the proposed method, we select a step size of $\eta = 0.08$. Figs. 2.3 and 2.4 compare the estimated DoF parameters for six iterations of the proposed method with that of the JB method using f_{pitch} = 0.2 Hz and f_{pitch} = 1 Hz, respectively. It is observed that for the ramp and parabolic portions of the trajectory, both methods track the generated motion with satisfactory precision. For the sinusoidal portion of the trajectory with the frequency f_{pitch} = 0.2 Hz in the pitch channel, on the other hand, the JB method estimates the DoF parameters with negligible error, as observed in Fig. 2.3. However, Fig. 2.4 shows that for a frequency as high as f_{pitch} = 1 Hz, the estimates obtained by the JB method exhibit considerable error, while those obtained by the proposed method display much smaller errors. This larger estimation error is partly due to the first-order approximation of the inverse kinematics function in the JB method

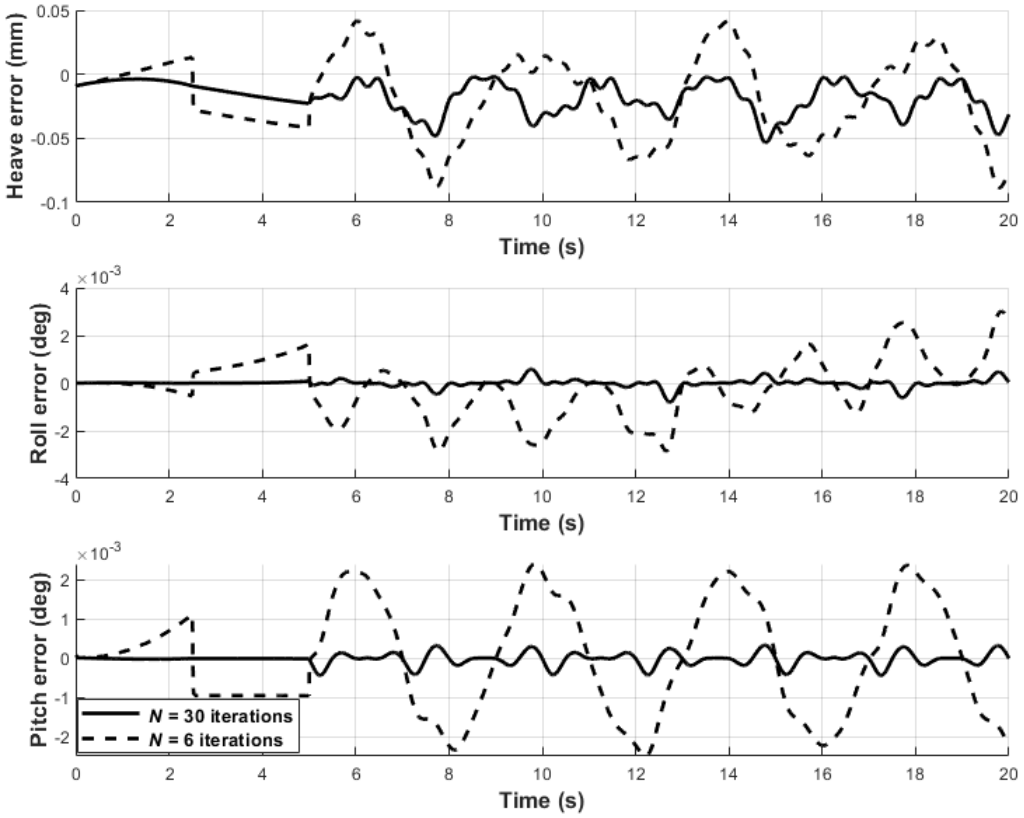

Figure 2.5: Comparison of the estimation error for six and thirty iterations of the proposed algorithm

Table 2.1: Required number of elementary arithmetic operations for the JB and the proposed methods

	$\pm$	$\times/$	Total
JB method	1953	11087	13040
Proposed method (One Iteration)	205	199	404

(i.e., neglecting higher order terms). Note that the first-order approximation error is larger for rapid motions. Finally, Fig. 2.5 illustrates the estimation error of the proposed method for the same trajectory for six and thirty iterations. It is observed that the estimation error is smaller for larger numbers of iterations, as expected. For 30 iterations, the estimation error has a maximum value of 0.05 mm, 5×10^{-4} degrees and 5×10^{-4} degrees in heave, roll and the pitch channels, respectively. Compared to the motion amplitude, the normalized maximum error is 0.16, 0.025, and 0.05 percent in the heave, roll and the pitch channels, respectively.

Remark 9. *In simulations, we observe that both the JB method and the proposed method perform similarly for ramp and parabolic trajectories. However, we observe that the performance of the JB method starts to deviate form the reference trajectory after some time. Note that for most industrial applications, the motion trajectory of the robot is generated from a limited set of well-known functions which usually do not contain high-frequency components. This effective trajectory generation strategy is the main reason for the popularity of the JB method in industrial applications.*

2.5 Conclusions

This chapter investigated the FK estimation of a 3SPR parallel manipulator by using a novel numerical method. The proposed algorithm is computationally more efficient than the Jacobian-based approaches. Furthermore, analytical results are established to evaluate the performance of the proposed algorithm in terms of accuracy, and simulations support the theoretical findings. As a future research direction, one can extend the algorithm to a more general class of parallel manipulators.

Appendix

Algebraic Expression of Parasitic Motions

The parasitic motion along the x axis is a fractional expression $X = \frac{P_x}{Q_x}$ where

$$
\begin{aligned}
Q_x = {} & \cos\alpha\cos\beta\times \\
& \Big(d_2^4\cos^2\alpha + d_3^4\cos^2\beta + d_1^2d_2^2\cos^2\alpha + d_2^4\sin^2\alpha\sin^2\beta \\
& \quad + 2d_1d_2^3\cos^2\alpha + d_1^2d_2^2\sin^2\alpha\sin^2\beta + 2d_2^2d_3^2\cos\alpha\cos\beta \\
& \quad + 2d_1d_2^3\sin^2\alpha\sin^2\beta + 2d_1d_2d_3^2\cos\alpha\cos\beta\Big).
\end{aligned}
$$

and

$$
\begin{aligned}
P_x = {} & d_2^5\cos^4\alpha + d_1^2d_2^3\cos^4\alpha - d_2^5\cos^3\alpha\cos\beta + d_2^5\sin^4\alpha\sin^4\beta + 2d_1d_2^4\cos^4\alpha \\
& - 2d_2^3d_3^2\cos^2\alpha\cos^2\beta + d_1^2d_2^3\sin^4\alpha\sin^4\beta - 2_1d_2^4\cos^3\alpha\cos\beta - d_2d_3^4\cos\alpha\cos^3\beta \\
& + Zd_2^4\cos^3\alpha\sin\beta + 2d_2^5\cos^2\alpha\sin^2\alpha\sin^2\beta - d_1^2d_2^3\cos^3\alpha\cos\beta + d_2d_3^4\cos^2\alpha\cos^2\beta \\
& + 2d_2^3d_3^2\cos^3\alpha\cos\beta + 2d_1d_2^4\sin^4\alpha\sin^4\beta + 4d_1d_2^4\cos^2\alpha\sin^2\alpha\sin^2\beta \\
& - d_2^5\cos\alpha\cos\beta\sin^2\alpha\sin^2\beta + 2d_1d_2^2d_3^2\cos^3\alpha\cos\beta + Zd_1^2d_2^2\cos^3\alpha\sin\beta \\
& + Zd_3^4\cos\alpha\cos^2\beta\sin\beta + 2d_1^2d_2^3\cos^2\alpha\sin^2\alpha\sin^2\beta \\
& - d_2^3d_3^2\cos^2\beta\sin^2\alpha\sin^2\beta - 2d_1d_2^2d_3^2\cos^2\alpha\cos^2\beta + (d_2^4 + d_1d_2^3)Z\cos\alpha\sin^2\alpha\sin^3\beta \\
& + 2Zd_1d_2^3\cos^3\alpha\sin\beta + Zd_1^2d_2^2\cos\alpha\sin^2\alpha\sin^3\beta + Zd_2^2d_3^2\cos\beta\sin^2\alpha\sin^3\beta \\
& + Zd_2^2d_3^2\cos^3\beta\sin^2\alpha\sin\beta + Zd_2^4\cos\alpha\cos^2\beta\sin^2\alpha\sin\beta \\
& - 2d_1d_2^4\cos\alpha\cos\beta\sin^2\alpha\sin^2\beta - 2d_1d_2^2d_3^2\cos^2\beta\sin^2\alpha\sin^2\beta - d_1^2d_2d_3^2\cos^2\beta\sin^2\alpha\sin^2\beta \\
& + (2d_2^3d_3^2 - d_1^2d_2^3)\cos\alpha\cos\beta\sin^2\alpha\sin^2\beta + 2Zd_2^2d_3^2\cos^2\alpha\cos\beta\sin\beta \\
& + Zd_1^2d_2^2\cos\alpha\cos^2\beta\sin^2\alpha\sin\beta + 2d_1d_2^2d_3^2\cos\alpha\cos\beta\sin^2\alpha\sin^2\beta \\
& + Zd_1d_2d_3^2\cos\beta\sin^2\alpha\sin^3\beta + Zd_1d_2d_3^2\cos^3\beta\sin^2\alpha\sin\beta \\
& + 2Zd_1d_2^3\cos\alpha\cos^2\beta\sin^2\alpha\sin\beta + 2Zd_1d_2d_3^2\cos^2\alpha\cos\beta\sin\beta.
\end{aligned}
$$

The parasitic motion along the y axis is a fractional expression $Y = \frac{P_y}{Q_y}$ where

$$
\begin{aligned}
Q_y = &\cos\alpha \times \\
&\Big(d_2^4 \cos^2\alpha + d_3^4 \cos^2\beta + d_1^2 d_2^2 \cos^2\alpha + d_2^4 \sin^2\alpha \sin^2\beta \\
&\quad + 2 d_1 d_2^3 \cos^2\alpha + d_1^2 d_2^2 \sin^2\alpha \sin^2\beta + 2 d_2^2 d_3^2 \cos\alpha \cos\beta \\
&\quad + 2 d_1 d_2^3 \sin^2\alpha \sin^2\beta + 2 d_1 d_2 d_3^2 \cos\alpha \cos\beta \Big).
\end{aligned}
$$

and

$$
\begin{aligned}
P_y = &\, d_2^3 d_3^2 \sin^3\alpha \sin^3\beta + Z d_3^4 \cos^3\beta \sin\alpha \\
&+ Z d_2^4 \cos^2\alpha \cos\beta \sin\alpha + Z d_3^4 \cos\beta \sin\alpha \sin^2\beta \\
&- d_1 d_3^4 \cos^2\beta \sin\alpha \sin\beta - d_2 d_3^4 \cos^2\beta \sin\alpha \sin\beta \\
&+ d_1 d_2^2 d_3^2 \sin^3\alpha \sin^3\beta + d_2^3 d_3^2 \cos^2\alpha \sin\alpha \sin\beta \\
&+ Z d_2^2 d_3^2 \cos\alpha \sin\alpha \sin^2\beta + d_1 d_2^2 d_3^2 \cos^2\alpha \sin\alpha \sin\beta \\
&- d_2^3 d_3^2 \cos\alpha \cos\beta \sin\alpha \sin\beta + 2 Z d_1 d_2^3 \cos^2\alpha \cos\beta \sin\alpha \\
&+ d_2 d_3^4 \cos\alpha \cos\beta \sin\alpha \sin\beta + Z d_1^2 d_2^2 \cos^2\alpha \cos\beta \sin\alpha \\
&+ 2 Z d_2^2 d_3^2 \cos\alpha \cos^2\beta \sin\alpha + Z d_1 d_2 d_3^2 \cos\alpha \sin\alpha \sin^2\beta \\
&- 2 d_1 d_2^2 d_3^2 \cos\alpha \cos\beta \sin\alpha \sin\beta - d_1^2 d_2 d_3^2 \cos\alpha \cos\beta \sin\alpha \sin\beta \\
&+ 2 Z d_1 d_2 d_3^2 \cos\alpha \cos^2\beta \sin\alpha.
\end{aligned}
$$

Part II

Energy Harvesting

Chapter 3

Maximizing Harvested Energy in Coulomb force parametric generators

Miniaturized wearable or implantable medical sensors (or actuators) are important components of the Internet of Things (IoT) in healthcare applications. However, their limited source of power is becoming a bottleneck for pervasive use of these devices, specially, as their functionality increases. Kinetic-based micro energy-harvesters can generate power through the natural human body motion. Therefore, they can be an attractive solution to supplement the source of power in medical wearables or implants. The architecture based on the Coulomb force parametric generator (CFPG) is the most viable micro-harvester solution for generating power from the human motion. This chapter proposes several methods to adaptively estimate the desirable electrostatic force in a CFPG using the input acceleration waveform. Through extensive simulations, the performance of the proposed methods in maximizing the output power of the micro-harvester is evaluated.

3.1 Introduction

Energy harvesting (EH) is the process of capturing energy from the ambient environment and converting it into electrical energy. Different sources for energy harvesting include solar, wind, thermal and kinetic energy. Micro energy harvesters (MEH) refer to a

class of miniaturized EH devices that can generate electrical power for small-scale sensors and actuators as critical components of the Internet-of-Things (IoT) technology [25, 26]. By reducing the frequency of battery replacement or recharge, MEH offers a prolonged operational lifetime or possibly self-sustainability for the IoT sensors and actuators. Integration and co-design of MEH with the sensor architecture has the potential to accelerate the development of green technology that positively impacts the environment. Kinetic-based MEH is considered to be a promising technology for small wearable or implantable devices [26–32]. As the nature of their applications necessitates, these small devices are typically expected to operate for long periods of time without interruptions. This is especially the case for medical implants. Large batteries or frequent recharge might not be practical or feasible for these devices, particularly when connectivity to IoT-Health infrastructure further increases their energy consumption. There are three general methodologies to convert kinetic energy into electrical energy: i) magnetic induction, ii) piezoelectric, and iii) electrostatic. Compared to the first two methods, the electrostatic-based conversion is much more effective in micro scales [33]. Therefore, this approach allows for further miniaturization of the harvester's size, making it more favorable for very small wearable or implantable medical sensors.

Coulomb force parametric generator (CFPG) is a kinetic-based MEH that can best harvest energy from low-frequency nonstationary movements [33–36]. As such, it is considered to be the most suitable architecture for wearables or implants that are placed on or inside the human body. The core component of a CFPG includes a proof mass that can move between two plates. An internal electrostatic force maintained by a transducer holds the proof mass to one of the plates. The proof mass stays attached to the plate until the input acceleration due to the movement of the human body overcomes this holding force. Then, the proof mass is detached and moves toward the other plate. Energy is generated only if the proof mass makes a full flight (i.e., reaching the other plate) against the direction of the electrostatic holding force. If the proof mass fails to make a full flight, the amount of the extracted energy during its flight is dissipated when it returns to its initial position. After each full flight, the direction of the holding force applied to the proof mass reverses, and

the energy harvesting process continues accordingly.

Typically, the magnitude of the holding force is kept constant during this process. However, the authors in [37–39] demonstrated that judicious adjustment of the holding force could significantly increase the output harvested power. The possibility of this adjustment to harvest the maximum amount of energy is especially important for wearable and implantable devices where a limited supply of energy is a critical bottleneck to their usability as well as increasing future functionality.

The authors in [37] investigated the output harvested power of a CFPG for different constant values of the holding force and daily activities. Through statistical analysis of the acceleration waveform generated by the human body movement, an upper bound on the harvested power of a CFPG device was obtained in [38]. The authors in [39] introduced a mathematical model for a more accurate estimation of the generated power by a CFPG. In addition, they formulated an adaptive optimization problem for adjusting the holding force with respect to the input acceleration waveform. The solution to this optimization problem is a mapping from the acceleration data in a given time interval to the optimal value of the holding force that should be used for the following time interval. As such, this type of adaptive adjustment of the holding force can be classified as an online (or dynamic) optimization problem. The underlying assumption in the proposed optimization is the temporal correlation in the acceleration waveform generated by the human body movement for sufficiently short time intervals. The authors in [40], [42] proposed several methodologies including machine learning approaches to solve this optimization problem and compared the harvested power for a limited number of acceleration waveforms.

In this thesis, we propose three methodologies to estimate the optimal value of the electrostatic holding force. First, by formulating a regularized optimization problem, we extend the linear estimator method proposed in [41], [42] to enhance its generalizability to unobserved acceleration data and reduce overfitting with respect to the training data. Then, we investigate the applicability of a multi-armed bandit algorithm to estimate the holding force using the history of the previously applied forces. Finally, by considering

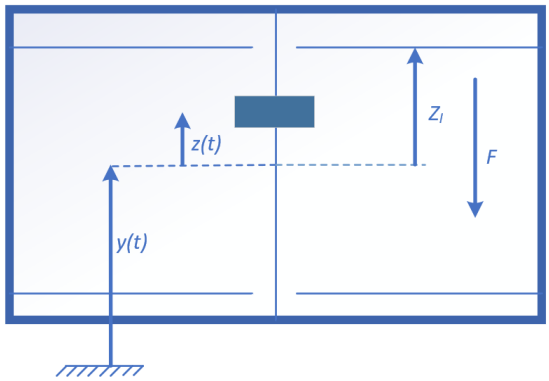

Figure 3.1: The generic model of the core component in a CFPG

the physical constraints of the proof mass, we propose an adaptive algorithm that estimates the holding force without the need for prior training with acceleration data. One important feature of the proposed adaptive methods is their relatively lower computational complexity compared to current methodologies in the literature discussed earlier. Reduced complexity directly impacts the net gain in the harvested energy compared to a constant holding force. The results presented in this thesis are based on a more comprehensive acceleration dataset obtained through physical measurements from different human body movements. This will ensure the maximization of the harvested energy regardless of the placement of the CFPG on the human body.

The rest of this chapter is organized as follows. In Section II, the problem formulation is described. Human acceleration data acquisition and calibration is discussed in Section III. In Section IV, the impact of the range and resolution of the electrostatic force as well as the length of the adaptation interval on the output power are studied. The proposed methods are presented in Sections V followed by a comparative performance evaluation in Section VI. Finally, conclusions and future directions are discussed in Section VII.

3.2 Problem Formulation

In this section, mathematical model of the CFPG and an optimization problem for its output power maximization is provided.

3.2.1 CFPG Mathematical Model

Fig. 3.1 depicts the generic model of the core component of a CFPG where a proof mass is able to move between two plates against the electrostatic holding force denoted by F. The proof mass is attached to either of the plates when the micro-generator is stationary or the external acceleration is not large enough. For sufficiently large external accelerations, the proof mass detaches from one plate and moves towards the other plate. Once the proof mass completes a full journey between the two plates with separation of $2Z_l$, the work done against the electrostatic force is converted to electric energy. When the proof mass reaches the other plate, the direction of the holding force reverses and the energy conversion process continues.

Let the relative position of the proof mass with respect to the device's frame be denoted by $z(t)$. Also, denote $y(t)$ as the device motion with respect to the inertial frame. The following nonlinear differential equation models the dynamics of a CFPG as presented in [39]

$$m\ddot{y}(t) = -m\ddot{z}(t) + F \times \mathrm{R}(z(t)), \tag{23}$$

where m denotes the mass, $\ddot{y}(t)$ denotes the acceleration with respect to the inertial frame, $\ddot{z}(t)$ is the relative acceleration of the proof mass with respect to the frame, and F denotes the electrostatic force (also referred to as the holding force). The reversal of the holding force direction after a full flight of the proof mass is represented by a Relay-Hysteresis function R(.) (see Fig. 3.2). The instantaneous power generated by the proof mass is given by

$$P(t) = F \times \dot{z}(t) \tag{24}$$

where $\dot{z}(t)$ is the relative velocity of the proof mass with respect to the frame.

Remark 10. *As long as the proof mass moves in the opposite direction of the holding force, the instantaneous generated power has a positive value. If the proof mass cannot make a full flight, the motion direction reverses and the instantaneous power will turn negative. If the proof mass returns*

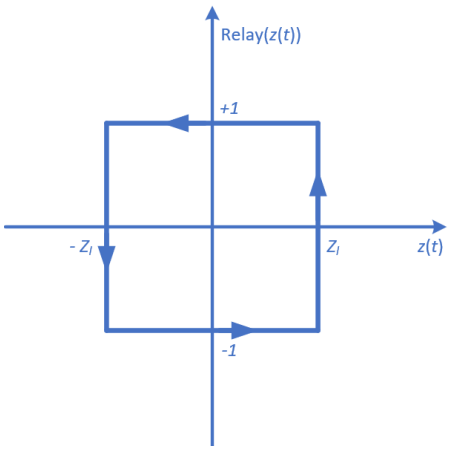

Figure 3.2: Relay-Hysteresis function

to the starting plate, its motion results in a zero-average harvested power.

The average power generated in a CFPG is affected by several factors: the input acceleration, the distance between the two plates, the value of the proof mass, and the magnitude of the electrostatic force. In this thesis, assuming a constant size and geometry for the CFPG component shown in Fig. 3.1, the effect of the holding force on the generated power for various input acceleration will be investigated.

3.2.2 Output Power Optimization

Assume that the holding force can be adjusted every Δ_i seconds. Then, at each time interval, the objective is to estimate the optimal holding force value which maximizes the average harvested power, i.e.,

$$\underset{F^{i+1,\Delta_i},\Delta_i}{\operatorname{argmax}} \left[\frac{1}{\sum_{i=1}^{N} \Delta_i} \times \sum_{i=1}^{N} \int_{t_i-\Delta_i}^{t_i} P(t)dt \right], \tag{25}$$

where Δ_i is the optimal length of the ith decision interval, F^{i+1,Δ_i} is the optimal constant electrostatic force in the $(i+1)$th interval (as a function of Δ_i), N denotes the number of decision intervals and $P(t)$ is the instantaneous output power. Assume, for simplicity, that the length of the decision interval is fixed and denoted by Δ. Then, equation (25) can be

written as

$$\underset{F^{i+1,\Delta}}{\operatorname{argmax}} \left[\frac{1}{N\Delta} \times \sum_{i=1}^{N} \int_{t_0+(i-1)\Delta}^{t_0+i\Delta} P(t)dt \right]. \tag{26}$$

For a fixed decision interval Δ, the optimal electrostatic force is a function of the acceleration waveform at the ith interval $[t_0+(i-1)\Delta, t_0+i\Delta]$. Hence, one way to estimate the holding force $F^{i+1,\Delta}$ is to employ a parametrized policy $\pi_{\boldsymbol{\theta}}$ such that

$$F^{i+1,\Delta} = \pi_{\boldsymbol{\theta}}(\ddot{\boldsymbol{y}}^i), \tag{27}$$

where $\boldsymbol{\theta}$ denotes a vector of parameters for the policy, and $\ddot{\boldsymbol{y}}^i \in \mathbb{R}^M$ denotes a vector of M acceleration samples in the ith interval in (27).

It should be emphasized that the optimization problem (26) can be categorized as an online optimization as knowledge of the future acceleration data is not required for the solution. In other words, $F^{i+1,\Delta}$ is estimated from the information in the ith interval (i.e., past acceleration data). For a known acceleration waveform, the maximum amount of the average harvested power can be obtained by an offline exhaustive search. Although this method cannot be utilized in practice, it provides an upper bound on the maximum harvested power for the given acceleration waveform. For each decision interval i, the optimal value of the holding force can be obtained by the following offline optimization problem

$$F^{\mathrm{Opt},i} = \underset{F^{i,\Delta}\in\mathcal{F}}{\operatorname{argmax}} \left[\frac{1}{N\Delta} \times \sum_{i=1}^{N} \int_{t_0+(i-1)\Delta}^{t_0+i\Delta} P(t)dt \right], \tag{28}$$

where $F^{\mathrm{Opt},i}$ denotes the optimal solution at reach decision interval. We have used this method to assess the effectiveness of the proposed methods.

In practice, the range and resolution of the values of the estimated electrostatic forces that solve the online (or offline) optimization problems (26) (or (28)) are limited. Here, we consider that these values are selected from a finite set $\mathcal{F}$ (hereafter referred to as the decision set) defined by

$$\mathcal{F} = \{F_i | F_{\min} \leq F_i \leq F_{\max}, F_i - F_{i-1} = \delta_F\}, \tag{29}$$

where $F_{\min}$, $F_{\max}$ and δ_F denote the minimum, maximum and the increments for the holding force values. Impact of the decision set and interval on the harvested power will be studied in Section 3.4.1.

3.3 Acceleration Data Acquisition

To evaluate our proposed power maximization methodologies and obtain a realistic measure of the harvested energy, we conducted various physical experiments to acquire human acceleration data. The following subsections describes data acquisition and calibration processes that we have used to prepare a sufficiently diverse dataset of human motion acceleration.

3.3.1 Data Acquisition

To collect acceleration data from the human body motions, the X16-mini triaxial accelerometer made by Gulf Coast Data Concepts, LLC[1] has been used in this study. The dimensions of this device are 51×25×13mm^3. It is small enough to be comfortably placed at various locations on the body and collect data. Body acceleration data is measured along three orthogonal axes. The measurement samples are time-stamped and stored in the device for later retrieval. The sampling rate of the device can be selected to be 12, 25, 50, 100, 200, 400, or 800Hz. Data are collected from various daily physical activities such as walking, jogging, sit-ups, roping, weight exercises and general random movements of hand and shoulder[2]. Data from each activity are collected for 5 minutes with the accelerometer attached on the volunteers' wrist, biceps, leg and chest. To account for changes in the frequency and amplitude of the acceleration waveform, we conducted data collection for three levels of slow, moderate, and intense activities. Fig. 3.3 shows a sample twenty-second acceleration waveform for walking in slow, moderate and fast modes with the accelerometer

[1]Commercial products mentioned in this thesis are merely intended to foster research and understanding. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology.

[2]The experiments were conducted according to the research ethics regulations under the approval number 30013664 at Concordia University and ITL-2021-0273 at NIST.

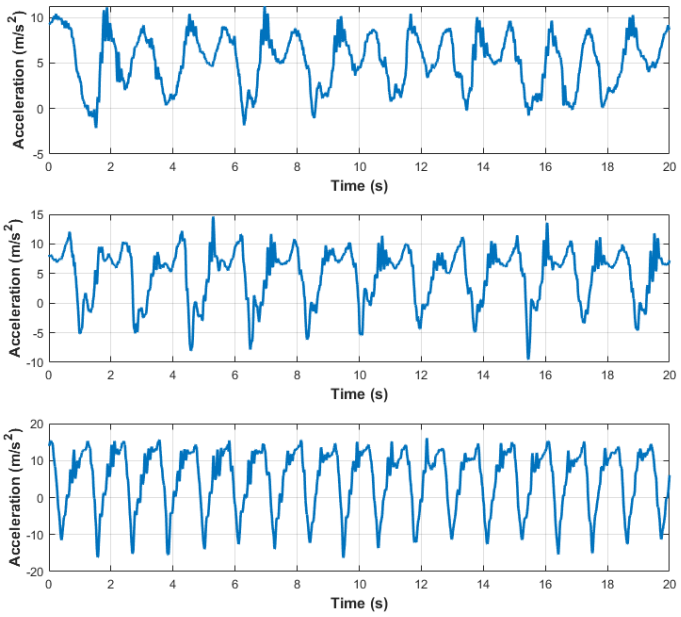

Figure 3.3: Comparison of acceleration waveform for walking in three modes of slow, moderate and fast with the accelerometer attached to the wrist

attached to the wrist.

Remark 11. *For analysis and performance evaluation of the proposed methodologies in this thesis, we have chosen the acceleration data in the z axis; however, similar results were observed using data from other axes as well.*

3.3.2 Accelerometer Calibration

The raw acceleration data are usually subject to various types of noise and bias. Here, we describe a method to calibrate the measurement data from the accelerometer in order to improve its accuracy. When the accelerometer is stationary, the gravity could impact the measurements. The axis that is perpendicular to the ground senses the constant value of 1g (due to earth gravity) while the other two axes should measure a value of zero. Fig. 3.4 depicts the results of our experiment to assess measurement errors. In consecutive time intervals, the static accelerometer is rolled in such way that first the z axis (and then y and finally x) is perpendicular to the ground in order to sense the full impact of the gravity in the direction of each axis. In this way, we can observe the combined effects of bias, scaling

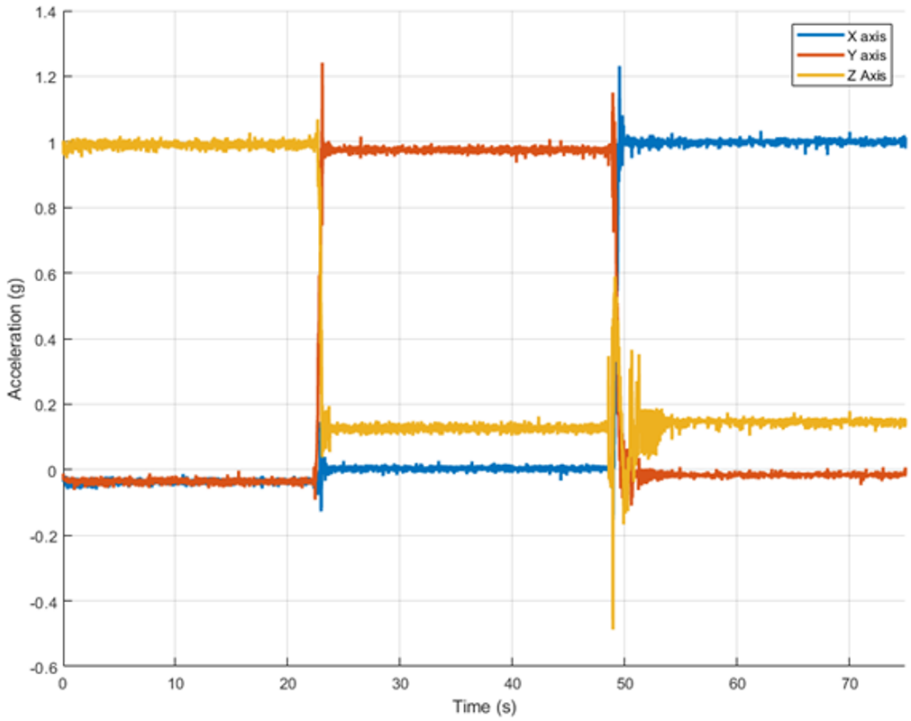

Figure 3.4: The acceleration data while the triaxial accelerometer is static

and cross-axis coupling on the accelerometer data. To account for scale factors and bias, the following model for tri-axial accelerometer is utilized [43]

$$\tilde{\boldsymbol{a}} = \boldsymbol{\Lambda}\boldsymbol{a} + \boldsymbol{\mu}, \tag{30}$$

where $\boldsymbol{a}, \tilde{\boldsymbol{a}} \in \mathbb{R}^3$ denote, respectively, the actual and the measured acceleration vectors along three axes. The matrix $\boldsymbol{\Lambda} \in \mathbb{R}^3$ represents the scaling and cross-axis coupling effects, and $\boldsymbol{\mu} \in \mathbb{R}^3$ is the bias vector. To obtain the exact acceleration vector, (30) is rewritten as

$$\boldsymbol{a} = \boldsymbol{\Lambda}^{-1}(\tilde{\boldsymbol{a}} - \boldsymbol{\mu}). \tag{31}$$

It is desired to calibrate the accelerometer and find the matrix $\boldsymbol{\Lambda}$ and bias vector $\boldsymbol{\mu}$.

Table 3.1: Calibration parameters for the accelerometer

Λ_{11}	1.0428	Λ_{22}	1.0072	μ_1	-0.1314
Λ_{12}	0.0043	Λ_{23}	-0.0069	μ_2	-0.3268
Λ_{13}	-0.0210	Λ_{33}	0.8230	μ_3	1.3532

Considering symmetry in the cross-axis coupling effects, let (30) be rewritten as

$$\tilde{\boldsymbol{a}} = \boldsymbol{A}\boldsymbol{X}, \tag{32}$$

where $\boldsymbol{A} \in \mathbb{R}^{3\times 9}$ is a matrix consisting of the elements of the actual acceleration measurements $\boldsymbol{a}$ and

$$\boldsymbol{X} = [\Lambda_{11} \quad \Lambda_{12} \quad \Lambda_{13} \quad \Lambda_{22} \quad \Lambda_{23} \quad \Lambda_{33} \quad \mu_1 \quad \mu_2 \quad \mu_3]^\intercal, \tag{33}$$

One way to obtain the calibration matrix is to collect multiple acceleration measurements for the accelerometer in a stationary mode. Considering S acceleration samples, equation (32) can be expressed in the augmented form as

$$\tilde{\boldsymbol{A}} = \boldsymbol{\Gamma}\boldsymbol{X}, \tag{34}$$

where $\boldsymbol{\Gamma} = [\boldsymbol{A}_1^\intercal, ..., \boldsymbol{A}_S^\intercal]^\intercal$ and $\tilde{\boldsymbol{A}} = [\tilde{\boldsymbol{a}}_1^\intercal, ..., \tilde{\boldsymbol{a}}_S^\intercal]^\intercal$. For the accelerometer utilized in this study, the calibration parameters were obtained by solving a least-squares problem and the result is reported in Table I.

3.4 Optimal Parameter Selection

In this section, we study the impact of the decision set and the decision interval on the harvested power of the micro-generator.

3.4.1 Impact of the Decision Set

Consider the decision set as defined in (29) with $F_{\min}$= 1 mN, $F_{\max} \in \{10, 20, 30\}$ mN and $\delta_F \in \{0.1, 0.25, 0.5, 1\}$ mN. These values will result in twelve different candidate decision sets. Fig. 3.5 demonstrates the harvested power resulting from each of the candidate decision sets using the offline optimization (28) averaged over acceleration data from various activities discussed in the previous section. As observed, the decision set with the largest range and smallest discretization steps offer approximately 13.9% more harvested energy compared to the set with the smallest range and largest discretization steps (which is approximately thirty times smaller in size). The increase in the harvested energy is achieved at the cost of additional computational complexity to estimate the holding force from larger decision sets. Hence, for lower computational cost, the candidate decision set with the smallest range and largest discretization step is selected to evaluate the performance of our proposed adaptive algorithms, i.e. $\mathcal{F} = \{1, ..., 10\}$ mN. For the simplicity of notation, we also represent the decision set as $\mathcal{F} = \{F_k\}_{k=1}^{10}$, where $F_k = k$ mN, $k \in \mathbb{N}_{10}$.

3.4.2 Impact of the Decision Interval

The length of the decision interval is another parameter that can affect the average harvested power in (26) and (28). To get a better understanding of this impact, Fig. 3.6 shows the average harvested power for different values of Δ and the acceleration waveform resulting from the random movement of the hand. The average power in Fig. 3.6 has been obtained using the offline optimization (28). For comparison, the average power using several constant values of the electrostatic force (i.e., F = 3, 5 and 10 mN) have also been plotted. As Δ increases, the result of the adaptive optimization (28) converges to the optimal constant electrostatic force, which is expected. For the example in Fig. 3.6, this optimal value is 2 mN. The harvested power for this constant holding force will be almost identical to the power generated through offline optimization (28) for $\Delta > 2000$ s.

As expected, the smaller values of the decision interval result in higher harvested power. For example, compared to the optimal constant holding force, a gain of about 130%

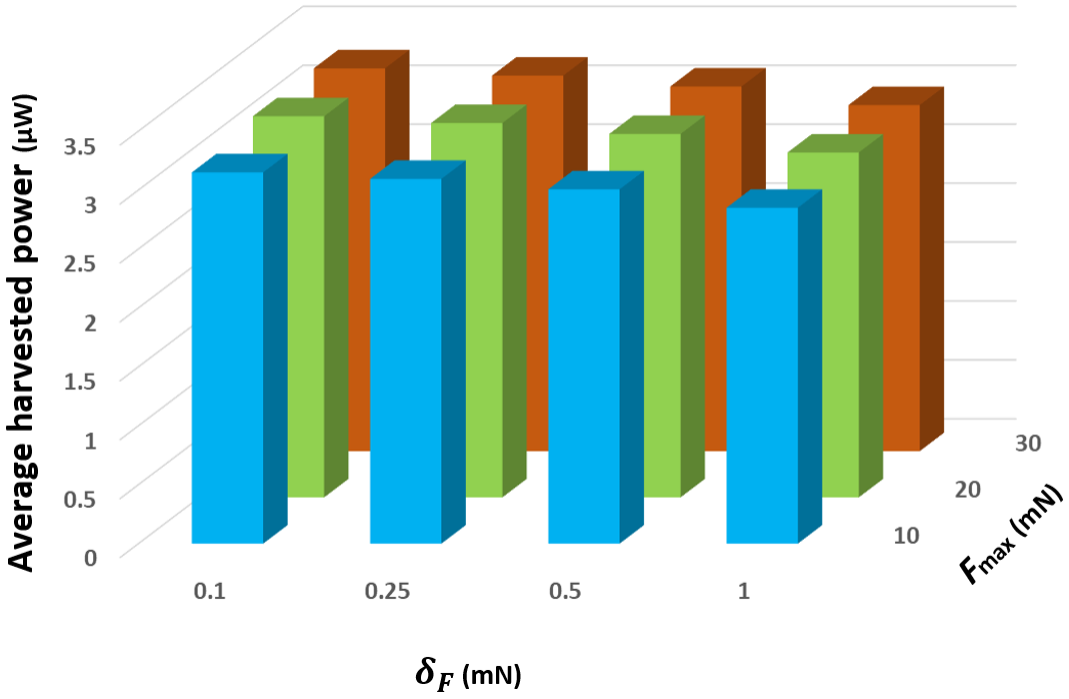

Figure 3.5: Comparison of harvested energy for different maximum holding forces and discretization steps

is observed for $\Delta = 0.5$ s. It should be noted that the value of the optimal constant holding force cannot be obtained without prior knowledge of the whole acceleration waveform. When other constant values are used for the holding force, the possible gain in the harvested power can be much more. For example, a gain of about 400% is observed in Fig. 3.6 when F = 10 mN and $\Delta = 0.5$ s. For very small values of the decision interval ($\Delta < 0.5$ s), a reduction in the harvested power is observed in Fig. 3.6. However, this is mainly due to the limitation in the number, range, and resolution of the elements in the decision set. Without such limitations, the increasing trend of the harvested power with decreasing decision interval would have continued.

Although, choosing smaller decision intervals might seem advantageous, one should consider that smaller intervals are equivalent to more frequent updates of the electrostatic force, requiring more frequent execution of the adaptive optimization algorithm. This will result in more power consumption by the adaptive methodology, reducing the overall output power of the micro-harvester. The trade-off between smaller decision interval to

harvest more power and the decrease in the overall output power due to the consumed energy by the adaptive algorithm module requires further investigation and is outside the scope of this thesis. The specific technology that is used to implement the adaptive methodology is one of the factors that can impact this trade-off. Recent technologies such as neuromorphic processors could be a good candidate to implement the proposed adaptive algorithms with ultra-low power consumption [31,32].

The proper choice of the decision interval also depends on the location of the wearable sensor with integrated micro-harvester on the body as well as the nature of the acceleration data and the time spent on the specific daily activities. Selection of this interval is more difficult for activities that involve non-repetitive motions. Consider the acceleration waveform generated by random movements of the hand shown in Fig. 3.7. The spectral content of this waveform and its cumulative energy in the frequency domain is also shown in Figs. 3.8(a), (b). We conjecture that there is a relationship between the spectral content of the acceleration waveform and the optimal length of the decision interval. Shorter decision intervals could allow an adaptive algorithm to capitalize on high frequency components of the acceleration waveform and harvest more energy, while longer decision intervals limit the algorithm's ability to harvest energy from lower frequencies.

As observed in Fig. 3.8, almost 80% of the acceleration waveform energy is included within [0 0.5] Hz interval, which corresponds to a decision interval of two seconds. Considering the trade-offs mentioned earlier and studying other acceleration waveforms in our dataset, we have selected $\Delta = 2$ s to evaluate and compare the performance of the adaptive methodologies described in the following sections.

3.5 Adaptive Methodologies

In this section, we describe our proposed methodologies that can adaptively estimate the electrostatic force to maximize the harvested power.

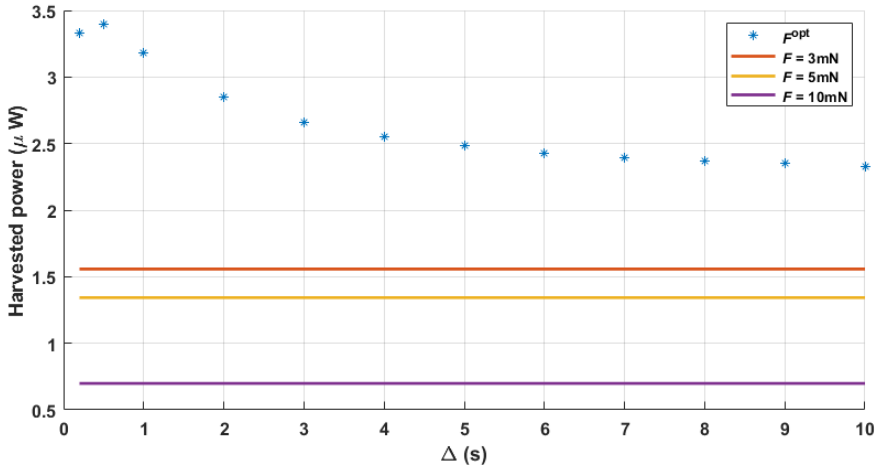

Figure 3.6: Comparison of the harvested power for different decision intervals with F^{opt} and constant holding force F = 3, 5, 10 mN

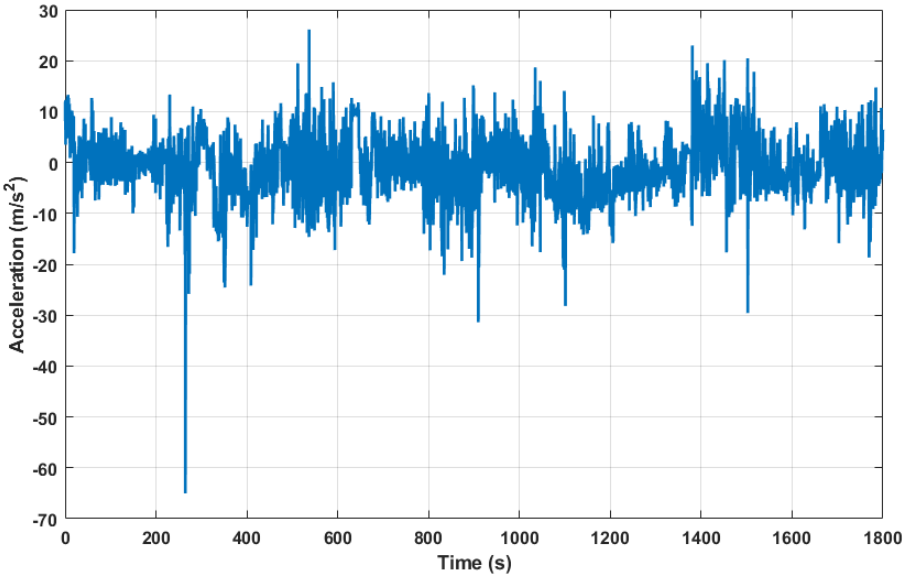

Figure 3.7: Acceleration waveform generated by random movements of the hand when the accelerometer is placed on the wrist

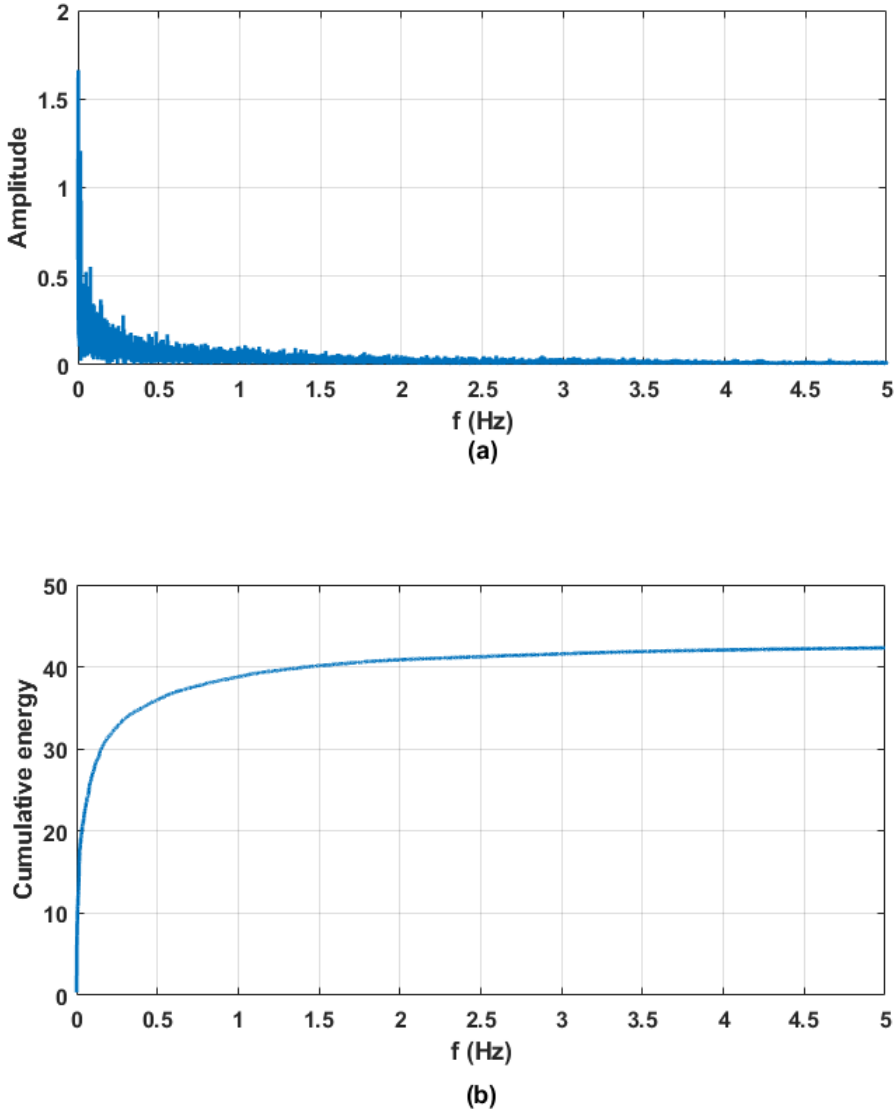

Figure 3.8: (a) Frequency spectrum of the acceleration waveform in Fig. 3.7, (b) Corresponding cumulative waveform energy versus frequency

3.5.1 Linear Estimation of the Holding Force

Consider that the holding force is estimated by a linear mapping from the absolute value of the acceleration data samples during the ith interval as

$$F^{\text{Lin},i} = \boldsymbol{\theta}^{\intercal} |\ddot{\boldsymbol{y}}|^{i-1}, \tag{35}$$

where $\hat{F}^{\text{Lin},i}$ denotes the estimated holding force for the ith interval, and $\ddot{\boldsymbol{y}}^{i-1} \in \mathbb{R}^M$ is the acceleration vector from the $(i-1)$th interval, and $|.|$ denotes the absolute value operator. In addition, $\boldsymbol{\theta} \in \mathbb{R}^M$ is the vector of the linear estimator's parameters. To find the estimation parameters for the electrostatic force $\hat{F}^{\text{Lin},i}$, the average distance between the estimated and actual values of the holding force should be minimized. Therefore, the following minimization problem is formulated

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \quad \|\mathbf{F}^{\text{opt}} - \mathbf{F}^{\text{lin}}\| = \|\mathbf{F}^{\text{opt}} - \boldsymbol{\theta}^{\intercal} \ddot{\mathbf{Y}}\|, \tag{36}$$

where $\mathbf{F}^{\text{opt}} = \left\{ F^{\text{opt},i} \right\}_{i=1}^{N}$ and $\mathbf{F}^{\text{lin}} = \left\{ \hat{F}^{\text{Lin},i} \right\}_{i=1}^{N}$ denote the vectors of the optimal (training label) and estimated holding forces, respectively. The right side of the above equation can be expressed as $\|\mathbf{F}^{\text{opt}} - \boldsymbol{\theta}^{\intercal} \ddot{\mathbf{Y}}\|$, where $\ddot{\mathbf{Y}} \in \mathbb{R}^{M \times N}$ is the matrix of input training data, containing absolute values of M acceleration measurements for N decision intervals.

Considering the L_2-norm in (36), the approach can be simplified to a least-squares problem. For limited acceleration data, we can find the closed-form solution in a computationally efficient manner. In practice, $\ddot{\mathbf{Y}}$ is a tall matrix that can be constructed by down-sampling the measurement dataset for each decision interval. The solution of the least-squares problem in this case is given by

$$\boldsymbol{\theta} = (\ddot{\mathbf{Y}}^{\intercal} \ddot{\mathbf{Y}})^{-1} \ddot{\mathbf{Y}}^{\intercal} \mathbf{F}^{\text{opt}} \tag{37}$$

If the input samples in the least-squares problem are not selected sufficiently distinct from each other, $\ddot{\mathbf{Y}}^{\intercal} \ddot{\mathbf{Y}}$ in (37) may be close to being singular, causing numerical problems.

In addition, the formulation in (36) may lead to overfitting and relatively large norms for the estimator $\boldsymbol{\theta}$. In the case of overfitting, the linear estimator fits well to the training acceleration data but performs poorly for unseen acceleration waveforms.

To keep the size of the estimator's parameters sufficiently small and to avoid possible overfitting, we can add a regularization term to (36) as follows

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \quad \|\mathbf{F}^{\text{opt}} - \mathbf{F}^{\text{lin}}\| = \|\mathbf{F}^{\text{opt}} - \boldsymbol{\theta}^{\intercal}\ddot{\mathbf{Y}}\| + \lambda\|\boldsymbol{\theta}\|, \tag{38}$$

where λ is the regularization constant introducing a trade-off between the minimization of the estimation error and that of the L_2-norm of the estimator vector.

The optimization in (38) is equivalent to solving a maximum likelihood problem with a priori that the parameters are sampled from a zero-mean Gaussian distribution, also known as maximum a posteriori (MAP) estimation [50]. Using Bayesian linear regression (BLR), one can have a broader view of the concept of parameter prior. In addition, instead of seeking a point-estimate of $\boldsymbol{\theta}$, BLR can evaluate the holding force estimation performance for a distribution of linear estimator functions. Here we assume that the estimator parameters are drawn from a Gaussian distribution, i.e.

$$p(\boldsymbol{\theta}) = \mathcal{N}(m_0, S_0),$$

where m_0 and S_0 denote the mean and variance of the distribution. Also, the estimated holding forces are assumed to be drawn from a Gaussian distribution such that

$$p(\hat{F}^{\text{Lin},i}||\ddot{\boldsymbol{y}}|^{\boldsymbol{i-1}}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^{\intercal}|\ddot{\boldsymbol{y}}|^{i-1}, \sigma^2),$$

where σ^2 denotes the measurement noise variance. Given this assumption, information about the distribution of the estimator parameters can be updated. This posterior over the parameters is obtained using the Bayes theorem as

$$p(\boldsymbol{\theta}|\ddot{\mathbf{Y}}, \mathbf{F}^{\text{opt}}) = \frac{p(\mathbf{F}^{\text{opt}}|\ddot{\mathbf{Y}}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{F}^{\text{opt}}|\ddot{\mathbf{Y}})}.$$

The following theorem provides the general form of the posterior over the parameters.

Theorem 2 (Theorem 9.1 in [50]). *Given Assumption 1, the parameter posterior can be computed as*

$$\begin{aligned} p(\boldsymbol{\theta}|\ddot{\mathbf{Y}}, \mathbf{F}^{opt}) &= \mathcal{N}(\boldsymbol{\theta}|m_N, S_N), \\ S_N &= (S_0^{-1} + \sigma^{-2}\ddot{\mathbf{Y}}^\intercal\ddot{\mathbf{Y}}), \\ m_N &= S_N(S_0^{-1}m_0 + \sigma^{-2}\ddot{\mathbf{Y}}^\intercal\mathbf{F}^{opt}). \end{aligned}$$

Having found the updated estimator parameters, the predictive distribution (posterior) for an unseen acceleration data can be obtained as

$$\begin{aligned} p(\hat{F}^{\text{lin},*}|\ddot{\mathbf{Y}}, \mathbf{F}^{\text{opt}}, \ddot{\boldsymbol{y}}^*) &= \int p(\hat{F}^{\text{lin},*}|\ddot{\boldsymbol{y}}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\ddot{\mathbf{Y}}, \mathbf{F}^{\text{opt}}) \mathbf{d}\boldsymbol{\theta} \\ &= \mathcal{N}(\hat{F}^{\text{lin},*}||\ddot{\boldsymbol{y}}^*|^\intercal\boldsymbol{\theta}, |\ddot{\boldsymbol{y}}^*|^\intercal S_N|\ddot{\boldsymbol{y}}|^* + \sigma^2). \end{aligned} \tag{39}$$

Remark 12. *As discussed in Section 3.2.2, the parameterized policy $\pi_{\boldsymbol{\theta}}$ utilizes the acceleration sample vector $\ddot{\boldsymbol{y}}^i$ to estimate the electrostatic holding force. The acceleration vector $\ddot{\boldsymbol{y}}^i$ contains acceleration samples with either positive or negative signs. If the signed values of the acceleration data are employed to learn the parameters of the estimation mapping $\pi_{\boldsymbol{\theta}}$, the learning parameters may not necessarily converge to a stationary value. This leads to large errors in the estimated values of the holding force. Hence, for the Linear Estimation approach, the magnitude of acceleration samples is utilized.*

3.5.2 Estimation with a Multi-Armed Bandit Approach

Given the real-time data processing in the adaptive estimator, one can use multi-armed bandit (MAB) approach as in [45]. Let $F^{\text{MAB},i} \in \mathcal{F}$ denote the estimated holding force at the ith decision interval by this approach. The knowledge about the power distribution resulting from a specific holding force is updated after each decision interval. Denote $P^i(F^{\text{MAB},i})$ as the harvested power at the ith decision interval as a function of a specific value of the holding force. The expected harvested power for the each holding force value

is defined as

$$\bar{P} = \mathbb{E}[P^i(F^{\text{MAB},i})], \tag{40}$$

where the expectation in (40) is taken with respect to all decision intervals. The optimal policy is to always select the holding force with the largest expected reward. To this end, this approach initially selects different holding forces to observe and explore their associated harvested power. With sufficient observations, the near-optimal holding force is selected by exploiting the previously collected information. Therefore, one key aspect of such approach is the trade-off between exploration and exploitation. A variety of algorithms are developed in literature to tackle problems in MAB [46, 47].

For our energy harvesting maximization problem, MAB algorithms with low computational effort are more favorable. Therefore, in this thesis we select the upper confidence bound (UCB) algorithm (shown as Algorithm 1 below) for estimating the holding force.

Algorithm 2 Estimation of the holding force with UCB

Input: The decision set $\mathcal{F} = \{F_k\}_{k=1}^{10}$, the number of decision intervals N, the confidence value c, the set of the average collected power of all actions $\{\bar{P}_k\}_{k=1}^{10}$, and the set of the number of playing all actions $\{N_k\}_{k=1}^{10}$

Output: Estimated holding force $F^{\text{MAB},i}$

Initialize $\{\bar{P}_p\}_{p=1}^{10} = 0$
Initialize $\{N_k\}_{k=1}^{10} = 0$
for $i = 1, ..., 10$ **do**
    Select the ith action $F^{\text{MAB},i} = F_i$
    Evaluate the the harvested power $P^i(F^{\text{MAB},i})$
    $\bar{P}^i = E^i(F^{\text{MAB},i})$
    $N_i \leftarrow N_i + 1$
end for
for $i = 11, ..., N$ **do**
    $F^{\text{MAB},i} = \text{argmax}\, \bar{P}_k + c\sqrt{\frac{\log(i)}{N_k}}$
    Evaluate the the harvested power $P^i(F^{\text{MAB},i})$
    $\bar{P}^i = E^i(F^{\text{MAB},i})$
    $N_i \leftarrow N_i + 1$
end for

Algorithm 3 Min-Max-Based Algorithm

Input: Acceleration samples $\ddot{y}$, decision interval length Δ, acceleration sampling frequency f_s, the decision set $\mathcal{F}$, force margin F_{marg}, total CFPG execution time T, proof mass m

Output: $F^{\text{MM},i}$

$zCross$, list of zero-crossing acceleration samples
Initially empty lists of acceleration $\mathcal{A}^+$ and $\mathcal{A}^-$
k and i (acceleration sample and decision interval counters, respectively)
$F^{1,\Delta} \leftarrow \min \mathcal{F}$
$k \leftarrow 2$
while $t[k] < T$ **do**
    if $\ddot{y}[k]\ddot{y}[k-1] < 0$ **then**
        Append k to $zCross$
    end if
    $indx1 \leftarrow zCross(end-1)$, $indx2 \leftarrow zCross(end)$
    if $\ddot{y}[indx1 : indx2]$ is a positive array **then**
        append max(abs($\ddot{y}[indx1 : indx2]$)) to $\mathcal{A}^+$
    else
        append max(abs($\ddot{y}[indx1 : indx2]$)) to $\mathcal{A}^-$
    end if
    if mod(k,$f_s\Delta$) = 0 **then**
        $i \leftarrow i + 1$
        $F^{i,\Delta^+} \leftarrow \text{argmin}_{F\in\mathcal{F}} |F - (m\bar{\mathcal{A}}^+ - F_{\text{marg}})|$
        $F^{i,\Delta^-} \leftarrow \text{argmin}_{F\in\mathcal{F}} |F - (m\bar{\mathcal{A}}^- - F_{\text{marg}})|$
        $F^{\text{MM},i} = \min\{F^{i,\Delta^+}, F^{i,\Delta^-}\}$
        Empty $\mathcal{A}^+$ and $\mathcal{A}^-$
    end if
    $k \leftarrow k + 1$
end while

3.5.3 Min-Max-Based Adaptive Approach

Considering the optimization problem (26), let $\ddot{y}^{i+}_{\max}$ and $\ddot{y}^{i-}_{\max}$ denote the maximum absolute value of the positive and negative lobes of the acceleration waveform during the ith decision interval, respectively. To harvest energy from the acceleration waveform during the ith decision interval, the optimal value of the electrostatic force must satisfy the following condition

$$\frac{F^{i,\Delta}}{m} < \min\{\ddot{y}^{i+}_{\max}, \ddot{y}^{i-}_{\max}\}. \tag{41}$$

The above inequality implicitly indicates that the electrostatic force must be sufficiently small to make a full flight between the two plates of the CFPG. In other words, the following condition should be taken into account

$$\int_{t_i}^{t_f} \int_{t_i}^{t} \ddot{z} \mathrm{d}\tau \mathrm{d}t \geq 2Z_l. \tag{42}$$

where $\ddot{z} = \frac{F^{i,\Delta}}{m} - \ddot{y}^i$ and t_i and t_f denote the initial and final times of the flight, respectively. Motivated by this observation, Algorithm 2 is proposed to estimate the value of the electrostatic force (i.e., F^{MM}).

To solve optimization problem (26), Algorithm 2 detects the zero-crossings of the acceleration waveform. Then, between each two zero-crossings, the maximum value of the acceleration waveform is obtained. These values are collected in $\mathcal{A}^+$ and $\mathcal{A}^-$ as two lists corresponding to positive and negative portions of the acceleration waveform. Let $\bar{\mathcal{A}}^+$ and $\bar{\mathcal{A}}^-$ denote the average of each list. To account for the full-flight condition in (42), we consider a force margin F_{marg}. Then, the holding forces associated with the posive and negative portions of the acceleration waveform are estimated as $m\bar{\mathcal{A}}^+ - F_{\mathrm{marg}}$ and $m\bar{\mathcal{A}}^- - F_{\mathrm{marg}}$, respectively. Finally, condition (41) gives the electrostatic force as the minimum of the two estimated holding forces and the acceleration lists $\mathcal{A}^+$ and $\mathcal{A}^-$ are reset to empty values.

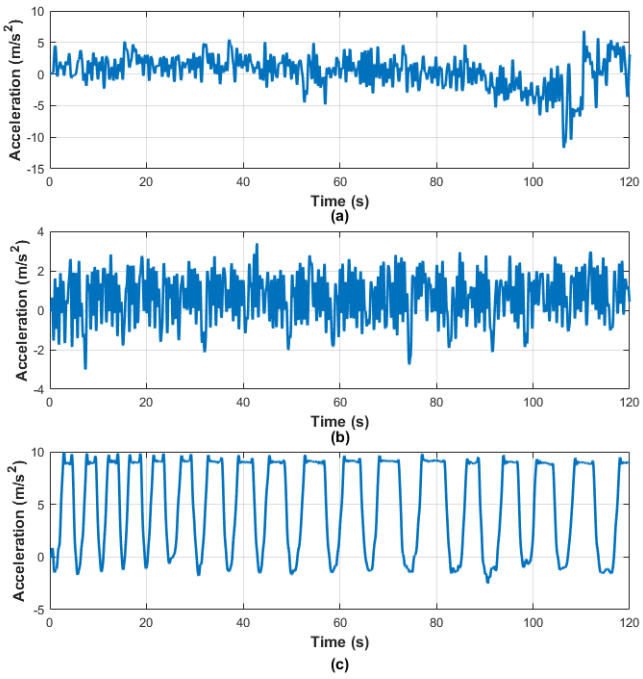

Figure 3.9: Acceleration waveforms as test data collected from (a) a human arm performing random motions, (b) human chest during sit-ups , and (c) a human leg during jogging

3.6 Performance Results

In this section, the performance of the proposed methodologies are assessed using the mathematical model (23) and experimental data. A CFPG with a proof mass of $m = 1$ g and plates separation of $2Z_l = 1$ mm are considered in our evaluations. From the analysis in Subsections 3.4.1 and 3.4.2, we consider the decision set $\{1, ..., 10\}$ mN and interval $\Delta = 2$ s. For the linear estimator, the acceleration data is down-sampled to 4 Hz, implying that $\boldsymbol{\theta} \in \mathbb{R}^8$ for a two-second decision interval. To learn the parameters of the linear estimator, we select 90% of the collected data for training and the rest is used for validation. For the MAB method described by Algorithm I, we consider a confidence factor of $c = 0.2$. Also, the force margin in Algorithm II is set to F_{marg} = 0.5 mN. We first provide the performance results for three scenarios with the accelerometer attached to different parts of the human body while doing different activities. These scenarios are accelerometer on the: human arm while doing random motions (scenario I), human chest while doing sit-ups (scenario II), and human leg during jogging (scenario III). The corresponding acceleration

waveforms are shown in Fig. 3.9.

Fig. 3.10 displays the harvested power for each scenario and the adaptive methodologies. For comparison, the average harvested power when a constant holding force is used is also shown for each scenario (i.e., the red bar). As observed, the linear estimator has the best performance for scenario I. However, Algorithm 2 provides more harvested power for scenarios II and III. One reason for the inferior performance of the linear estimator in these two scenarios is the relatively large asymmetry in their corresponding acceleration waveforms (especially in scenario III). In particular, it does not take into account the magnitude of the acceleration data in selecting the holding force, as shown in (41). Compared to the average harvested power using a constant electrostatic force, the best adaptive approach in scenarios I, II and III offers a gain of about 300%, 400% and 200%, respectively.

Next, we evaluate the performance of the proposed adaptive approaches for a combination of different human activities over a longer period of time. Consider a mix of various human activities as described in Subsection 3.3.1 producing an acceleration waveform with a duration of 4000 s. Fig. 3.11 displays the harvested energy as a result of using our proposed adaptive methodologies on this waveform. For comparison, the average harvested energy using three different constant holding forces is also shown in the Fig. 3.11. Considering the CFPG parameter values as well as the decision set and interval constraints, the upper bound on the harvestable energy (achievable through offline optimization (28)) is also plotted. As observed, among the adaptive approaches, the min-max algorithm performs best, with over 10% more harvested energy compared to the linear estimator. The improvement is due to the fact that this algorithm considers the asymmetry of the acceleration waveform for selecting the proper electrostatic holding force. The min-max adaptive methodology on average generates over 100% more energy compared to the case when a constant holding force is used. This is a promising gain, especially for low-power wearable (or implantable) medical sensors or actuators.

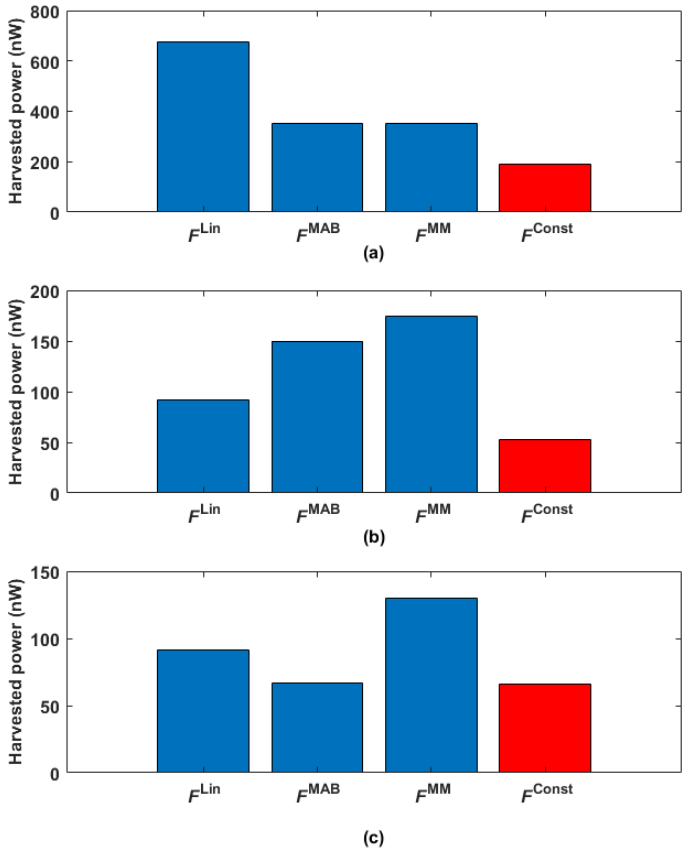

Figure 3.10: Comparison of the harvested power using the proposed adaptive methodologies and constant electromagnetic force: (a) scenario I; (b) scenario II, and (c) scenario III

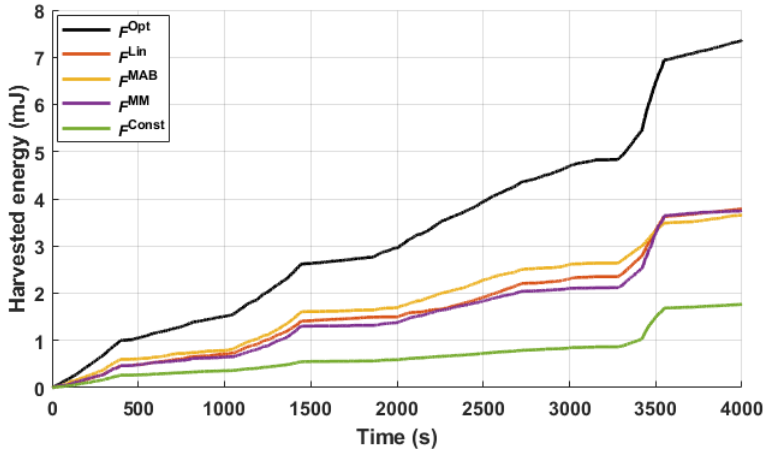

Figure 3.11: Comparison of the harvested energy using the proposed adaptive methodologies for a mixed acceleration waveform corresponding to different human activities with a duration of 4000 s.

3.7 Conclusions

In this chapter, three different adaptive approaches were proposed to increase the harvested power in the Coulomb force parametric generators: the linear estimation method, the multi-armed bandit approach and the min-max technique. The exact performance of the proposed methodologies depends on the nature of the acceleration waveform. However, in almost all practical scenarios and on average, there is a noticeable improvement in the harvested kinetic power from the human body motion compared to the case when a constant electrostatic force is used. The additional harvested power could easily supply the required resource to run a low-complexity adaptive algorithm as part of the CFPG architecture. The Min-Max algorithm proposed here could be a good candidate that not only incurs very low implementation complexity but also results in a significant gain in the harvested power in most scenarios.

Knowledge of the exact location of the medical sensor on or inside the body could provide more specific information about the characteristics of the acceleration waveform that an embedded micro-harvester would experience. This information can help to further optimize the adaptive approach with the proper choice of the decision set. Also, the relationship between the best adaptation interval and the spectral content of the acceleration waveform requires additional exploration in future studies.

Chapter 4

An Asymmetric Adaptive Approach to Enhance Output Power in Kinetic-Based Microgenerators

Following the previous chapter on CFPG micro-energy harvesters, in this chapter, we formulate the energy maximization problem in a new framework. We consider an asymmetric adaptive approach to estimate the electrostatic force in a CFPG using the acceleration waveform. Simulations using human motion measurements show that the proposed approach achieves considerable gain in the harvested energy compared to the previously studied symmetric adaptive methodologies.

4.1 Introduction

In Chapter 3, we studied output energy maximization in CFPGs using (26). Although the formulation in (26) provides some insight into the problem of optimal tuning of the electrostatic force, it obtains the same value of F in both flight directions of the proof mass. Observations show that such formulation cannot extract energy from specific acceleration waveforms efficiently (see Fig. 4.1). It is observed that the acceleration waveform has assymetric positive an negative lobes. Obviously the proof mass can make a full flight

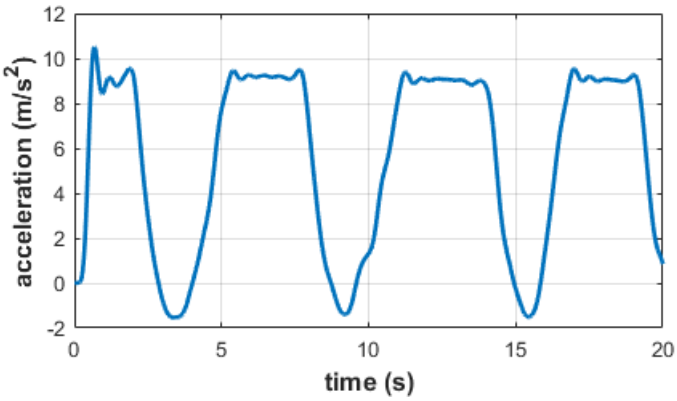

Figure 4.1: A twenty-second sample acceleration collected from an accelerometer attached to a human leg during jogging

with smaller values of the holding force in the negative lobe of the acceleration. However, relatively larger acceleration waveform in the positive lobe allows the use of larger F. Therefore, for a constant decision interval Δ, choosing a pair of holding force values associated with upward and downward flight directions can extract higher amounts of harvested energy. Next, we introduce a new asymmetric adaptive strategy for the electrostatic force.

4.2 Proposed Methodology

A necessary condition for harvesting energy in a CFPG is a full flight of the proof mass from one plate to the other. From the dynamics of a CFPG in (23), this condition is met when the input acceleration $\ddot{y}$ changes sign and is sufficiently large to overcome the holding force. In particular, a full flight is achieved if the relative acceleration of the proof mass $\ddot{z}(t)$ satisfies the following condition,

$$\int_{t_i}^{t_f} \int_{t_i}^{t} \ddot{z}(\tau) d\tau dt \geq 2Z_l, \tag{43}$$

where t_i and t_f denote the initial and final times of the flight, respectively.

By observing the asymmetry in the acceleration signal shown in Fig. 4.2, we can conclude that using unequal holding forces for different directions of the proof mass flights could

lead to higher generated power. As such, we propose the following optimization problem

$$\underset{F_{\text{dn}}^{i,\Delta},F_{\text{up}}^{i,\Delta}}{\operatorname{argmax}} \left[\frac{1}{N\Delta} \times \sum_{i=1}^{N} \int_{t_0+(i-1)\Delta}^{t_0+i\Delta} P(t)dt\right], \tag{44}$$

where $(F_{\text{dn}}^{i,\Delta}, F_{\text{up}}^{i,\Delta})$ denotes the optimal pair of the downward and upward holding forces at the ith interval. To solve the optimization problem in (44), we propose Algorithm 1 described in the next section to estimate the optimal values of the holding force.

Algorithm 4 Online estimation of the holding force

Input: Acceleration samples $\ddot{y}$, decision interval length Δ, acceleration sampling frequency f_s, holding force permissible set $\mathcal{F}$, force margin F_{marg}, total simulation time T, proof mass m

Output: $(F_{\text{dn}}^{i,\Delta}, F_{\text{up}}^{i,\Delta})$

$zCross$, list of zero-crossing acceleration samples
Initially empty lists of acceleration $\mathcal{A}^+$ and $\mathcal{A}^-$
k and i, acceleration sample and decision interval counter
$F_{\text{dn}}^{1,\Delta} \leftarrow \min \mathcal{F}$ and $F_{\text{up}}^{1,\Delta} \leftarrow \min \mathcal{F}$
$k \leftarrow 2$
while $t[k] < T$ **do**
    if $\ddot{y}[k]\ddot{y}[k-1] < 0$ **then**
        Append k to $zCross$
    end if
    $indx1 \leftarrow zCross(end-1)$, $indx2 \leftarrow zCross(end)$
    if $\ddot{y}[indx1 : indx2]$ is a positive array **then**
        append max(abs($\ddot{y}[indx1 : indx2]$)) to $\mathcal{A}^+$
    else
        append max(abs($\ddot{y}[indx1 : indx2]$)) to $\mathcal{A}^-$
    end if
    if mod(k,$f_s\Delta$) = 0 **then**
        $i \leftarrow i + 1$
        $F_{\text{dn}}^{i,\Delta} \leftarrow \operatorname{argmin}_{F\in\mathcal{F}} |F - (m\bar{\mathcal{A}}^+ - F_{\text{marg}})|$
        $F_{\text{up}}^{i,\Delta} \leftarrow \operatorname{argmin}_{F\in\mathcal{F}} |F - (m\bar{\mathcal{A}}^- - F_{\text{marg}})|$
        Empty $\mathcal{A}^+$ and $\mathcal{A}^-$
    end if
    $k \leftarrow k + 1$
end while

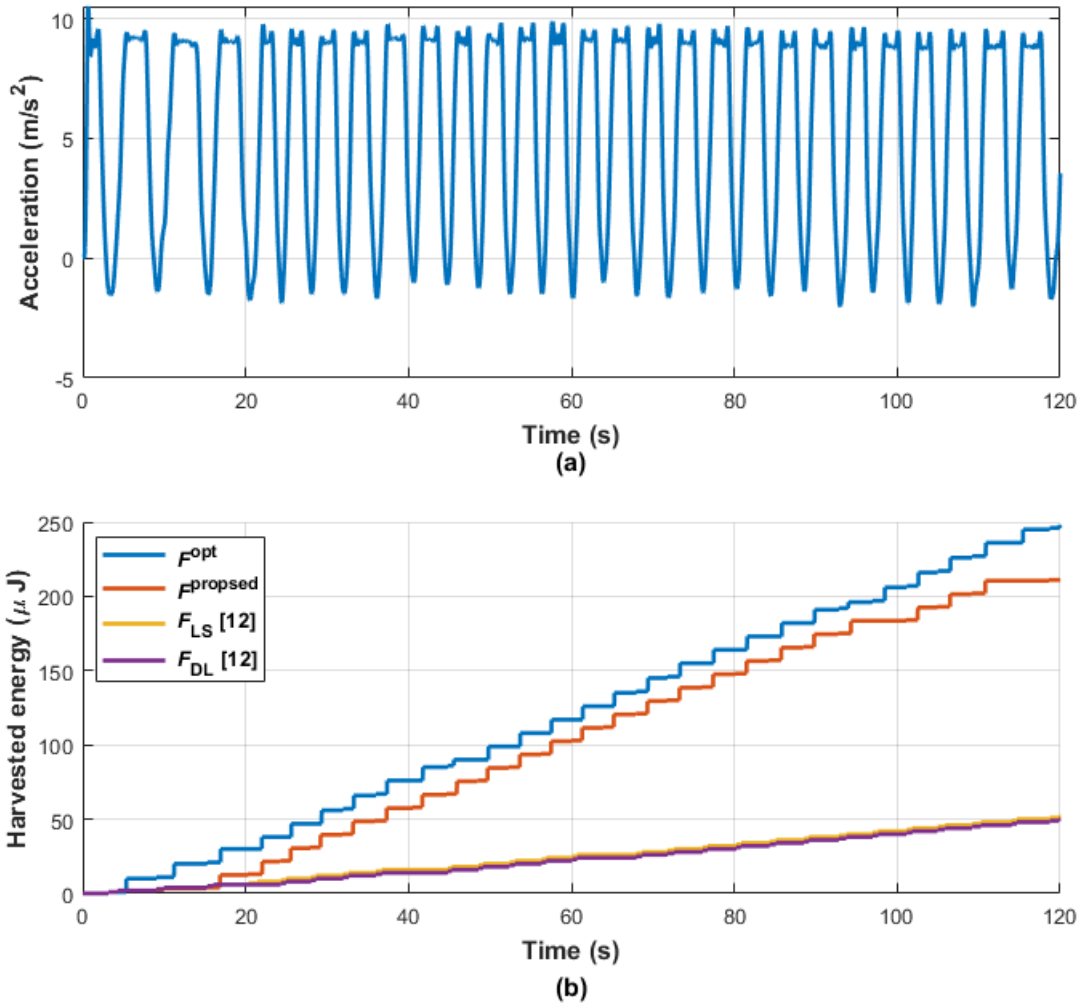

Figure 4.2: Scenario I: (a) Acceleration waveform, and (b) the resulting harvested energy for a sample acceleration collected from an accelerometer attached to a human leg during jogging

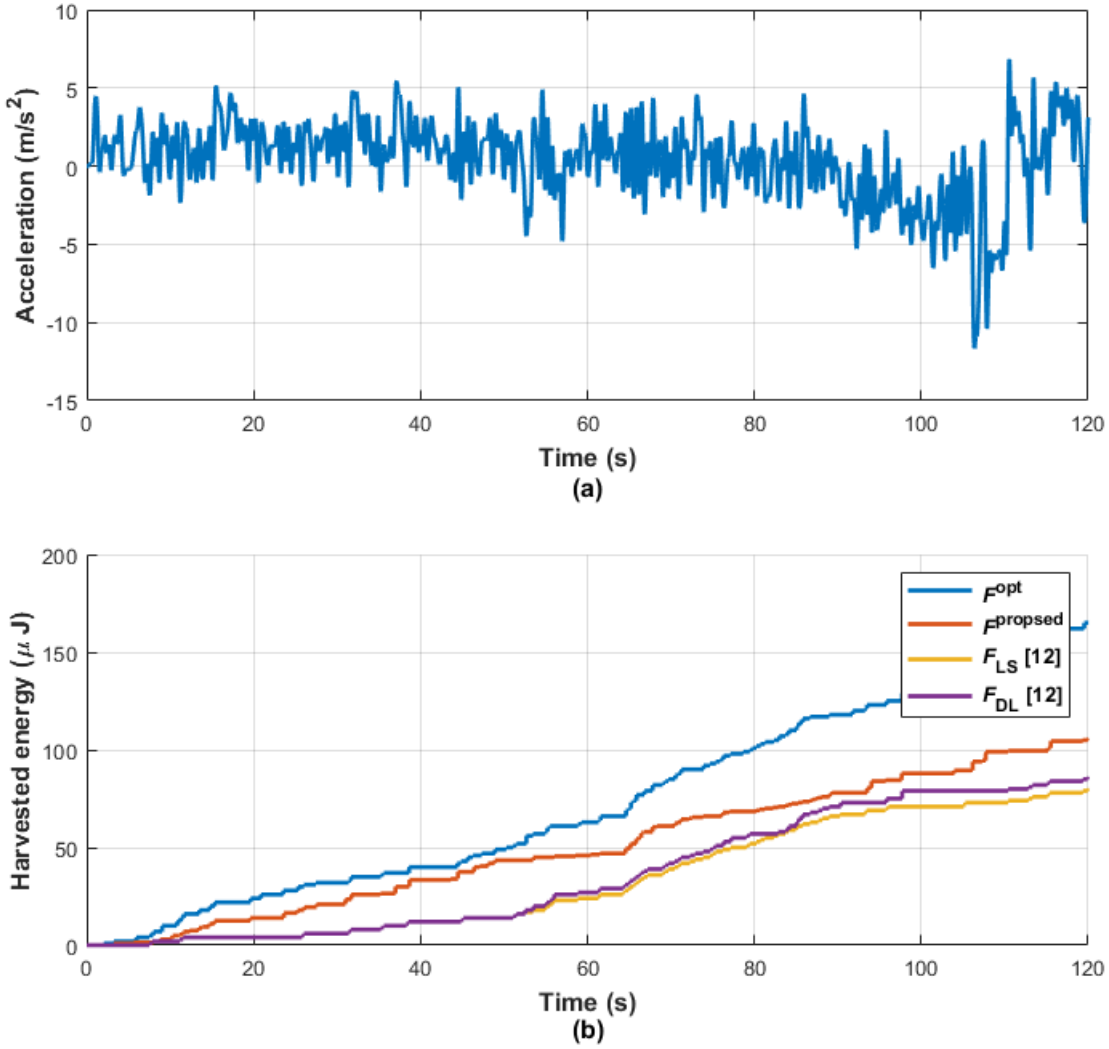

Figure 4.3: Scenario II: (a) Acceleration waveform, and (b) the resulting harvested energy for a sample acceleration collected from an accelerometer attached to a human arm performing random motions

4.3 Simulations

It is assumed that $(F_{\text{dn}}^{i,\Delta}, F_{\text{up}}^{i,\Delta})$ belong to a set of predefined values of holding forces denoted by $\mathcal{F}$. To solve (44), Algorithm 1 uses observations of the acceleration waveform for a fixed decision interval Δ and applies the estimated holding forces to the next decision time interval. First, the algorithm detects the zero-crossings of the acceleration waveform. Then, between each two zero-crossings, the maximum value of the acceleration waveform is obtained. These values are collected in $\mathcal{A}^+$ and $\mathcal{A}^-$ as two lists corresponding to positive and negative portions of the acceleration waveform. Let $\bar{\mathcal{A}}^+$ and $\bar{\mathcal{A}}^-$ denote the average of each list. As an initial guess, $(F_{\text{dn}}^{i,\Delta}, F_{\text{up}}^{i,\Delta})$ may be approximated by $(m\bar{\mathcal{A}}^-, m\bar{\mathcal{A}}^+)$ for the next decision interval. However, such a simple guess leads to overestimated holding force that do not let the proof mass detach from the CFPG plates to make a full flight. To account for such effect, we consider a force margin F_{marg}, i.e. the holding forces are estimated as $(m\bar{\mathcal{A}}^- - F_{\text{marg}}, m\bar{\mathcal{A}}^+ - F_{\text{marg}})$.

We run simulations for a CFPG with $m = 1$ g and $Z_l = 1$ mm. We consider $\mathcal{F} = \{0.5, 1, ..., 10\}$ mN and $F_{\text{marg}} = 0.5$ mN and $\Delta = 2$ s. We report performance of the proposed algorithm for two scenarios. Figs. 4.2 and 4.3 depict 120-second sample acceleration waveforms collected from an accelerometer attached to a human leg while jogging (scenario I), and a human arm while making random moves (scenario II). In both scenarios, we compare the harvested energy using our proposed algorithm with 1) the case where the optimal asymmetric electrostatic force (F^{opt}) in Equation (44) is calculated using exhaustive search, and 2) the cases where the symmetric electrostatic force is estimated by the linear regression (F_{LS}) and the deep learning approach (F_{DL}) in [42]. For scenario I, we observe that the proposed algorithm, F_{LS}, and F_{DL} harvest 89 %, 20.1 %, and 20.3 % of the maximum achievable energy which is obtained under (F^{opt}), respectively. For scenario II, the percentages of harvested energy under the proposed algorithm, F_{LS}, and F_{DL} are 62 %, 53 %, and 49 %, respectively.

4.4 Conclusions

In this chapter, a new approach for maximization of the harvested power in a CFPG has been proposed and studied. The performance of the approach was investigated for different acceleration waveforms. The results indicate that exploiting asymmetry in the acceleration waveform could lead to significant gain in the harvested energy. An important feature of the proposed approach is its low computational complexity for implementation in the next generation of CFPGs.

Part III

Linear Quadratic Mean-Field Systems

Chapter 5

Reinforcement Learning in Nonzero-sum Linear Quadratic Deep Structured Games: Global Convergence of Policy Optimization

We study model-based and model-free policy optimization in a class of nonzero-sum stochastic dynamic games called linear quadratic (LQ) deep structured games. In such games, players interact with each other through a set of weighted averages (linear regressions) of the states and actions. In this chapter, we focus our attention to homogeneous weights; however, for the special case of infinite population, the obtained results extend to asymptotically vanishing weights wherein the players learn the sequential weighted mean-field equilibrium. Despite the non-convexity of the optimization in policy space and the fact that policy optimization does not generally converge in game setting, we prove that the proposed model-based and model-free policy gradient descent and natural policy gradient descent algorithms globally converge to the sub-game perfect Nash equilibrium. To the best of our knowledge, this is the first result that provides a global convergence proof of policy optimization in a nonzero-sum LQ game. One of the salient features of the

proposed algorithms is that their parameter space is independent of the number of players, and when the dimension of state space is significantly larger than that of the action space, they provide a more efficient way of computation compared to those algorithms that plan and learn in the action space. Finally, some simulations are provided to numerically verify the obtained theoretical results.

5.1 Introduction

In recent years, there has been a growing interest in the application of reinforcement learning (RL) algorithms in networked control systems. One of the most popular reinforcement learning (RL) algorithms in practice is policy gradient, due to its stability and fast convergence. However, from the theoretical point of view, there is not much known about it. Recently, it is shown in [51] that a single-agent linear quadratic (LQ) optimal control problem enjoys the global convergence, despite the fact that the optimization problem is not convex in the policy space. A similar result is obtained for zero-sum LQ games in [52]. On the other hand, a nonzero-sum LQ game is more challenging than the above problems, where the existing results on the global (or even local) convergence of the policy gradient methods are generally not encouraging [53]. Specifically, the authors in [53] consider the general form of a mean-field problem and argue that it has no solution. However, in this chapter, we show that under certain conditions on the system (i.e., evolution of the agents' states in linear mean-field dynamics), there exists a solution for the mean-field problem. Specifically, we show that under this structure, the problem can be addressed by solving two Riccati equations with the same dimension as the individual players' state space, independent of the number of players. Hence, the computational complexity does not grow with the number of players for this specific structure.

Inspired by recent developments in deep structured teams and games [55,56,58,68,82], we study a class of LQ games wherein the effect of other players on any individual player is characterized by a linear regression of the states and actions of all players. The closest field of research to deep structured games is mean-field games [59]. In a classical LQ

mean-field game, one often has: (a) homogeneous individual weights (i.e., players are equally important); (b) the number of players n is asymptotically large with independent primitive random variables (to be able to predict the trajectory of the mean-field using the strong law of large numbers); (c) the coupling is through the mean of the states, where the control coupling (called extended coupling) is more challenging; (d) the proof technique revolves around the fact that the effect of a single player on others is negligible, reducing the game to a coupled forward-backward optimal control problem; (e) the solution concept is Nash equilibrium; (f) given some fixed-point conditions across the time horizon, the forward-backward equation admits a solution leading to an approximate Nash in the finite-population game; (g) they are often not practical for long-horizon and reinforcement learning applications wherein the common practice is to adopt a weaker solution concept called stationary Nash equilibrium (where the trajectory of the mean-field is stationary), and (h) since the results are asymptotic, the models are limited to those that are uniformly bounded in n. In contrast to mean-field game, LQ deep structured game often has: (a') heterogeneous individual weights that are not necessarily homogeneous; (b') the number of players is arbitrary (not necessarily very large) with possibly correlated primitive random variables; (c') the coupling is through the weighted mean of the states and actions; (d') the proof technique revolves around a gauge transformation initially proposed in [60] (not based on the negligible effect); (e') the solution concept is sequential Nash; (f') the solution is exact (not an approximate one) for any arbitrary number of players and it is identified by Riccati equations; (g') since the solution concept is sequential, it is well suited for long-horizon and reinforcement learning, and (h') since the results are also valid for finite-population game, the dynamics and cost are not necessarily limited to uniformly bounded functions with respect to n. It is shown in [55] that the classical LQ mean-field game with the tracking cost formulation is a special case of deep structured games under standard conditions, where the mean-field equilibrium coincides with the sequential mean-field equilibrium. It is to be noted that the LQ mean-field-type game [61–63] is a single-agent control problem (i.e., it is not a non-cooperative game), which resembles a team problem

with social welfare cost function.[1] In particular, it may be viewed as a special case of risk-neutral LQ mean-field teams introduced in [60], showcased in [64–68], and extended to deep structured LQ teams in [55]. The interested reader is referred to [55, Section VI] for more details on similarities and differences between mean-field games, mean-field-type games and mean-field teams.

The rest of the chapter is organized as follows. In Section 5.2, the problem of LQ deep structured game is formulated. In Section 5.3, the global convergence of model-based and model-free policy gradient descent and natural policy gradient descent algorithms are presented. In Section 5.4, some numerical examples are provided to validate the theoretical results. The chapter is concluded in Section 5.5.

5.2 Problem Formulation

Throughout the chapter, $\mathbb{R}$, $\mathbb{R}_{>0}$ and $\mathbb{N}$ refer to the sets of real, positive real and natural numbers, respectively. Given any $n \in \mathbb{N}$, $\mathbb{N}_n$, $x_{1:n}$ and $\mathbf{I}_{n\times n}$ denote the finite set $\{1,\dots,n\}$, vector $(x_1,\dots,x_n)$ and the $n \times n$ identity matrix, respectively. $\|\cdot\|$ is the spectral norm of a matrix, $\|\cdot\|_F$ is the Frobenius norm of a matrix, $\mathrm{Tr}(\cdot)$ is the trace of a matrix, $\sigma_{\min}(\cdot)$ is the minimum singular value of a matrix, $\rho(\cdot)$ is the spectral radius of a matrix, and $\mathrm{diag}(\Lambda_1,\Lambda_2)$ is the block diagonal matrix $[\Lambda_1 \quad 0; 0 \quad \Lambda_2]$. For vectors x, y and z, $\mathrm{vec}(x,y,z) = [x^\intercal, y^\intercal, z^\intercal]^\intercal$ is a column vector. The superscript $-i$ refers to all players except the i-th player. In addition, $\mathrm{poly}(\cdot)$ denotes polynomial function.

Consider a nonzero-sum stochastic dynamic game with $n \in \mathbb{N}$ players. Let $x_t^i \in \mathbb{R}^{d_x}$, $u_t^i \in \mathbb{R}^{d_u}$ and $w_t^i \in \mathbb{R}^{d_x}$ denote the state, action and local noise of player $i \in \mathbb{N}_n$ at time $t \in \mathbb{N}$, where $d_x, d_u \in \mathbb{N}$. Define the weighted averages:

$$\bar{x}_t := \frac{1}{n}\sum_{i=1}^{n} \alpha_n^i x_t^i, \quad \bar{u}_t := \frac{1}{n}\sum_{i=1}^{n} \alpha_n^i u_t^i, \tag{45}$$

where $\alpha_n^i \in \mathbb{R}$ is the *influence factor* (weight) of player i among its peers. From [55, 82], we

[1] When the mean field is replaced by the expectation of the state of the genetic player, the resultant problem is called mean-field-type game.

refer to the above linear regressions as *deep state* and *deep action* in the sequel. To ease the presentation, the weights are normalized as follows: $\sum_{i=1}^{n} \alpha_n^i = 1$.

The initial states $\{x_1^1, \ldots, x_1^n\}$ are random with finite covariance matrices. The evolution of the state of player $i \in \mathbb{N}_n$ at time $t \in \mathbb{N}$ is given by:

$$x_{t+1}^i = Ax_t^i + Bu_t^i + \bar{A}\bar{x}_t + \bar{B}\bar{u}_t + w_t^i, \tag{46}$$

where $\{w_t^i\}_{t=1}^{\infty}$ is an i.i.d. zero-mean noise process with a finite covariance matrix. The primitive random variables $\{\{x_1^i\}_{i=1}^n, \{w_1^i\}_{i=1}^n, \{w_2^i\}_{i=1}^n, \ldots\}$ are defined on a common probability space and are mutually independent across time. The above random variables can be non-Gaussian and correlated (not necessarily independent) across players. The cost of player $i \in \mathbb{N}_n$ at time $t \in \mathbb{N}$ is given by:

$$\begin{aligned} c_t^i =& (x_t^i)^\intercal Q x_t^i + 2(x_t^i)^\intercal S^x \bar{x}_t + (\bar{x}_t)^\intercal \bar{Q} \bar{x}_t \\ &+ (u_t^i)^\intercal R u_t^i + 2(u_t^i)^\intercal S^u \bar{u}_t + (\bar{u}_t)^\intercal \bar{R} \bar{u}_t, \end{aligned} \tag{47}$$

where $Q, S^x, \bar{Q}, R, S^u$ and $\bar{R}$ are symmetric matrices with appropriate dimensions.

From [55, 82], an information structure called *deep state sharing* (DSS) is considered wherein each player $i \in \mathbb{N}_n$ at any time $t \in \mathbb{N}$ observes its local state x_t^i and the deep state $\bar{x}_t$, i.e., $u_t^i = g_t^i(x_{1:t}^i, \bar{x}_{1:t})$, where g_t^i is a measurable function adapted to the filteration of the underlying primitive random variables of $\{x_{1:t}^i, \bar{x}_{1:t}\}$. When the number of players is very large, one can use *no-sharing* (NS) information structure wherein each player observes only its local state. However, such a fully decentralized information structure comes at a price that one must predict the trajectory of the deep state in time (which introduces the computational complexity in time horizon in terms of storage and computation). For example, if the dynamics of the deep state (i.e., $A + \bar{A}$ and $B + \bar{B}$) is known (which is not applicable for model-free applications), the deep state can be predicted a head of time when primitive random variables are mutually independent by the strong law of large numbers. Alternatively, one can assume to have access to an external simulator for the dynamics of the deep state (which is basically DSS structure). In this chapter, we focus on

DSS information structure wherein there is no loss of optimality in restricting attention to stationary strategies despite the fact that the deep state is not stationary. The interested reader is referred to [82] for the convergence analysis of NS (approximate) solution to the DSS solution, as $n \to \infty$.

Define $\mathbf{g}_n^i := \{g_t^i\}_{t=1}^{\infty}$ and $\mathbf{g}_n := \{\mathbf{g}^1, \dots, \mathbf{g}^n\}$. The admissible set of actions are square integrable such that $\mathbb{E}\big[\sum_{t=1}^{\infty} \gamma^{t-1}(u_t^i)^\intercal u_t^i\big] < \infty$. Given a discount factor $\gamma \in (0,1)$, the cost-to-go for any player $i \in \mathbb{N}_n$ is described by:

$$J_{n,\gamma}^i(\mathbf{g}_n^i, \mathbf{g}_n^{-i})_{t_0} = (1-\gamma)\mathbb{E}\big[\sum_{t=t_0}^{\infty} \gamma^{t-1} c_t^i\big], \quad t_0 \in \mathbb{N}. \tag{48}$$

Problem 2. *Suppose that the weights are homogeneous, i.e.* $\alpha_n^i = \frac{1}{n}$, $i \in \mathbb{N}_n$. *When a sequential Nash strategy* $\mathbf{g}_n^*$ *exists, develop model-based and model-free gradient descent and natural policy gradient descent procedures under DSS information structure such that for any player* $i \in \mathbb{N}_n$ *at any stage of the game* $t_0 \in \mathbb{N}$, *and any arbitrary strategy* $\mathbf{g}^i$*:*

$$J_{n,\gamma}^i(\mathbf{g}_n^{*,i}, \mathbf{g}_n^{*,-i})_{t_0} \le J_{n,\gamma}^i(\mathbf{g}^i, \mathbf{g}_n^{*,-i})_{t_0}. \tag{49}$$

Remark 13. It is to be noted that Problem 2 holds for arbitrary number of players n, where the solution depends on n. Since the infinite-population solution is easier for analysis and may be viewed as a special case, one can generalize the homogeneous weights $\alpha_n^i = \frac{1}{n}$ to heterogeneous weights $\alpha_n^i = \frac{\beta^i}{n}$, where $\beta^i \in [-\beta_{\max}, \beta_{\max}]$, $\beta_{\max} \in \mathbb{R}_{>0}$, $i \in \mathbb{N}_n$. The resultant solution is called *sequential weighted mean-field equilibrium* (SWMFE) in [82]. The SWMFE constructs an approximate solution at any stage of the game $t_0 \in \mathbb{N}$ such that $J_{n,\gamma}^i(\mathbf{g}_\infty^{*,i}, \mathbf{g}_\infty^{*,-i})_{t_0} \le J_{n,\gamma}^i(\mathbf{g}^i, \mathbf{g}_\infty^{*,-i})_{t_0} + \varepsilon(n)$, where $\lim_{n\to\infty} \varepsilon(n) = 0$. For more details, see [55, Theorem 4].

5.2.1 Main challenges and contributions

There are several challenges to solve Problem 2. The first one is the *curse of dimensionality*, where the computational complexity of the solution increases with the number of players.

The second one is the *imperfect information* structure, where players do not have perfect information about the states of other players. The third challenge is that the resultant optimization problem is *non-convex* in the policy space, see a counterexample in [51]. The forth one lies in the fact that policy optimization is *not even locally convergent* in a game with continuous spaces, in general; see a counterexample in [53]. The main contribution of this chapter is to present an analytical proof for the global convergence of model-based and model-free policy gradient algorithms. In contrast to the model-based solution in [55] (whose number of unknowns increases quadratically with d_x), the number of unknown parameters in the proposed algorithms increases linearly with d_x and d_u. To the best of our knowledge, this is the first result on the global convergence of policy optimization in nonzero-sum LQ games.

5.3 Main Results

In this section, we first present a model-based algorithm introduced in [82] that requires $2d_x \times 2d_x$ parameters to construct the solution. Then, we propose two model-based gradient algorithms and prove their global convergence to the above solution, where their planning space is the policy space (that requires $2d_u \times d_x$ parameters to identify the solution). Based on the proposed gradient methods, we develop two model-free (reinforcement learning) algorithms and establish their global convergence to the model-based solution.

From [82], we use a gauge transformation to define the following variables for any player $i \in \mathbb{N}_n$ at any time $t \in \mathbb{N}$: $\mathbf{x}_t^i := \mathrm{vec}(x_t^i - \bar{x}_t, \bar{x}_t)$, $\mathbf{u}_t^i := \mathrm{vec}(u_t^i - \bar{u}_t, \bar{u}_t)$ and $\mathbf{w}_t^i := \mathrm{vec}(w_t^i - \bar{w}_t, \bar{w}_t)$, where $\bar{w}_t := \frac{1}{n}\sum_{i=1}^n w_t^i$. In addition, we define the following matrices: $\mathbf{A} := \mathrm{diag}(A, A + \bar{A})$, $\mathbf{B} := \mathrm{diag}(B, B + \bar{B})$, and

$$\mathbf{Q} := \begin{bmatrix} Q & Q + S^x \\ Q + S^x & Q + 2S^x + \bar{Q} \end{bmatrix}, \quad \mathbf{R} := \begin{bmatrix} R & R + S^u \\ R + S^u & R + 2S^u + \bar{R} \end{bmatrix}. \tag{50}$$

We now express the per-step cost of each player in (47) as:

$$c_t^i = (\mathbf{x}_t^i)^\intercal \mathbf{Q} \mathbf{x}_t^i + (\mathbf{u}_t^i)^\intercal \mathbf{R} \mathbf{u}_t^i. \tag{51}$$

To formulate the solution, we present a non-standard algebraic Riccati equation, introduced in [82], as follows:

$$\mathbf{M}(\boldsymbol{\theta}) = \mathbf{Q} + \boldsymbol{\theta}^\intercal \mathbf{R} \boldsymbol{\theta} + \gamma(\mathbf{A} - \mathbf{B}\boldsymbol{\theta})^\intercal \mathbf{M}(\boldsymbol{\theta})(\mathbf{A} - \mathbf{B}\boldsymbol{\theta}), \tag{52}$$

where $\boldsymbol{\theta} := \mathrm{diag}(\theta(n), \bar{\theta}(n))$, $\theta(n) := (F_n)^{-1} K_n$, $\bar{\theta}(n) := (\bar{F}_n)^{-1} \bar{K}_n$, and matrices F_n, $\bar{F}_n$, K_n and $\bar{K}_n$ are given by:

$$\begin{aligned}
F_n &= (1 - \frac{1}{n}) \Big[R + \gamma B^\intercal \mathbf{M}^{1,1}(\boldsymbol{\theta}) B \Big] + \frac{1}{n} \Big[R + S^u + \gamma (B + \bar{B})^\intercal \mathbf{M}^{1,2}(\boldsymbol{\theta}) B \Big], \\
\bar{F}_n &= (1 - \frac{1}{n}) \left[R + S^u + \gamma B^\intercal \mathbf{M}^{2,1}(\boldsymbol{\theta})(B + \bar{B}) \right] \\
&+ \frac{1}{n} \left[R + 2S^u + \bar{R} + \gamma (B + \bar{B})^\intercal \mathbf{M}^{2,2}(\boldsymbol{\theta})(B + \bar{B}) \right], \\
K_n &= (1 - \frac{1}{n}) \gamma B^\intercal \mathbf{M}^{1,1}(\boldsymbol{\theta}) A + \frac{\gamma}{n} (B + \bar{B})^\intercal \mathbf{M}^{1,2}(\boldsymbol{\theta}) A, \\
\bar{K}_n &= (1 - \frac{1}{n}) \gamma B^\intercal \mathbf{M}^{2,1}(\boldsymbol{\theta})(A + \bar{A}) + \frac{\gamma}{n} (B + \bar{B})^\intercal \mathbf{M}^{2,2}(\boldsymbol{\theta})(A + \bar{A}).
\end{aligned} \tag{53}$$

Assumption 1. *Suppose equations (52) and (53) admit a unique stable solution, which is also the limit of the finite-horizon solution. In addition, let F_n and $\bar{F}_n$ be invertible matrices, and $(1 - \frac{1}{n})F_n + \frac{1}{n}\bar{F}_n$ be a positive definite matrix.*

We now provide two sufficient conditions for Assumption 1 ensuring the existence of a stationary solution. Let G denote the mapping from $\mathbf{M}$ to $\boldsymbol{\theta}$ displayed in (53) (where $\boldsymbol{\theta} = G(\mathbf{M})$), and L denote the mapping from $\boldsymbol{\theta}$ to $\mathbf{M}$ expressed in (52) (where $\mathbf{M} = L(\boldsymbol{\theta})$). Thus, $\mathbf{M} = L(G(\mathbf{M}))$ is a fixed-point equation to be solved by fixed-point methods.

Assumption 2. *Let the mapping $L(G(\cdot))$ be a contraction, implying that equations (52) and (53) admit a unique fixed-point solution. In addition, let F_n and $\bar{F}_n$ be invertible matrices, and $(1 - \frac{1}{n})F_n + \frac{1}{n}\bar{F}_n$ be a positive definite matrix.*

Assumption 3 (Infinite-population decoupled Riccati equations)**.** Let Q and $Q+S^x$ be positive semi-definite, R and $R+S^u$ be positive definite, and $\bar{A}$ and $\bar{B}$ be zero. Suppose (A,B) is stabilizable, and $(A,Q^{1/2})$ and $(A,(Q+S^x)^{1/2})$ are detectable. When n is asymptotically large, the non-standard Riccati equation (52) decomposes into two decoupled standard Riccati equations; see [82, Proposition 2].

Theorem 3 (Model-based solution using non-standard Riccati equation [55])**.** *Let Assumption 1 hold. There exists a stationary subgame perfect Nash equilibrium such that for any player $i \in \mathbb{N}_n$ at any time $t \in \mathbb{N}_T$,*

$$u_t^{*,i} = -\theta^*(n)x_t^i - (\bar{\theta}^*(n) - \theta^*(n))\bar{x}_t, \tag{54}$$

where the gains are obtained from (53). In addition, the optimal cost of player $i \in \mathbb{N}_n$ from the initial time $t_0 = 1$ is given by: $J^i_{n,\gamma}(\boldsymbol{\theta}^*) = (1-\gamma)\,\mathrm{Tr}(\mathbf{M}(\boldsymbol{\theta}^*)\boldsymbol{\Sigma}^i_x) + \gamma\,\mathrm{Tr}(\mathbf{M}(\boldsymbol{\theta}^*)\boldsymbol{\Sigma}^i_w)$, where $\boldsymbol{\Sigma}^i_x := \mathbb{E}\big[(\mathrm{vec}(\Delta x^i_1), \bar{x}_1)(\mathrm{vec}(\Delta x^i_1), \bar{x}_1)^\intercal\big]$ and $\boldsymbol{\Sigma}^i_w := \mathbb{E}\big[\,\mathrm{vec}(\Delta w^i_t), \bar{w}_t)\,\mathrm{vec}(\Delta w^i_t), \bar{w}_t)^\intercal\big]$.

5.3.1 Model-based solution using policy optimization

From Theorem 3, there is no loss of optimality in restricting attention to linear identical stationary strategies of the form $\boldsymbol{\theta} = \mathrm{diag}(\theta, \bar{\theta})$. Therefore, we select one arbitrary player i as a learner and other players as imitators (that are passive during the learning process). More precisely, at each time instant, player i uses a gradient algorithm to update its strategy whereas other players employ the updated strategy to determine their next actions. In this chapter, we discard the process of selecting the learner, but in order to have a fair implementation, the learner may be chosen randomly at each iteration.[2] For simplicity of presentation, we omit the superscript i and the subscription of the cost function. Hence, the strategy of the learner can be described by: $\mathbf{u}_t = -\boldsymbol{\theta}\mathbf{x}_t, \mathbf{u}_t \in \mathbb{R}^{2d_u}, \mathbf{x}_t \in \mathbb{R}^{2d_x}, t \in \mathbb{N}$.

[2]For the special case of infinite population, it is also possible that all players become learners, i.e., they simultaneously learn the strategies as long as their exploration noises are i.i.d. In such a case, the infinite-population deep state reduces to weighted mean-field and remains unchanged.

Lemma 6. *The following holds at the initial time $t_0 = 1$:*

$$[\nabla_\theta J(\boldsymbol{\theta}), \nabla_{\bar{\theta}} J(\boldsymbol{\theta})] = 2\mathbf{P}_n \mathbf{E}_{\boldsymbol{\theta}} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}, \tag{55}$$

where

$$\begin{cases} \mathbf{P}_n := [(1-\frac{1}{n})\mathbf{I}_{d_u \times d_u}, \frac{1}{n}\mathbf{I}_{d_u \times d_u}], \\ \mathbf{E}_{\boldsymbol{\theta}} := (\mathbf{R} + \gamma \mathbf{B}^\intercal \mathbf{M}(\boldsymbol{\theta})\mathbf{B})\boldsymbol{\theta} - \gamma \mathbf{B}^\intercal \mathbf{M}(\boldsymbol{\theta})\mathbf{A}, \\ \boldsymbol{\Sigma}_{\boldsymbol{\theta}} := \mathbb{E}\big[(1-\gamma)\sum_{t=1}^{\infty} \gamma^{t-1} \mathbf{x}_t \mathbf{x}_t^\intercal\big]. \end{cases} \tag{56}$$

Proof. To compute the best-response of the learner, we fix the strategies of other players, and then find the gradient of the cost function with respect to θ and $\bar{\theta}$. Suppose player $i \in \mathbb{N}_n$ uses the strategy $u_t^i = \theta^i x_t^i + (\bar{\theta}^i - \theta^i)\bar{x}_t$. Therefore, one has:

$$\mathbf{u}_t^i = \begin{bmatrix} (1-\frac{1}{n})\theta^i & (1-\frac{1}{n})\bar{\theta}^i \\ \frac{1}{n}\theta^i & \frac{1}{n}\bar{\theta}^i \end{bmatrix} \mathbf{x}_t^i + \sum_{j \neq i} \begin{bmatrix} -\frac{1}{n}\theta^j & -\frac{1}{n}\bar{\theta}^j \\ \frac{1}{n}\theta^j & \frac{1}{n}\bar{\theta}^j \end{bmatrix} \mathbf{x}_t^j. \tag{57}$$

From (48) and (51), $J_{\mathbf{x}_1^i}(\boldsymbol{\theta}) = \mathbb{E}\big[(\mathbf{x}_1^i)^\intercal \mathbf{Q} \mathbf{x}_1^i + (\mathbf{u}_1^i)^\intercal \mathbf{R} \mathbf{u}_1^i\big] + \gamma J_{\mathbf{x}_2^i}(\boldsymbol{\theta}) = \mathbb{E}\big[(\mathbf{x}_1^i)^\intercal \mathbf{Q} \mathbf{x}_1^i + (\mathbf{u}_1^i)^\intercal \mathbf{R} \mathbf{u}_1^i\big] + \gamma \mathbb{E}\big[(\mathbf{x}_2^i)^\intercal \mathbf{M}(\boldsymbol{\theta}) \mathbf{x}_2^i\big]$. Taking the derivatives with respect to θ^i and $\bar{\theta}^i$, and then making $\theta^i = \theta^j = \theta$ and $\bar{\theta}^i = \bar{\theta}^j = \bar{\theta}$, leads to:

$$\begin{cases} \nabla_\theta J_{\mathbf{x}_1}(\boldsymbol{\theta}) = 2\Big((1-\frac{1}{n})(\mathbf{R}^{1,1} + \gamma \mathbf{B}^{1,1\intercal} \mathbf{M}^{1,1}(\boldsymbol{\theta}) \mathbf{B}^{1,1}) \\ + \frac{1}{n}(\mathbf{R}^{2,1} + \gamma \mathbf{B}^{2,1\intercal} \mathbf{M}^{2,1}(\boldsymbol{\theta}) \mathbf{B}^{2,1})\Big) \theta \mathbb{E}\big[\Delta x_1 \Delta x_1^\intercal\big] \\ + 2\Big((1-\frac{1}{n})(\mathbf{R}^{1,2} + \gamma \mathbf{B}^{1,2\intercal} \mathbf{M}^{1,2}(\boldsymbol{\theta}) \mathbf{B}^{1,2}) + \frac{1}{n}(\mathbf{R}^{2,2} \\ + \gamma \mathbf{B}^{2,2\intercal} \mathbf{M}^{2,2}(\boldsymbol{\theta}) \mathbf{B}^{2,2})\Big) \bar{\theta} \mathbb{E}\big[\bar{x}_1 \Delta x_1^\intercal\big] + \gamma \nabla_\theta J_{\mathbf{x}_2}(\boldsymbol{\theta}), \\ \nabla_{\bar{\theta}} J_{\mathbf{x}_1}(\boldsymbol{\theta}) = 2\Big((1-\frac{1}{n})(\mathbf{R}^{1,1} + \gamma \mathbf{B}^{1,1\intercal} \mathbf{M}^{1,1}(\boldsymbol{\theta}) \mathbf{B}^{1,1}) \\ + \frac{1}{n}(\mathbf{R}^{2,1} + \gamma \mathbf{B}^{2,1\intercal} \mathbf{M}^{2,1}(\boldsymbol{\theta}) \mathbf{B}^{2,1})\Big) \theta \mathbb{E}\big[\Delta x_1 \bar{x}_1^\intercal\big] \\ + 2\Big((1-\frac{1}{n})(\mathbf{R}^{1,2} + \gamma \mathbf{B}^{1,2\intercal} \mathbf{M}^{1,2}(\boldsymbol{\theta}) \mathbf{B}^{1,2}) \\ + \frac{1}{n}(\mathbf{R}^{2,2} + \gamma \mathbf{B}^{2,2\intercal} \mathbf{M}^{2,2}(\boldsymbol{\theta}) \mathbf{B}^{2,2})\Big) \bar{\theta} \mathbb{E}\big[\bar{x}_1 \bar{x}_1^\intercal\big] + \gamma \nabla_{\bar{\theta}} J_{\mathbf{x}_2}(\boldsymbol{\theta}). \end{cases} \tag{58}$$

The the rest of the proof follows from the recursive application of (58) and equations (52), (53) and (56). □

In this chapter, we consider two gradient-based methods.

- **Policy gradient descent**:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \operatorname{diag}(\nabla_\theta J(\boldsymbol{\theta}_k), \nabla_{\bar{\theta}} J(\boldsymbol{\theta}_k)). \tag{59}$$

- **Natural policy gradient descent**:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \operatorname{diag}(\nabla_\theta J(\boldsymbol{\theta}_k), \nabla_{\bar{\theta}} J(\boldsymbol{\theta}_k)) \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}. \tag{60}$$

To prove our convergence results, we impose extra standard assumptions, described below.

Assumption 4. *The initial policy is stable. A policy $\boldsymbol{\theta}$ is said to be stable if $\rho(\mathbf{A} - \mathbf{B}\boldsymbol{\theta}) < 1$.*

Assumption 5. *Given the learner, $\mathbb{E}\big[\mathbf{x}_1(\mathbf{x}_1)^\intercal\big]$ is positive definite. For the special case of i.i.d. initial states, $\mathbb{E}\big[\mathbf{x}_1^i(\mathbf{x}_1^i)^\intercal\big] = \operatorname{diag}((1-\frac{1}{n})cov(x_1), \frac{1}{n}cov(x_1) + \mathbb{E}\big[x_1\big]\mathbb{E}\big[x_1\big]^\intercal)$ is positive definite if $cov(x_1^i) =: cov(x_1)$ and $\mathbb{E}\big[x_1^i\big]\mathbb{E}\big[x_1^i\big]^\intercal =: \mathbb{E}\big[x_1\big]\mathbb{E}\big[x_1\big]^\intercal$, $i \in \mathbb{N}_n$, are positive definite.*

Assumption 6. *For finite-population model, $\mathbf{Q}$ and $\mathbf{R}$ are positive definite matrices. For the infinite-population case satisfying Assumption 3, Q and $Q + S$ are positive definite.*

Assumptions 4–6 are standard conditions in the literature of LQ reinforcement learning [51, 69], which ensure that for any stable $\boldsymbol{\theta}$, $J(\boldsymbol{\theta})$ is properly bounded and $\boldsymbol{\Sigma}_{\boldsymbol{\theta}} \succcurlyeq \mathbb{E}\big[\mathbf{x}_1(\mathbf{x}_1)^\intercal\big]$ is positive definite. We now show that the best-response optimization at the learner satisfies the Polyak-Lojasiewicz (PL) condition [70, 71], which is a relaxation of the notion of strong convexity. Let $\mu := \sigma_{min}(\mathbb{E}\big[\mathbf{x}_1\mathbf{x}_1^\intercal\big])$.

Lemma 7 (PL condition)**.** *Let Assumptions 1, 4, 5 and 6 hold. Let also $\boldsymbol{\theta}^*$ be the Nash policy in Theorem 3. There exists a positive constant $L_1(\boldsymbol{\theta}^*)$ such that*

$$J(\boldsymbol{\theta}) - J(\boldsymbol{\theta}^*) \leq L_1(\boldsymbol{\theta}^*) \|[\nabla_\theta J(\boldsymbol{\theta}), \nabla_{\bar{\theta}} J(\boldsymbol{\theta})]\|_F^2, \tag{61}$$

where $L_1(\boldsymbol{\theta}^*) = \frac{n^2\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|}{4\mu^2\sigma_{min}(\mathbf{R})}$. *For the special case of infinite population (i.e.* $n = \infty$*) with i.i.d. initial states under Assumptions 3, 4, 5 and 6, one has*

$$L_1(\boldsymbol{\theta}^*) = \frac{\|\boldsymbol{\Sigma}^{1,1}_{\boldsymbol{\theta}^*}\|}{4\sigma_{min}(cov(x_1))^2\sigma_{min}(R)} + \frac{\|\boldsymbol{\Sigma}^{2,2}_{\boldsymbol{\theta}^*}\|}{4\sigma_{min}(\mathbb{E}[x_1]\mathbb{E}[x_1]^\intercal)^2\sigma_{min}(R+S^u)}.$$

Proof. Let $\tilde{\mathbf{P}}_n := \text{diag}((1-\frac{1}{n})\mathbf{I}_{d_u\times d_u}, \frac{1}{n}\mathbf{I}_{d_u\times d_u})$. We express (55) in terms of the square matrix $\tilde{\mathbf{P}}_n$ such that $\Delta J_\theta(\boldsymbol{\theta}) =: \Delta J^1_\theta(\boldsymbol{\theta}) + \Delta J^2_\theta(\boldsymbol{\theta})$ and $\Delta J_{\bar{\theta}}(\boldsymbol{\theta}) =: \Delta J^1_{\bar{\theta}}(\boldsymbol{\theta}) + \Delta J^2_{\bar{\theta}}(\boldsymbol{\theta})$, where

$$\nabla_{\boldsymbol{\theta}}\tilde{J} := \begin{bmatrix} \Delta J^1_\theta(\boldsymbol{\theta}) & \Delta J_{\bar{\theta}}(\boldsymbol{\theta}) \\ \Delta J^2_\theta(\boldsymbol{\theta}) & \Delta J_{\bar{\theta}}(\boldsymbol{\theta}) \end{bmatrix} = 2\tilde{\mathbf{P}}_n\mathbf{E}_{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}. \tag{62}$$

Following [**?**, Lemma 10] and after some algebraic manipulations, we can derive the following inequality for sequences $\{\mathbf{x}^*_t\}_{t=1}^\infty$ and $\{\mathbf{u}^*_t\}_{t=1}^\infty$ generated by the Nash policy $\boldsymbol{\theta}^*$. In particular, from (55), $\Sigma_{\boldsymbol{\theta}} \succcurlyeq \mathbb{E}\big[(\mathbf{x}_1\mathbf{x}_1^\intercal)\big]$, and the fact that $\tilde{\mathbf{P}}_n$ is positive definite for any finite n, it results that:

$$\begin{aligned} J(\boldsymbol{\theta}) - J(\boldsymbol{\theta}^*) &\leq (1-\gamma)\mathbb{E}\sum_{t=1}^{\infty}\gamma^{t-1}\,\text{Tr}(\mathbf{x}^*_t\mathbf{x}^{*\intercal}_t\mathbf{E}^\intercal_{\boldsymbol{\theta}}(\mathbf{R}+\gamma\mathbf{B}^\intercal\mathbf{M}_{\boldsymbol{\theta}}\mathbf{B})^{-1}\mathbf{E}_{\boldsymbol{\theta}}) \\ &= \text{Tr}(\Sigma_{\boldsymbol{\theta}^*}\mathbf{E}^\intercal_{\boldsymbol{\theta}}(\mathbf{R}+\gamma\mathbf{B}^\intercal\mathbf{M}_{\boldsymbol{\theta}}\mathbf{B})^{-1}\mathbf{E}_{\boldsymbol{\theta}}) \leq \frac{\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|}{\sigma_{\min}(\mathbf{R})}\,\text{Tr}(\mathbf{E}^\intercal_{\boldsymbol{\theta}}\mathbf{E}_{\boldsymbol{\theta}}) \\ &= \frac{\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|}{4\sigma_{\min}(\mathbf{R})}\,\text{Tr}(\boldsymbol{\Sigma}_{\boldsymbol{\theta}}{}^{-1}\nabla_{\boldsymbol{\theta}}\tilde{J}^\intercal\mathbf{P}_n^{-2}\nabla_{\boldsymbol{\theta}}\tilde{J}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}{}^{-1}) \\ &\leq \frac{0.25\mu^{-2}\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|}{\sigma_{\min}(\tilde{\mathbf{P}}_n)^2\sigma_{\min}(\mathbf{R})}\|[\nabla_\theta J(\boldsymbol{\theta}), \nabla_{\bar{\theta}}J(\boldsymbol{\theta})]\|^2_F. \end{aligned}$$

For $n = \infty$, $\tilde{\mathbf{P}}_n$ is not invertible; however, equation (52) under Assumption 3 decomposes into two *decoupled* standard Riccati equations with matrices (A, B, Q, R) and $(A, B, Q + S^x, R + S^u)$. By following the approach proposed in [51, Lemma 11], it is straightforward to show that the cost difference in this case is upper bounded by:

$$\frac{\|\boldsymbol{\Sigma}^{1,1}_{\boldsymbol{\theta}^*}\|}{4\sigma_{min}(\text{cov}(x_1))^2\sigma_{\min}(R)}\|\nabla_\theta J(\boldsymbol{\theta})\|^2_F + \frac{\|\boldsymbol{\Sigma}^{2,2}_{\boldsymbol{\theta}^*}\|}{4\sigma_{min}(\mathbb{E}[x_1]\mathbb{E}[x_1]^\intercal)^2\sigma_{\min}(R+S^u)}\|\nabla_{\bar{\theta}}J(\boldsymbol{\theta})\|^2_F.$$

□

In the following lemmas, we show that the cost function and its gradient are locally Lipschitz functions.

Lemma 8 (Locally Lipschitz cost function). *For any $\boldsymbol{\theta}'$ satisfying the inequality $\|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_F < \varepsilon(\boldsymbol{\theta})$, there exists a positive constant $L_2(\boldsymbol{\theta})$ such that $|J(\boldsymbol{\theta}') - J(\boldsymbol{\theta})| \leq L_2(\boldsymbol{\theta})\|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_F$, where the explicit expressions of $\varepsilon(\boldsymbol{\theta})$ and $L_2(\boldsymbol{\theta})$ can be obtained in a similar manner as [69, Lemma 15].*

Proof. Using the extended form of the gains $\boldsymbol{\theta}$, the form of the solution can be reduced to a single-agent setting. Then, the proof follows like that of [51, Lemma 15] . □

Lemma 9 (Locally Lipschitz gradient). *For any $\boldsymbol{\theta}'$ satisfying the inequality $\|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_F < \varepsilon(\boldsymbol{\theta})$, there exists a positive constant $L_3(\boldsymbol{\theta})$ such that*

$$\|[\nabla J_\theta(\boldsymbol{\theta}'), \nabla J_{\bar{\theta}}(\boldsymbol{\theta}')] - [\nabla J_\theta(\boldsymbol{\theta}), \nabla J_{\bar{\theta}}(\boldsymbol{\theta})]\|_F \leq L_3(\boldsymbol{\theta})\|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_F, \tag{63}$$

where the explicit expressions of $\varepsilon(\boldsymbol{\theta})$ and $L_3(\boldsymbol{\theta})$ can be obtained in a similar manner as [69, Lemma 16].

Proof. Using the extended form of the gains $\boldsymbol{\theta}$, the form of the solution can be reduced to a single-agent setting. Then, the proof follows like that of [51, Lemma 16] . □

Theorem 4 (Global convergence via model-based gradient). *Let Assumptions 1, 4, 5 and 6 hold. For a sufficiently small fixed step size η chosen as $\eta = \text{poly}\left(\frac{\mu\sigma_{min}(\mathbf{Q})}{J(\boldsymbol{\theta}_1)}, \frac{1}{\sqrt{\gamma}\|\mathbf{A}\|}, \frac{1}{\sqrt{\gamma}\|\mathbf{B}\|}, \frac{1}{\|\mathbf{R}\|}, \sigma_{min}(\mathbf{R})\right)$, and for a sufficiently large number of iterations K such that*

$$K \geq \frac{\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|}{\mu} \log \frac{J(\boldsymbol{\theta}_1) - J(\boldsymbol{\theta}^*)}{\varepsilon} \text{poly}\left(\frac{J(\boldsymbol{\theta}_1)}{\mu\sigma_{min}(\mathbf{Q})}, \sqrt{\gamma}\|\mathbf{A}\|, \sqrt{\gamma}\|\mathbf{B}\|, \|\mathbf{R}\|, \frac{1}{\sigma_{min}(\mathbf{R})}\right),$$

the gradient descent algorithm (59) leads to the following bound: $J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*) \leq \varepsilon$. In particular, for a fixed step size $\eta = \frac{1}{\|\mathbf{P}_n^\intercal \mathbf{P}_n\|(\|\mathbf{R}\| + \frac{\gamma\|\mathbf{B}\|^2 J(\boldsymbol{\theta}_1)}{\mu})}$ and for a sufficiently large number of iterations K, i.e.,

$$K \geq \frac{\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|\|\mathbf{P}_n^\intercal \mathbf{P}_n\|}{\mu}\left(\frac{\|\mathbf{R}\|}{\sigma_{min}(\mathbf{R})} + \frac{\gamma\|\mathbf{B}\|^2 J(\boldsymbol{\theta}_1)}{\mu\sigma_{min}(\mathbf{R})}\right) \log \frac{J(\boldsymbol{\theta}_1) - J(\boldsymbol{\theta}^*)}{\varepsilon},$$

the natural policy gradient descent algorithm (60) enjoys the bound: $J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*) \leq \varepsilon$.

Proof. Following the proof technique in [51, Theorem 7], we choose a sufficiently small step size η such that the value of the cost decreases at each iteration. More precisely, for the natural policy gradient descent at iteration K,

$$J(\boldsymbol{\theta}_{K+1}) - J(\boldsymbol{\theta}^*) \leq (1 - \frac{\mu\sigma_{\min}(\mathbf{R})}{\|\mathbf{P}_n^\intercal \mathbf{P}_n\|(\|\mathbf{R}\| + \frac{\gamma\|\mathbf{B}\|^2 J(\boldsymbol{\theta}_1)}{\mu})\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|})(J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*))$$
$$= (1 - \eta\frac{\mu\sigma_{\min}(\mathbf{R})}{\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}^*}\|})(J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*)).$$

The above recursion is contractive for the specified η. □

5.3.2 Model-free solution using policy optimization

It is desired now to develop a model-free RL algorithm.

Lemma 10 (Finite-horizon approximation)**.** *For any $\boldsymbol{\theta}$ with finite $J(\boldsymbol{\theta})$, define $\tilde{J}_T(\boldsymbol{\theta}) := (1 - \gamma)\mathbb{E}[\sum_{t=1}^T \gamma^{t-1} c_t]$ and $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}} = (1-\gamma)\mathbb{E}[\sum_{t=1}^T \gamma^{t-1}\mathbf{x}_t\mathbf{x}_t^\intercal]$. Let $\varepsilon(T) := \frac{d_x(J(\boldsymbol{\theta}))^2}{(1-\gamma)T\mu\sigma_{min}^2(\mathbf{Q})}$ and $\bar{\varepsilon}(T) := \varepsilon(T)(\|\mathbf{Q}\| + \|\mathbf{R}\|\|\boldsymbol{\theta}\|^2)$, then $\|\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}} - \boldsymbol{\Sigma}_{\boldsymbol{\theta}}\| \leq \varepsilon(T)$ and $|\tilde{J}_T(\boldsymbol{\theta}) - J(\boldsymbol{\theta})| \leq \bar{\varepsilon}(T)$.*

Proof. The proof is omitted due to space limitation. □

Let $\mathbb{S}_r$ be a set of uniformly distributed points with norm $r > 0$ (e.g., the surface of a sphere). In addition, let $\mathbb{B}_r$ denote the set of all uniformly distributed points whose norms are at most r (e.g., all points within the sphere). For a matrix $\tilde{\boldsymbol{\theta}} = \mathrm{diag}(\tilde{\theta}, \tilde{\tilde{\theta}})$, these distributions are defined over the Frobenius norm ball. Hence, $J_r(\boldsymbol{\theta}) = \mathbb{E}_{\tilde{\boldsymbol{\theta}} \sim \mathbb{B}_r}[J(\boldsymbol{\theta} + \tilde{\boldsymbol{\theta}})]$. Since the expectation can be expressed as an integral function, one can use Stokes' formula to compute the gradient of $J_r(\boldsymbol{\theta})$ with only query access to the function values.

Lemma 11 (Zeroth-order optimization)**.** *For a smoothing factor $r > 0$, $[\nabla_\theta J(\boldsymbol{\theta}), \nabla_{\tilde{\theta}} J(\boldsymbol{\theta})] = \frac{2d_x d_u}{r^2}\mathbb{E}_{\tilde{\boldsymbol{\theta}} \sim \mathbb{S}_r}[J(\boldsymbol{\theta} + \tilde{\boldsymbol{\theta}})[\tilde{\theta}, \tilde{\tilde{\theta}}]]$.*

Proof. The proof follows directly from the zeroth-order optimization approach [72, Lemma 1]. □

Lemma 12. *Let $\tilde{\boldsymbol{\theta}}_1, \ldots, \tilde{\boldsymbol{\theta}}_L$, $L \in \mathbb{N}$, be i.i.d. samples drawn uniformly from $\mathbb{S}_r$. There exists $\varepsilon(L) := \mathrm{poly}(1/L) > 0$, such that $[\tilde{\nabla}_\theta^L J(\boldsymbol{\theta}), \tilde{\nabla}_{\tilde{\theta}}^L J(\boldsymbol{\theta})] = \frac{2d_x d_u}{r^2 L}\sum_{l=1}^L J(\boldsymbol{\theta} + \tilde{\boldsymbol{\theta}}_l)[\tilde{\theta}, \tilde{\tilde{\theta}}]$ converges*

to $[\nabla_\theta J(\boldsymbol{\theta}), \nabla_{\bar{\theta}} J(\boldsymbol{\theta})]$ *in the Frobenius norm with a probability greater than* $1 - (\frac{2d_x d_u}{\varepsilon(L)})^{-2d_x d_u}$. *From Lemma 10, there exists* $\varepsilon(L,T) := \text{poly}(1/L, 1/T) > 0$ *such that* $[\tilde{\nabla}_\theta^{L,T} J(\boldsymbol{\theta}), \tilde{\nabla}_{\bar{\theta}}^{L,T} J(\boldsymbol{\theta})] = \frac{2d_x d_u (1-\gamma)}{r^2 L} \sum_{l=1}^{L} [\sum_{t=1}^{T} \gamma^{t-1}(c_t)][\tilde{\theta}_l, \tilde{\bar{\theta}}_l]$ *is* $\varepsilon(L,T)$ *close to* $[\nabla_\theta J(\boldsymbol{\theta}), \nabla_{\bar{\theta}} J(\boldsymbol{\theta})]$ *with a probability greater than* $1 - (\frac{2d_x d_u}{\varepsilon(L,T)})^{-2d_x d_u}$ *in the Frobenius norm.*

Theorem 5 (Global convergence via model-free gradient)**.** *Let Assumptions 1, 4, 5 and 6 hold. For a sufficiently large horizon* T *and samples* L*, model-free gradient descent and natural policy gradient decent with the empirical gradient in Lemma 12 and covariance matrix in Lemma 10 converge to the model-based solutions in Theorem 4. In particular, the gradient descent algorithm converges with a probability greater than* $1 - (\frac{2d_x d_u}{\varepsilon(L,T)})^{-2d_x d_u}$*, where* $\varepsilon(L,T) = \text{poly}(1/L, 1/T)$.

Proof. From [51, Theorem 31] and Theorem 4, one has the following inequality at iteration $K \in \mathbb{N}$ for a sufficiently small step size $\eta \leq \eta_{max}$,

$$J(\boldsymbol{\theta}_{K+1}) - J(\boldsymbol{\theta}^*) \leq (1 - \eta\eta_{max}^{-1})(J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*)).$$

At iteration K, denote by $\tilde{\nabla}_K$ the empirical gradient and by $\hat{\boldsymbol{\theta}}_{K+1} = \boldsymbol{\theta}_K - \eta\tilde{\nabla}_K$ the update with the empirical gradient. From Lemma 8, $|J(\hat{\boldsymbol{\theta}}_{K+1}) - J(\boldsymbol{\theta}_{K+1})| \leq \frac{1}{2}\eta\eta_{max}^{-1}\varepsilon(L,T)$, when $\|\hat{\boldsymbol{\theta}}_{K+1} - \boldsymbol{\theta}_{K+1}\| \leq \frac{1}{2}\eta\eta_{max}^{-1}\varepsilon(L,T)(1/L_2(\boldsymbol{\theta}_{K+1}))$, upon noting that $\hat{\boldsymbol{\theta}}_{K+1} - \boldsymbol{\theta}_{K+1} = \eta(\nabla_K - \tilde{\nabla}_K)$ and $\|\nabla_K - \tilde{\nabla}_K\| \leq \frac{1}{2}\eta_{max}^{-1}\varepsilon(L,T)(1/L_2(\boldsymbol{\theta}_{K+1}))$. According to the Bernstein inequality, the above inequality holds with a probability greater than $1 - (\frac{2d_x d_u}{\varepsilon(L,T)})^{-2d_x d_u}$. Therefore, from Lemmas 9 and 12, the distance between the empirical gradient and the exact one monotonically decreases as the number of samples and rollouts increases, provided that the smoothing factor r is sufficiently small. Consequently, one arrives at

$$J(\hat{\boldsymbol{\theta}}_{K+1}) - J(\boldsymbol{\theta}^*) \leq (1 - \frac{1}{2}\eta\eta_{max}^{-1})(J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*)),$$

when $J(\boldsymbol{\theta}_K) - J(\boldsymbol{\theta}^*) \leq \varepsilon(L,T)$. This recursion is contractive; i.e., the rest of the proof will be similar to that of Theorem 4. □

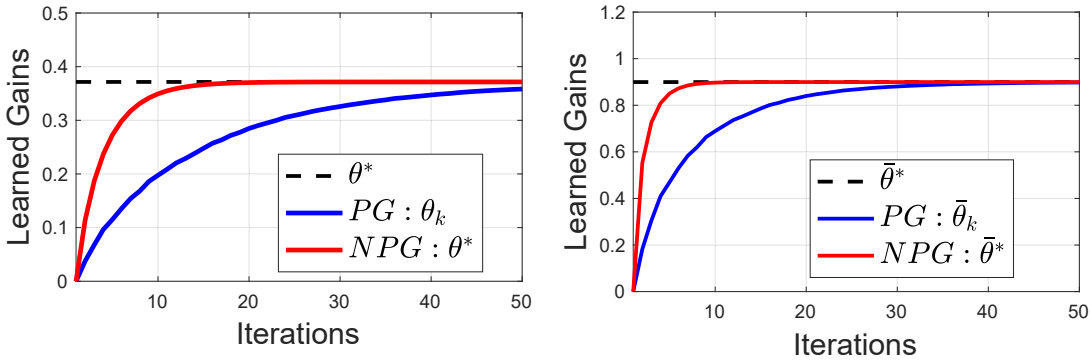

Figure 5.1: Convergence of the model-based gradient descent and natural policy gradient descent algorithms in Example 1.

5.4 Simulations

In this section, simulations are conducted to demonstrate the global convergence of the proposed gradient methods. To compute the Nash policy, plotted in dashed lines in the figures, we use the solution of equation (52).

Example 1. Consider a dynamic game with the following parameters: $\eta = 0.1, n = 100$, $T = 100$, $L = 3$, $A = 0.7, B = 0.4, \bar{A} = 0, \bar{B} = 0, Q = 1, R = 1, S^x = 4, S^u = 0, \bar{Q} = 0$, $\bar{R} = 0$, $\Sigma_x = 1$ and $\Sigma_w = 0.4$. It is observed in Figure 5.1 that natural policy gradient descent reaches the Nash strategy faster than the gradient descent.

Example 2. Let the system parameters be $\eta = 0.04$, $T = 10$, $r = 0.09$, $L = 1500$, $A = 1, B = 0.5, \bar{A} = 0, \bar{B} = 0, Q = 1, R = 1, S^x = 2, S^u = 0, \bar{Q} = 1$, $\bar{R} = 0$, $r = 0.09$, $\Sigma_x = 0.05$ and $\Sigma_w = 0.01$. The model-free policy gradient algorithm was run on a 2.7 GHz Intel Core i5 processor for 10 random seeds. After 6000 iterations, which took roughly 10 hours, both θ and $\bar{\theta}$ reached their optimal values as depicted in Figure 5.2.

Example 3. In this example, let the system parameters be $\eta = 0.1$, $T = 100$, $L = 3$, $A = 0.8, B = 0.2, \bar{A} = 0, \bar{B} = 0, Q = 1, R = 1, S^x = 2, S^u = 0, \bar{Q} = 4$, $\bar{R} = 0$, $\Sigma_x = 1$ and $\Sigma_w = 0.1$. To investigate the effect of the number of players, we considered five different values for $n \in \{2, 5, 10, 20, 100\}$. It is shown in Figure 5.3 that the policies converge to a limit as the number of players increases, which is known as the mean-field limit.

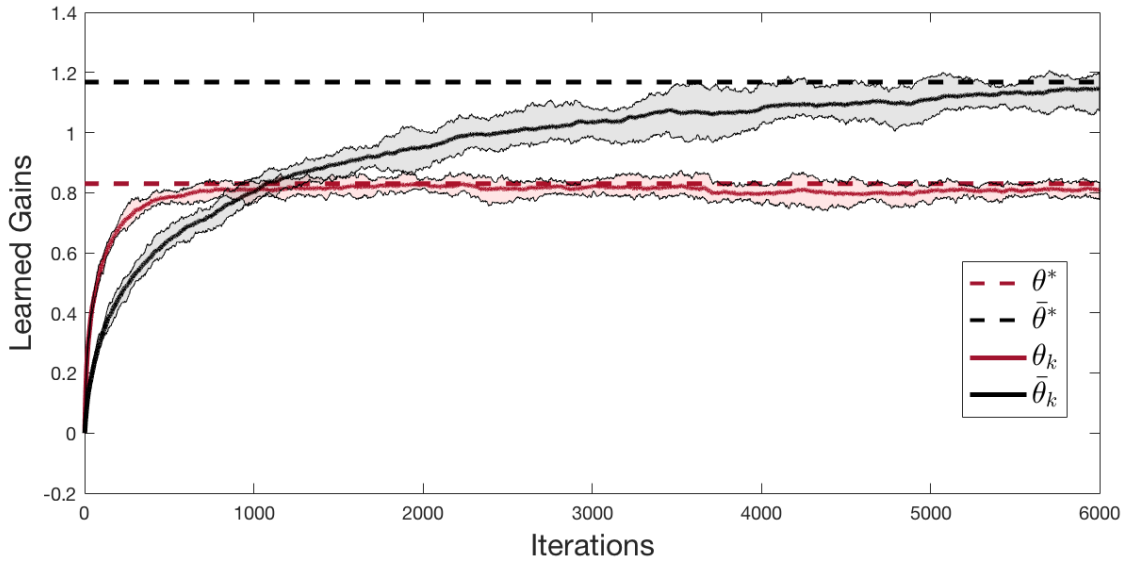

Figure 5.2: Convergence of the proposed model-free algorithm in Example 2.

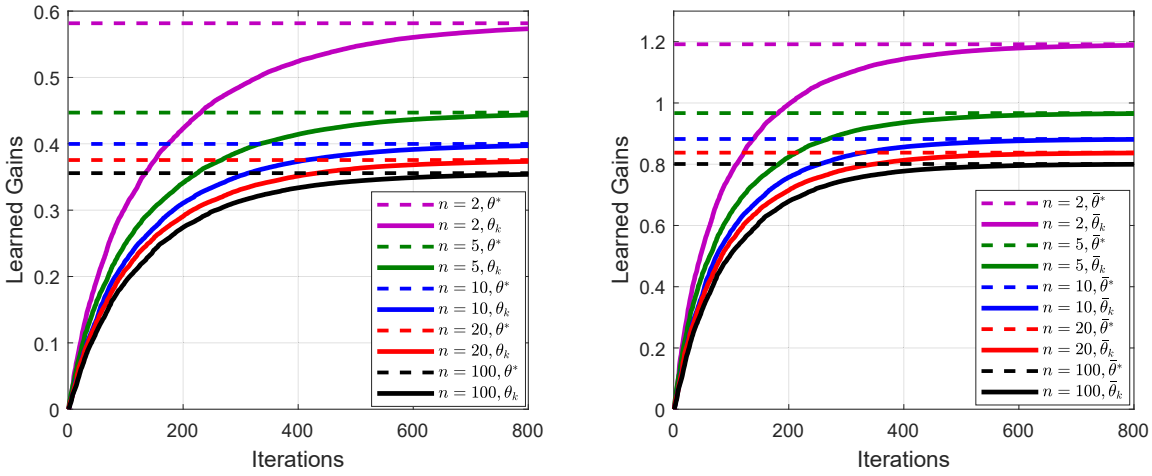

Figure 5.3: The effect of the number of players on the policy in Example 3.

5.5 Conclusions

In this chapter, we investigated model-based and model-free gradient descent and natural policy gradient descent algorithms for LQ deep structured games with homogeneous weights. It was shown theoretically and verified by simulations, that the gradient-based methods enjoy the global convergence to the sequential Nash solution. One of the main features of the proposed solutions is that their planning space is independent of the number

of players. The obtained results naturally extend to asymptotically vanishing weights and other variants of policy gradient algorithms such as REINFORCE and actor-critic methods.

Chapter 6

Risk-Constrained Control of Mean-Field Linear Quadratic Systems

The risk-neutral LQR controller is optimal for stochastic linear dynamical systems. However, the classical optimal controller performs inefficiently in the presence of low-probability yet statistically significant (risky) events. The present research focuses on infinite-horizon risk-constrained linear quadratic regulators in a mean-field setting. We address the risk constraint by bounding the cumulative one-stage variance of the state penalty of all players. It is shown that the optimal controller is affine in the state of each player with an additive term that controls the risk constraint. In addition, we propose a solution independent of the number of players. Finally, simulations are presented to apply the results to a microgrid system to verify the theoretical findings.

6.1 Introduction

The performance evaluation of dynamical systems in the optimal control framework has long been studied in the literature. Specifically, in the linear quadratic regulator (LQR) with noisy inputs, the focus is on minimizing the expected cumulative time-average quadratic cost, also known as a risk-neutral setting [73]. However, such a risk-neutral framework often exhibits unsatisfactory performance in real-world control systems. For instance, there

exists a rich body of research to address risk in different areas, including robotics [74,75], financial systems [76,77], power grids [78,79], and multi-agent networks [58,80]. Moreover, neglecting the effect of low-probability severe external events may lead to catastrophic consequences in dynamic systems, like crashing in a flock of UAVs or an autonomous vehicle hitting other vehicles and pedestrians.

There has been an increasing interest in the research community recently in the risk assessment of dynamical systems by deriving closed-form solutions for a single-agent setting [83, 84]. Specifically, by solving a set of Riccati and fixed-point equations, one can obtain an affine form of the policy to meet the system's constraints. However, in the control of a large number of agents, such a method may not provide sufficient efficacy.

This research considers the problem of exchangeable agents (players) in a mean-field setting. In such a setting, all agents have similar dynamics, and the players' states evolve as a linear function of their previous states and the overall average state. Using the results in mean-field theory, we show that the required Riccati equation (whose size increases with the number of players) can be decomposed into two Riccati equations with the same dimension as the agents' states. Furthermore, we propose a primal-dual algorithm to solve the problem iteratively.

The rest of this chapter is organized as follows. In Section II, we present some preliminaries and formulate the problem. The solution to the optimization problem is derived in Section III, followed by simulations to validate the results in Section IV. Finally, some concluding remarks and directions for future research are given in Section V.

6.2 Problem Formulation

Throughout this chapter, $\mathbb{R}$, $\mathbb{R}_{>0}$ and $\mathbb{N}$ represent the sets of real, positive real and natural numbers, respectively. Given any $n \in \mathbb{N}$, $\mathbb{N}_n$, and $\mathbf{I}_{n\times n}$ denote the finite set $\{1, \ldots, n\}$, and the $n \times n$ identity matrix, respectively. $\|\cdot\|$ is the spectral norm of a matrix, $\mathrm{Tr}(\cdot)$ is the trace of a matrix, $\tau_{\min}(\cdot)$ is the minimum singular value of a matrix, $\rho(\cdot)$ is the spectral radius of a matrix, and $\mathrm{diag}(\Lambda_1, \Lambda_2)$ is the block diagonal matrix $[\Lambda_1 \quad 0; 0 \quad \Lambda_2]$, and $\mathrm{diag}(\Lambda)_{i=1}^{k}$

denotes a bloack-diagonal matrix with k times repetition of the matrix Λ. For vectors x, y and z, vec$(x, y, z) = [x^\intercal, y^\intercal, z^\intercal]^\intercal$ is a column vector, $x_{1:t}$ denotes the vector $(x_1, ..., x_t)$ and the operator $\otimes$ denotes the Kronecker product between two matrices of appropriate size. Also, the rectified linear function is denoted by the operator $[x]_+ = \max\{0, x\}$.

6.2.1 General Form of the Problem

Given $n \in \mathbb{N}$ players, let $x_t^i \in \mathbb{R}^{d_x}$, $u_t^i \in \mathbb{R}^{d_u}$ and $w_t^i \in \mathbb{R}^{d_x}$ denote, respectively, the state, action and local noise of player $i \in \mathbb{N}_n$ at time $t \in \mathbb{N}$, where $d_x, d_u \in \mathbb{N}$. Define the mean-state of the players as $\bar{x}_t := \frac{1}{n}\sum_{i=1}^n x_t^i$. The initial states $\{x_0^1, \dots, x_0^n\}$ are random with finite covariance matrices. The evolution of the state of any player $i \in \mathbb{N}_n$ at time $t \in \mathbb{N}$ is given by:

$$x_{t+1}^i = Ax_t^i + Bu_t^i + \bar{A}\bar{x}_t + \bar{B}\bar{u}_t + w_t^i, \tag{64}$$

where $\{w_t^i\}_{t=0}^\infty$ is an independent and identically distributed (i.i.d.) zero-mean noise process with a finite covariance matrix.

The per-step cost of all players at time $t \in \mathbb{N}$ is given by:

$$c_t = (\bar{x}_t)^\intercal \bar{Q}\bar{x}_t + (\bar{u}_t)^\intercal \bar{R}\bar{u}_t + \frac{1}{n}\sum_{i=1}^n (x_t^i)^\intercal Q x_t^i + (u_t^i)^\intercal R u_t^i, \tag{65}$$

where Q, $\bar{Q}$, R, and $\bar{R}$ are symmetric matrices with appropriate dimensions.

Definition 3. *Let $h_t^i = \{x_0^i, u_0^i, ..., x_{t-1}^i, u_{t-1}^i, x_t^i\}$ denote the history trajectory of player $i \in \mathbb{N}_n$. Then, the per-step risk factor for the ith player is defined as*

$$d_t^i = \Big((x_t^i)^\intercal Q x_t^i - \mathbb{E}\big[(\mathbf{x}_t^i)^\intercal Q \mathbf{x}_t^i | h_t^i\big]\Big)^2. \tag{66}$$

Assumption 7. *It is assumed hereafter that the pair (A, B) is stabilizable, the pair $(A, Q^{\frac{1}{2}})$ is detectable, and matrices Q and R are positive semi-definite and positive definite, respectively.*

Assumption 8. *The local noises $w_t^1, ..., w_t^n$ have the same distribution.*

Assumption 9. *The noise w_t^i for every player $i \in \mathbb{N}_n$ has a finite fourth-order moment, i.e.,*

$\mathbb{E}\|w_t^i\|^4 < \infty$.

In this chapter, we consider the infinite-horizon risk-constrained LQR for a team of cooperative players to minimize a common cost. Also, it is desired to constrain the cumulative per-step risk of all players. This leads to the following constrained optimization problem

$$\text{minimize} \quad J = \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T} c_t\right] \tag{67a}$$

$$\text{s.t.} \quad (64) \text{ and} \tag{67b}$$

$$J_c = \frac{1}{n}\sum_{i=1}^{n} \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T} d_t^i\right] \leq \Gamma, \quad \forall i \in \mathbb{N}_n, \tag{67c}$$

where $\Gamma > 0$ is a predefined risk tolerance of the user.

Remark 14. *From [81, 82], when player $i \in \mathbb{N}_n$ at any time $t \in \mathbb{N}$ observes its local state x_t^i and the mean state $\bar{x}_t$, i.e. $\{x_{1:t}^i, \bar{x}_{1:t}\}$, an information structure called deep state sharing (DSS) is considered.*

Definition 4. *Let the control input of player $i \in \mathbb{N}_n$ at time t be denoted by $u_t^i = \phi_t^i(x_{1:t}^i, \bar{x}_{1:t})$. Define $\Phi^i := \{\phi_t^i\}_{t=1}^{\infty}$ and $\mathbf{\Phi}_n := \{\mathbf{\Phi}^1, \dots, \mathbf{\Phi}^n\}$ as the control strategy of player i and that of all players, respectively.*

We now present the main problem of this chapter.

Problem 3. *Consider the risk-constrained mean-field LQR problem in (67). Given the system dynamics (64), find an optimal control strategy $\mathbf{\Phi}^*$ such that for any arbitrary control law $\mathbf{\Phi}$, the cost function (67a) under the constraints (67b) and (67c) satisfies the following inequality*

$$J(\mathbf{\Phi}^*) \leq J(\mathbf{\Phi}).$$

6.3 Main Results

In this section, we propose a step by step solution to the optimization problem (67).

6.3.1 Problem Reformulation

Define a new transformed state $\tilde{x}_t^i = x_t^i - \bar{x}_t$ for player $i \in \mathbb{N}_n$. Define also the mean control input of all players as $\bar{u}_t := \frac{1}{n}\sum_{i=1}^n u_t^i$, and the transformed control input of player $i \in \mathbb{N}_n$ as $\tilde{u}_t^i = u_t^i - \bar{u}_t$. It follows from [82] that

$$\begin{aligned} \tilde{x}_{t+1}^i &= A\tilde{x}_t^i + B\tilde{u}_t^i + \tilde{w}_t^i \\ \bar{x}_{t+1} &= \mathcal{A}\bar{x}_t + \mathcal{B}\bar{u}_t + \bar{w}_t, \end{aligned} \tag{68}$$

where $\mathcal{A} = A + \bar{A}$, $\mathcal{B} = B + \bar{B}$, $\bar{w}_t := \frac{1}{n}\sum_{i=1}^n w_t^i$ and $\tilde{w}_t^i = w_t^i - \bar{w}_t$.

Next, define the first and second-order moments (mean and covariance) of each player's local noise as $m_1 = \mathbb{E}[w_t^i]$ and $M_2 = \mathbb{E}[(w_t^i - m_1^i)(w_t^i - m_1^i)^\intercal]$, respectively. Furthermore, let the next two higher order moments of the local noise be defined as

$$\begin{aligned} M_3 &= \mathbb{E}[(w_t^i - m_1^i)(w_t^i - m_1^i)^\intercal Q(w_t^i - m_1^i)], \\ M_4 &= \mathbb{E}[(w_t^i - m_1^i)^\intercal Q(w_t^i - m_1^i) - \mathrm{Tr}(M_2 Q)]^2. \end{aligned} \tag{69}$$

Also, for future reference, define $\mathbb{m}_1 = \mathbb{E}[\tilde{w}_t^i]$ and $\mathbb{M}_1 = \mathbb{E}[(\tilde{w}_t^i - \mathbb{M}_1)(\tilde{w}_t^i - \mathbb{m}_1)^\intercal]$.

Lemma 13. *The risk-constrained optimization problem in (67) can be reformulated as*

$$\begin{aligned} \text{minimize} \quad & J = \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^T c_t^i\right] \\ \text{s.t.} \quad & (68) \ \text{ and,} \\ & \tilde{J}_c = J_{\bar{c}} + \sum_{i=1}^n \tilde{J}_c^i \le \Lambda, \quad \forall i \in \mathbb{N}_n, \end{aligned} \tag{70}$$

where

$$\begin{aligned} J_{\tilde{c}}^i &= \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^T \frac{4}{n}(\tilde{x}_t^i)^\intercal Q M_2 Q \tilde{x}_t^i\right], \\ J_{\bar{c}} &= \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^T 4(\bar{x}_t)^\intercal Q M_2 Q \bar{x}_t + 4(\bar{x}_t)^\intercal Q M_3\right], \end{aligned}$$

and $\Lambda = \Gamma - m_4 + Tr(M_2 Q)^2$.

Proof. Using the result in [83], the constraint in (67c) can be reformulated as

$$J_c = \frac{1}{n}\sum_{i=1}^{n} \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\sum_{t=0}^{T} 4(x_t^i)^\intercal Q M_2 Q x_t^i + 4(x_t^i)^\intercal Q M_3.$$

The proof follows immediately by rewriting the above equation as $x_t^i = \tilde{x}_t^i + \bar{x}_t$, and on noting that $\sum_{i=1}^n \tilde{x}_t^i = 0$. □

6.3.2 Primal-Dual Approach

To solve the constrained optimization problem (70), we use $\lambda \geq 0$ as the Lagrange multiplier. The Lagrangian can then be expressed as

$$\mathcal{L}(\mathbf{\Phi}, \lambda) = J + \lambda(\tilde{J}_c - \Lambda). \tag{71}$$

Definition 5. *Define the matrices* $Q_c = \frac{4}{n} Q M_2 Q$, $Q_{\bar{c}} = 4 Q M_2 Q$, $Q_\lambda = \frac{1}{n} Q + \lambda Q_c$, *and* $Q_{\bar{\lambda}} = Q + \bar{Q} + \lambda Q_{\bar{c}}$.

Lemma 14. *The Lagrangian in* (71) *can be reformulated as*

$$\mathcal{L}(\mathbf{\Phi}, \lambda) = \bar{\mathcal{L}} + \sum_{i=1}^{n} \mathcal{L}^i, \tag{72}$$

where

$$\mathcal{L}^i = \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[(\tilde{x}_t^i)^\intercal (Q_\lambda)\tilde{x}_t^i + (\tilde{u}_t^i)^\intercal \frac{1}{n} R \tilde{u}_t^i\right],$$
$$\bar{\mathcal{L}} = \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\bar{x}_t^\intercal (Q_{\bar{\lambda}})\bar{x}_t + S_\lambda \bar{x}_t + \bar{u}_t^\intercal (R + \bar{R})\bar{u}_t\right].$$

Proof. The result follows directly from Lemma 13, the definition of the per-step cost in (65), and on noting that $\sum_{i=1}^n \tilde{x}_t^i = 0$ and $\sum_{i=1}^n \tilde{u}_t^i = 0$. □

To solve for the optimal value of the Lagrangian $\mathcal{L}^*$ in (71), we find the general form of the policies for a constant multiplier.

Theorem 6. *For a fixed multiplier λ, the optimal policy for each player is affine, such that*

$$u_t^i = -\theta(\lambda)x_t^i - (\bar{\theta}(\lambda) - \theta(\lambda))\bar{x}_t + \tau(\lambda) + \bar{\tau}(\lambda), \tag{73}$$

in which

$$\begin{aligned} \theta &= -(R + B^\intercal P B)^{-1} B^\intercal P A, \\ \bar{\theta} &= -(\mathcal{R} + \mathcal{B}^\intercal \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^\intercal \mathcal{P} \mathcal{A}, \end{aligned} \tag{74}$$

and

$$\begin{aligned} \tau &= -\frac{1}{2}(R + B^\intercal P B)^{-1} B^\intercal (2P\mathbb{m}_1 + g), \\ \bar{\tau} &= -\frac{1}{2}(\mathcal{R} + \mathcal{B}^\intercal \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^\intercal (2\mathcal{P} m_1 + \mathbf{g}), \end{aligned} \tag{75}$$

where P, $\mathcal{P}$, g and $\mathbf{g}$ are obtained by solving the following recursive equations

$$\begin{aligned} P &= Q_\lambda A^\intercal P A - A^\intercal P B (R + B^\intercal P B)^{-1} B^\intercal P A, \\ \mathcal{P} &= Q_{\bar{\lambda}} \mathcal{A}^\intercal \mathcal{P} \mathcal{A} - \mathcal{A}^\intercal \mathcal{P} \mathcal{B} (\mathcal{R} + \mathcal{B}^\intercal \mathcal{P} \mathcal{B})^{-1} \mathcal{B}^\intercal \mathcal{P} \mathcal{A}, \\ g^\intercal &= (2\mathbb{m}_1^\intercal P + g^\intercal)(A - B\theta), \\ \mathbf{g}^\intercal &= (2m_1^\intercal \mathcal{P} + \mathbf{g}^\intercal)(\mathcal{A} - \mathcal{B}\bar{\theta}) + 4\lambda (Q M_3)^\intercal. \end{aligned} \tag{76}$$

Proof. Define the generalized state and action of all agents in an augmented form as $\mathbf{x}_t = [\text{vec}(\tilde{x}_t^i)_{i=1}^n, \bar{x}_t]$ and $\mathbf{u}_t = [\text{vec}(\tilde{u}_t^i)_{i=1}^n, \bar{u}_t]$, respectively. Then, it follows that

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t,$$

where

$$\mathbf{A} = \text{diag}(\text{diag}(A)_{i=1}^n, \bar{A}), \quad \mathbf{B} = \text{diag}(\text{diag}(B)_{i=1}^n, \bar{B}).$$

Define the finite-horizon Lagrangian as the value function V_T and note that the results in

Theorem 2 of [83] imply that the Lagrangian has a quadratic form as

$$V_T = \mathbf{x}_t^\intercal \mathbf{P} \mathbf{x}_t + \mathbf{g} \mathbf{x}_t + \mathbf{z}_t.$$

Instead of solving for the optimal policy in the larger state-space of $\mathbf{x}_t$, from Lemma 14, the value function can also be decomposed into a set of smaller value functions such that

$$V_T = \bar{V}_T + \sum_{i=1}^{n} \tilde{V}_T^i.$$

Since the Lagrangians $\bar{\mathcal{L}}$ and $\tilde{\mathcal{L}}^i$ have complete square forms, the minimization can be carried out over the smaller state space of $\tilde{x}_t^i$ and $\bar{x}_t$. Therefore, by employing dynamic programming, we have the following two recursive optimality equations

$$\tilde{V}_T^i = \min_{\tilde{u}_t} \big((\tilde{x}_t^i)^\intercal Q_\lambda \tilde{x}_t^i + \frac{1}{n} (\tilde{u}_t^i)^\intercal R \tilde{u}_t^i + \bar{V}_{T+1}^i \big),$$
$$\bar{V}_T = \min_{\bar{u}_t} (\bar{x}_t^\intercal Q_{\bar{\lambda}} \bar{x}_t + \bar{u}_t^\intercal (R + \bar{R}) \bar{u}_t + \bar{V}_{T+1}).$$

The proof follows by taking the derivative with respect to $\tilde{u}_t^i$ and $\bar{u}_t$ and using backward dynamic programming. □

Remark 15 (Strong Duality). *Using the results established in Theorem 2 of [84] and [85], there exists an optimal multiplier λ^* such that the policy*

$$u_t^i = -\theta(\lambda^*) x_t^i - (\bar{\theta}(\lambda^*) - \theta(\lambda^*)) \bar{x}_t + \tau(\lambda^*) + \bar{\tau}(\lambda^*) \tag{77}$$

is the optimal solution to (70).

6.3.3 Solution of the Dual Problem with Subgradients

Since there is no optimality gap in the optimization problem (67), we can alternatively solve the following dual problem

$$\max_{\lambda \geq 0} \; D(\lambda) = \max_{\lambda \geq 0} \min_{\mathbf{u}} \; \mathcal{L}(\mathbf{u}, \lambda) \tag{78}$$

which is also concave in λ. Let d denote the subgradient. Then, from the results in [86,87] and the subgradient of $D(\lambda)$ can be expressed as

$$d = \tilde{J}_c(\theta, \bar{\theta}, \lambda) - \Lambda. \tag{79}$$

The following Theorem provides the explicit form of the constraints for deriving the subgradient vector.

Theorem 7. *Consider the stabilizing control input given by (73). Then,*

$$\begin{aligned} J_{\tilde{c}}^i &= Tr\Big[P_{\tilde{c}}\big(\mathbb{M}_2 + (B\tau + \mathbb{m}_1)(B\tau + \mathbb{m}_1)^\intercal\big)\Big] \\ &\quad + g_{\tilde{c}}^\intercal(B\tau + \mathbb{m}_1), \\ J_{\bar{c}} &= Tr\Big[P_{\bar{c}}(\mathcal{B}\bar{\tau} + m_1)(\mathcal{B}\bar{\tau} + m_1)^\intercal\Big] + g_{\bar{c}}^\intercal(\mathcal{B}\bar{\tau} + m_1), \end{aligned}$$

where $P_{\bar{c}}$ and P_c are the positive definite solutions of the following Lyapunov equations

$$\begin{aligned} P_{\tilde{c}} &= \frac{4}{n} Q M_2 Q + (A - B\theta)^\intercal P_{\tilde{c}}(A - B\theta), \\ P_{\bar{c}} &= 4 Q M_2 Q + (\mathcal{A} - \mathcal{B}\bar{\theta})^\intercal P_{\bar{c}}(\mathcal{A} - \mathcal{B}\bar{\theta}), \end{aligned} \tag{80}$$

where

$$\begin{aligned} g_{\tilde{c}}^\intercal &= 2\big[(B\tau + \mathbb{m}_1)^\intercal P_{\tilde{c}}(A - B\theta)\big]\big(I - A + B\theta\big)^{-1}, \\ g_{\bar{c}}^\intercal &= 2\big[(\mathcal{B}\bar{\tau} + m_1)^\intercal P_{\bar{c}}(\mathcal{A} - \mathcal{B}\bar{\theta}) + 2M_3^\intercal Q\big]\big(I - \mathcal{A} + \mathcal{B}\bar{\theta}\big)^{-1}. \end{aligned}$$

Proof. Define the relative value functions

$$\begin{aligned} V_{\tilde{c}}^i &= \mathbb{E}\Big[\sum_{t=0}^{\infty} \frac{4}{n}(\tilde{x}_t^i)^\intercal Q M_2 Q \tilde{x}_t^i - J_{\tilde{c}}^i\Big], \\ V_{\bar{c}} &= \mathbb{E}\Big[\sum_{t=0}^{\infty} 4(\bar{x}_t)^\intercal Q M_2 Q \bar{x}_t + 4(\bar{x}_t)^\intercal Q M_3 - J_{\bar{c}}\Big]. \end{aligned} \tag{81}$$

Using backward dynamic programming, it can be shown that such value functions have a quadratic form, i.e. $V_{\tilde{c}}^i = (\tilde{x}_t^i)^\intercal P_{\tilde{c}} \tilde{x}_t^i + g_{\tilde{c}}^\intercal \tilde{x}_t^i + z_{\tilde{c}}$ and $V_{\tilde{c}}^i = \bar{x}_t^\intercal P_{\bar{c}} \bar{x}_t + g_{\tilde{c}}^\intercal \bar{x}_t + z_{\bar{c}}$. Using the

Bellman equation, for $V_{\tilde{c}}^i$ one has

$$\begin{aligned}
V_{\tilde{c}}^i =&(\tilde{x}_t^i)^\intercal P_{\tilde{c}} \tilde{x}_t^i + g_{\tilde{c}}^\intercal \tilde{x}_t^i + z_{\tilde{c}} \\
=&\frac{4}{n}(\tilde{x}_t^i)^\intercal Q M_2 Q \tilde{x}_t^i - J_{\tilde{c}}^i + \mathbb{E}[g_{\tilde{c}}^\intercal \big((A - B\theta)\tilde{x}_t^i + B\tau + \tilde{w}_t^i\big)] + z_{\tilde{c}} \\
&+ \mathbb{E}[(A - B\theta)\tilde{x}_t^i + B\tau + \tilde{w}_t^i]^\intercal P_{\tilde{c}} [(A - B\theta)\tilde{x}_t^i + B\tau + \tilde{w}_t^i] \\
=&(\tilde{x}_t^i)^\intercal \left[\frac{4}{n} Q M_2 Q + (A - B\theta)^\intercal P_{\tilde{c}} (A - B\theta)\right](\tilde{x}_t^i) \\
&\left[2(B\tau + m_1)^\intercal P_{\tilde{c}} (A - B\theta) + g_{\tilde{c}}^\intercal (A - B\theta)\right](\tilde{x}_t^i) - J_{\tilde{c}}^i + z_{\tilde{c}} + \\
&\mathrm{Tr}\left[P_{\tilde{c}} (\mathbb{M}_2 + (B\tau + \mathbb{m}_1)(\mathbb{M}_2 + (B\tau + \mathbb{m}_1)^\intercal\right] + g_{\tilde{c}}^\intercal (B\tau + \mathbb{m}_1),
\end{aligned}$$

Then, it follows that

$$J_{\tilde{c}}^i = \mathrm{Tr}\left[P_{\tilde{c}} (\mathbb{M}_2 + (B\tau + \mathbb{m}_1)(\mathbb{M}_2 + (B\tau + \mathbb{m}_1)^\intercal\right] + g_{\tilde{c}}^\intercal (B\tau + \mathbb{m}_1).$$

Using a similar argument, $V_{\bar{c}}$ can be written as

$$\begin{aligned}
V_{\bar{c}} =&\bar{x}_t^\intercal P_{\bar{c}} \bar{x}_t + g_{\bar{c}}^\intercal \bar{x}_t \\
=&4\bar{x}_t^\intercal Q M_2 Q \bar{x}_t + 4M_3^\intercal Q \bar{x}_t - J_{\bar{c}} + \mathbb{E}[g_{\bar{c}}^\intercal \big((\mathcal{A} - \mathcal{B}\bar{\theta})\bar{x}_t + \mathcal{B}\tau + \bar{w}_t\big)] \\
&+ \mathbb{E}[(\mathcal{A} - \mathcal{B}\bar{\theta})\bar{x}_t + \mathcal{B}\tau + \bar{w}_t]^\intercal P_{\bar{c}} [(\mathcal{A} - \mathcal{B}\bar{\theta}))\bar{x}_t + \mathcal{B}\tau + \bar{w}_t] + z_{\bar{c}} \\
=&\bar{x}_t^\intercal \left[Q M_2 Q + (\mathcal{A} - \mathcal{B}\bar{\theta})^\intercal P_{\bar{c}} (\mathcal{A} - \mathcal{B}\bar{\theta})\right]\bar{x}_t \\
&\left[2(\mathcal{B}\tau + m_1)^\intercal P_{\bar{c}} (\mathcal{A} - \mathcal{B}\bar{\theta}) + 4M_3^\intercal Q + g_{\bar{c}}^\intercal (\mathcal{A} - \mathcal{B}\bar{\theta})\right]\bar{x}_t \\
&+ \mathrm{Tr}\left[P_{\bar{c}} (\mathcal{B}\tau + m_1)(\mathcal{B}\tau + m_1)^\intercal\right] + g_{\bar{c}}^\intercal (\mathcal{B}\tau + m_1) - J_{\bar{c}} + z_{\bar{c}},
\end{aligned}$$

and it results that

$$J_{\bar{c}} = \mathrm{Tr}\left[P_{\bar{c}} (\mathcal{B}\bar{\tau} + m_1)(\mathcal{B}\bar{\tau} + m_1)^\intercal\right] + g_{\bar{c}}^\intercal (\mathcal{B}\bar{\tau} + m_1).$$

□

From Theorem 7, we can compute $\tilde{J}_c = J_{\bar{c}} + \sum_{i=1}^{n} \tilde{J}_c^i$ and then find the subgradients, accordingly. Algorithm 1 describes the proposed primal-dual method to solve the optimization problem in (67).

Algorithm 5 Primal-Dual Algorithm for Risk-Constrained Mean-field LQR

Input: Initial λ_0, step size η

1: Iteration counter k
2: **for** $k = 1, 2, ...$ **do**
3: Obtain $u_t = \text{argmin}\mathcal{L}(\mathbf{u}_t, \boldsymbol{\lambda}_k)$ from Theorem 6
4: Compute d_k from Theorem 7
5: Update the multiplier $\lambda_{k+1} = [\lambda_k + \eta_k . d_k]_+$
6: **end for**

Remark 16. *Since the policy in* (73) *is stabilizable, the subgradients and multipliers vectors have upper bounds.*

Remark 17. *Since the subgradients and multipliers are upper bounded, using an argument analogous to that in Theorem 3 in [84], Algorithm 1 converges to the optimal policy after a sufficient numbers of iterations.*

6.4 Simulations

We validate the proposed methods using numerical simulations on a low-inertia microgrid (MG) system. Consider the load frequency problem (LFC) with risk constraints on the frequency and the mean state of all the agents. The MGs exchange information with each other through the mean-state of the system.

Parameter	Symbol	Value	Units
Damping Factor	D	16.66	MW/Hz
Speed Droop	R	1.2×10^{-3}	Hz/MW
Turbine Static Gain	K_t	1	MW/MW
Turbine Time Constant	T_t	0.3	s
Area Static Gain	K_p	0.06	Hz/MW
Area Time Constant	T_p	24	s
Tie-line Coefficient	K_{tie}	1090	MW/Hz

Consider microgrids in n areas. Let $\Delta P_{\text{tie},i}$ and Δf_a denote the power inflow and the frequency deviation corresponding to the ith microgrid. We assume that this power flow is proportional to the discrepancy of the frequency deviation of each area and the mean frequency deviation of all areas, i.e.

$$\Delta P_{\text{tie},i} = \int K_{\text{tie},i}(\Delta f_i - \Delta \bar{f})dt$$

In addition, the control signal of the ith area is the sum of two terms below

$$\Delta u_{tot,i} = \Delta P_{f,i} + \Delta P_{C,i},$$

where $\Delta P_{f,i} = -\frac{1}{R_i}\Delta f_i$, and $\Delta P_{C,i}$ denotes the automatic generation control (AGC). These two controls specify the output power of the microgrid at the ith area denoted by $\Delta P_{G,i}$. The other state variable is the area control error (ACE) denoted by $z_i := \beta\Delta f_i + \Delta P_{\text{tie},i}$ with the bias factor $\beta_i = D_i + \frac{1}{R_i}$.

The overall state of each microgrid is

$$x^i = [\Delta f_i, \Delta P_{G,i}, \Delta P_{\text{tie},i}, \int z_i].$$

The dynamics of the system is

$$x^i_{t+1} = Ax^i_t + \bar{A}\bar{x}_t + Bu^i_t,$$

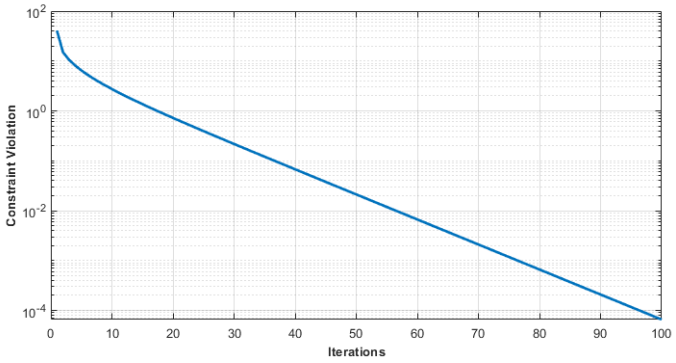

Figure 6.1: Constraint violation with iterations for the microgrid problem

where

$$A = \begin{bmatrix} -\frac{1}{T_p} & \frac{K_p}{T_p} & -\frac{K_p}{T_p} & 0 \\ -\frac{K_t}{RT_t} & -\frac{1}{T_t} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \beta & 0 & 1 & 0 \end{bmatrix}$$

$$\bar{A} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ K_{tie} & 0 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{K_t}{T_t} \\ 0 \end{bmatrix}$$

We use the parameters in Table I from [88]. Also, we select $Q = \text{diag}(800, 80, 80, 4000)$ and $R = 100$. Fig. 6.1 illustrates constraint violation as defined in [84]. It is observed that as the number of iterations grows, the constraint violation tends to zero. In other words, the algorithm obtains the optimal control law that minimizes the common cost function of players while the constraints are satisfied.

6.5 Conclusions

We proposed a computationally-efficient method to tackle the problem of risk-constrained control of mean-field linear quadratic systems. The method only requires the solution of two Riccati equations and is independent of the number of players. This is a feature that is essential in controlling a multi-agent system of large size. The application of policy gradient methods as an alternative approach and considering individual constraints for the players are two interesting topics for the extension of the current research.

Chapter 7

Conclusions & Future Research Directions

We now summarize the main contributions of this dissertation and present some ideas for future research in the area.

7.1 Conclusions

This dissertation studied optimization problems in interconnected systems with various applications. Although these problems are seemingly different, they can all be tackled by some control and optimization principles.

In the first part, we investigated the forward kinematics of a parallel robot. We showed that for certain geometries and specific types of parallel robots, one could estimate all workspace parameters as a function of only one parameter and the joint space variables (joint angles or lengths). The estimation errors are upper bounded as functions of the robot's geometry. Then, we formulated forward kinematics as an error minimization problem. Finally, for a given configuration, we proved that a gradient descent approach always estimates the forward kinematics parameters with sufficient accuracy.

In the second part, we studied methods for output power maximization of Coulomb force parametric generators (CFPGs) as another dynamical system. CFPGs are a special

class of micro-energy harvesters, for which we introduce several electrostatic force estimators. Bearing in mind the constraint on the computational complexity of the estimators, we considered several solutions ranging from machine learning and optimization methods to an approach based on the kinematics of a CFPG. Finally, we ran simulations to validate their performance by the acceleration data recorded from volunteers to emulate daily human activities.

In the third part, we studied mean-field linear quadratic systems as a special class of multi-agent networks. First, we considered a game setting and rigorously proved the convergence of the policy gradient (PG) methods. Then, we focused on a cooperative setting with constraints on the cumulative per-step state variance of all players and introduced a primal-dual approach that converges to the optimal solution of the problem.

We highlight the following contributions in this dissertation:

(1) Trade-off between the computational complexity and accuracy of an optimization algorithm: In some applications such as wearable and medical implants, the available computational resources are limited. As a compromise, we present an approximate solution with relatively low computational complexity, while an acceptable accuracy is ensured.

(2) In some applications an approximate solution may serve the purpose. For instance, regarding the problem on forward kinematics of parallel manipulators, we consider a simplified form of the kinematics at the cost of some bounded estimation error.

(3) Imposing certain conditions on the structure of an optimization problem can simplify the solution. Regarding control of multi-agent systems, we note that the mean-field dynamics can be used to find the optimal control by solving two Riccati equations of a reduced order. This is important as it helps to find a solution whose complexity does not grow with the number of players.

(4) Regarding the energy harvesting problem, we We argue that each of the proposed methods can be the optimal solution for a specific acceleration waveform.

7.2 Future Research Directions

The optimization topics studied in this dissertation all have the potential to introduce new research directions in both theory and practice.

Regarding the research on the FK of parallel robots, the proposed methodology may be extended to other types of parallel robots. Specifically, in some applications, the robot is designed such that one of the FK parameters can be directly estimated from the joint space variables. Furthermore, in some robotics settings, there are constraints on the range of the FK parameters, allowing for the linearization of the kinematic constraints with minor and bounded estimation errors. Therefore, it may be possible to obtain estimates of the other FK parameters as a function of the joint variables and the previously estimated FK parameter. Such a methodology can have applications in motion systems, where high accuracy FK estimations with relatively low computational burden are required.

As for the research on energy maximization of CFPGs, there are several possible ways to improve the results. Given the non-stationary nature of the acceleration signals, one can employ contextual multi-armed bandit methods that may provide more accurate estimates of the electrostatic force in a CFPG by considering the acceleration data. Furthermore, given an acceleration profile, one can use an adaptive decision interval length (as a secondary optimization variable) for enhanced amounts of output power.

The research on the control of mean-field linear quadratic systems may also be further expanded in a more general framework. As for the cooperative setting, one can study the convergence of PG methods with the cumulative state variance of all players. Finally, considering a game setting with risk constraints may be a theoretical prospect for future research.

Bibliography

[1] K. Wen, C. Gosselin, “Forward Kinematic Analysis of Kinematically Redundant Hybrid Parallel Robots," *Journal of Mechanisms and Robotics*, Dec. 2020, 12(6):061008.

[2] A. Pott, V. Schmidt, “On the Forward Kinematics of Cable-driven Parallel Robots," in Proceedings of the *IEEE International Conference on Intelligent Robots and Systems*, Sep. 2015, pp. 3182-3187.

[3] Y. Liu, M. Kong, N. Wan, P. Ben-Tzvi, “A Geometric Approach to Obtain the Closed-form Forward Kinematics of H4 Parallel Robot," *Journal of Mechanisms and Robotics*. Oct. 2018, 10(5):051013.

[4] J. Sanjuan, M. Rahman, I. Rulik “Forward Kinematic Analysis of Dobot Using Closed-loop Method," *International Journal of Robotics and Automation*. Sep. 2020, 9(3): 153.

[5] N. Rojas, F. Thomas, “The Forward Kinematics of 3-R$\underline{P}$R Planar Robots: A Review and a Distance-Based Formulation," in *IEEE Transactions on Robotics*. Dec. 2010, 27(1):143-150.

[6] C. D. Zhang, S. M. Song, “Forward Kinematics of a Class of Parallel (Stewart) Platforms with Closed-form Solutions," *Journal of Robotic Systems*. Feb. 1992, 9(1): 93-112.

[7] Y. Wang, J. Yu, X. Pei, “Fast Forward Kinematics Algorithm for Real-time and High-precision Control of the 3-RPS Parallel Mechanism," *Frontiers of Mechanical Engineering*, Sep. 2018, 13(3):368-375.

[8] H. Q. Zhang, Q. Gao, M. Zhang, Y. A. Yao, “Forward Kinematics of Parallel Robot Based on Neural Network Newton-Raphson Iterative Algorithm," in Proceedings of the *International Conference on Intelligent Equipment and Special Robots*, Dec. 2021, Vol. 12127, pp. 347-352.

[9] A. Prado, H. Zhang, S. K. Agrawal, ”Artificial Neural Networks to Solve Forward Kinematics of a Wearable Parallel Robot with Semi-rigid Links," in Proceedings of the *IEEE International Conference on Robotics and Automation*, May 2021, 30, pp. 14524-14530.

[10] Q. Zhu, Z. Zhang, “An Efficient Numerical Method for Forward Kinematics of Parallel Robots," *IEEE Access*, Sep. 2019, 7:128758-128766.

[11] C. Yang, Q. Huang, P. O. Ogbobe, J. Han, “Forward Kinematics Analysis of Parallel Robots Using Global Newton-Raphson Method," in *Proceedings of the Second International Conference on Intelligent Computation Technology and Automation*, Oct. 2009, pp. 407-410.

[12] H. Dai, G. Izatt, R. Tedrake, “Global Inverse Kinematics via Mixed-integer Convex Optimization," *The International Journal of Robotics Research*, Oct. 2019, 38(12-13):1420-1441.

[13] X. Yang, H. Wu, Y. Li, B. Chen, “A Dual Quaternion Solution to the Forward Kinematics of a Class of Six-DOF Parallel Robots with Full or Reductant Actuation," *Mechanism and Machine Theory*. Jan. 2017, 107: 27-36.

[14] A. Morell, M. Tarokh, L. Acosta, “Solving the Forward Kinematics Problem in Parallel Robots Using Support Vector Regression," *Engineering Applications of Artificial Intelligence*. Aug. 2013, 26(7): 1698-1706.

[15] D. Zhang, Z. Gao, “Forward Kinematics, Performance Analysis, and Multi-Objective Optimization of a Bio-inspired Parallel Manipulator," *Robotics and Computer-Integrated Manufacturing*. Aug. 2012, 28(4): 484-492.

[16] G. Liu, Y. Wang, Y. Zhang, Z. Xie, “Real-time Solution of the Forward Kinematics for a Parallel Haptic Device Using a Numerical Approach Based on Neural Networks," *Journal of Mechanical Science and Technology*. Jun. 2015, 29(6):2487-2499.

[17] M. Dehghani, M. Ahmadi, A. Khayatian, M. Eghtesad, M. Farid, “Neural Network Solution for Forward Kinematics Problem of HEXA Parallel Robot," in *Proceedings of the American Control Conference*, Jun. 2008, pp. 4214-4219.

[18] P. K. Jamwal, S. Q. Xie, Y. H. Tsoi, K. C. Aw, “Forward Kinematics Modelling of a Parallel Ankle Rehabilitation Robot Using Modified Fuzzy Inference," *Mechanism and Machine Theory*. Nov. 2010, 45(11): 1537-1554.

[19] J. P. Merlet, “Kinematics of the Wire-driven Parallel Robot MARIONET Using Linear Actuators," in Proceedings of the *IEEE International Conference on Robotics and Automation*, May 2008, pp. 3857-3862.

[20] M. Roudneshin, K. Ghaffari, A. G. Aghdam, “On Forward Kinematics of a 3SPR Parallel Manipulator," *arXiv preprint* :2205.15518. May 2022.

[21] F. Pierrot, O. Company, “H4: A New Family of 4-DoF Parallel Robots," in *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Sep. 1999, pp. 508-513.

[22] J. P. Merlet, “Parallel Robots," *Springer Science & Business Media*; Dec. 2005.

[23] A. Jadbabaie, A. Rakhlin, S. Shahrampour, K. Sridharan, “Online Optimization: Competing with dynamic Comparators," in *Artificial Intelligence and Statistics* . Feb. 2015, pp. 398-406.

[24] E. Hazan, “Introduction to Online Convex Optimization," *Foundations and Trends® in Optimization.* Aug. 2016, 2(3-4):157-325.

[25] D. Briand, E. Yeatman, O. Brand, C. Hierold, S. Roundy, J. G. Korvink, G. K. Fedder,O. Tabata, “Micro Energy Harvesting," *John Wiley & Sons*; Apr. 2015.

[26] M. Magno, D. Kneubuhler, P. Mayer, L. Benini, “Micro Kinetic Energy Harvesting for Autonomous Wearable Devices," in Proceedings of the *IEEE International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)* , Jun. 2018, pp. 105-110.

[27] K. Li, Q. He, J. Wang, Z. Zhou, X. Li, “Wearable Energy Harvesters Generating Electricity from Low-frequency Human Limb Movement," *Microsystems & Nanoengineering*. Sep. 2018, 4(1):1-3.

[28] H. C. Koydemir and A. Ozcan, “Wearable and Implantable Sensors for Biomedical Applications,” *Annual Review of Analytical Chemistry*, Jun. 2018, vol. 11, pp. 127–146.

[29] Y. Khan, A. E. Ostfeld, C. M. Lochner, A. Pierre, and A. C. Arias,“Monitoring of Vital Signs with Flexible and Wearable Medical Devices,” *Advanced Materials*, Jun. 2016, vol. 28, no. 22, pp. 4373–4395.

[30] A. Cadei, A. Dionisi, E. Sardini, M. Serpelloni, “Kinetic and Thermal Energy Harvesters for Implantable Medical Devices and Biomedical Autonomous Sensors," *Measurement Science and Technology*. Nov. 2013, 25(1):012003.

[31] M. R. Azghadi, C. Lammie, J. K. Eshraghian, M. Payvand, E. Donati, B. Linares-Barranco, G. Indiveri, “Hardware Implementation of Deep Network Accelerators Towards Healthcare and Biomedical Application," *IEEE Transactions on Biomedical Circuits and Systems*. Nov. 2020, 14(6):1138-1159.

[32] E. Donati, M. Payvand, N. Risi, R. Krause, G. Indiveri, “Discrimination of EMG Signals Using a Neuromorphic Implementation of a Spiking Neural Network," *IEEE Transactions on Biomedical Bircuits and Systems.* Jun. 2019; 13(5):795-803.

[33] P. D. Mitcheson, T. Sterken, C. He, E. M. Kiziroglou, E. M. Yeatman, and R. Puers, “Electrostatic Microgenerators,” *Measurement and Control*, vol. 41, no. 4, pp. 114–119, 2008.

[34] P. D. Mitcheson, “Analysis and Optimisation of Energy-Harvesting Micro-Generator Systems,” *Ph.D. Dissertation,* Imperial College London, 2005.

[35] T. Von Buren, P. D. Mitcheson, T. C. Green, E. M. Yeatman, A. S. Holmes, G. Troster, “Optimization of Inertial Micropower Generators for Human Walking Motion," *IEEE Sensors journal*. Jan. 2006; 6(1):28-38.

[36] P. D. Mitcheson, “Energy Harvesting for Human Wearable and Implantable Bio-Sensors," in Proceedings of the *Annual International Conference of the IEEE Engineering in Medicine and Biology*, Aug. 2010, (pp. 3432-3436).

[37] D. Budić, D. Šimunić, and K. Sayrafian, “Kinetic-Based Micro Energy-Harvesting for Wearable Sensors," in Proceedings of the *6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, 2015.

[38] N. Yarkony, K. Sayrafian, and A. Possolo, “Energy Harvesting from the Human Leg Motion,” in *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*, 2014, pp. 88–92.

[39] M. Dadfarnia, K. Sayrafian, P. D. Mitcheson, and J. S. Baras, “Maximizing Output Power of a CFPG Micro Energy-Harvester for Wearable Medical Sensors," in *Proceedings of the 4th International Conference on Wireless Mobile Communication and Healthcare-Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH)*, 2014.

[40] M. Roudneshin, K. Sayrafian, and A. G. Aghdam, “A Machine Learning Approach to the Estimation of Near-Optimal Electrostatic Force in Micro Energy-Harvesters," in *Proceedings of the IEEE International Conference on Wireless for Space and Extreme Environments*, Oct. 2019.

[41] M. Roudneshin, K. Sayrafian, and A. G. Aghdam, “Adaptive Estimation of Near-Optimal Electrostatic Force in Micro Energy-Harvesters," in *Proceedings of the IEEE International Conference on Control Technology and Applications*, 2020.

[42] M. Roudneshin, K. Sayrafian, and A. G. Aghdam, “Maximizing Harvested Energy in Coulomb Force Parametric Generators," in *Proceedings of the American Control Conference (ACC)*, Jun. 2022.

[43] P. Batista, C. Silvestre, P. Oliveira, B. Cardeira, “Accelerometer Calibration and Dynamic Bias and Gravity Estimation: Analysis, Design, and Experimental Evaluation,” *IEEE Transactions on Control Systems Technology*, Oct. 2010, 19(5):1128-1137.

[44] K. P. Murphy, “Machine Learning: A Probabilistic Perspective," *MIT press*, Sep. 2012.

[45] R. S. Sutton, A. G. Barto, “Reinforcement Learning: An Introduction," *MIT press*, Nov. 2018.

[46] D. Bouneffouf, I. Rish, C. Aggarwal, “Survey on Applications of Multi-armed and Contextual Bandits," In *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, Jul. 2020, pp. 1-8.

[47] L. Zhou, “A Survey on Contextual Multi-armed Bandits," *arXiv preprint*:1508.03326. Aug. 2015.

[48] T. Von Buren, P. D. Mitcheson, T. C. Green, E. M. Yeatman, A. S. Holmes, G. Troster, “Optimization of Inertial Micropower Generators for Human Walking Motion," *IEEE Sensors journal*, Jan. 2006,6(1):28-38.

[49] C. Cepnik, R. Lausecker, U. Wallrabe, “Review on Electrodynamic Energy Harvesters—A Classification Approach," *Micromachines*. Jun. 2013, 4(2):168-196.

[50] M. P. Deisenroth, A. A. Faisal, C. S. Ong, " Mathematics for Machine Learning," *Cambridge University Press*; Apr. 2020.

[51] M. Fazel, R. Ge, S. M. Kakade, and M. Mesbahi, “Global convergence of Policy Gradient Methods for the Linear Quadratic Regulator,” *arXiv preprint arXiv:1801.05039*, 2018.

[52] K. Zhang, Z. Yang, and T. Basar, “Policy Optimization Provably Converges to Nash Equilibria in Zero-sum Linear Quadratic Games,” in *Advances in NeurIPS*, 2019, pp. 11 598–11 610.

[53] E. Mazumdar, L. J. Ratliff, M. I. Jordan, and S. S. Sastry, "Policy-gradient Algorithms Have no Guarantees of Convergence in Continuous Action and State Multi-agent Settings," *arXiv preprint:1907.03712*, 2019.

[54] J. Arabneydi and A. G. Aghdam, "Deep Teams: Decentralized Decision Making with Finite and Infinite Number of Agents," *IEEE Transactions on Automatic Control*, DOI: 10.1109/TAC.2020.2966035, 2020.

[55] J. Arabneydi and A. G. Aghdam, "Deep Structured Teams with Linear Quadratic Model: Partial Equivariance and Gauge Transformation," *https://arxiv.org/abs/1912.03951*, 2019.

[56] J. Arabneydi and A. G. Aghdam, "Deep structured Teams and Games with Markov-chain Model: Finite and Infinite Number of Players," *submitted,* 2019.

[57] J. Arabneydi, M. Roudneshin, and A. G. Aghdam, "Reinforcement Learning in Deep Structured Teams: Initial Results with Finite and Infinite Valued Features," in Proceedings of the *IEEE Conference on Control Technology and Applications*, 2020.

[58] V. Fathi, J. Arabneydi, and A. G. Aghdam, "Reinforcement Learning in Linear Quadratic Deep Structured Teams: Global Convergence of Policy Gradient Methods," in Proceedings of the *59th IEEE Conference on Decision and Control*, 2020.

[59] P. E. Caines, M. Huang, and R. P. Malhame, "Mean Field Games," in *Handbook of Dynamic Game Theory*, T. Basar and G. Zaccour, Eds. Springer International Publishing, 2018, pp. 345–372.

[60] J. Arabneydi, "New Concepts in Team Theory: Mean field Teams and Reinforcement Learning," *Ph.D. Dissertation*, Department of Electrical and Computer Engineering, McGill University, Canada, 2016.

[61] R. Elliott, X. Li, and Y.-H. Ni, "Discrete Time Mean-field Stochastic Linear-quadratic Optimal Control Problems," *Automatica*, vol. 49, no. 11, pp. 3222–3233, 2013.

[62] A. Bensoussan, J. Frehse, and P. Yam, “Mean field Games and Mean Field Type Control Theory," *Springer-Verlag New York*, 2013.

[63] R. Carmona and F. Delarue, “Probabilistic Theory of Mean Field Games with Applications I-II," *Springer*, 2018.

[64] J. Arabneydi and A. Mahajan, “Linear Quadratic Mean Field Teams: Optimal and Approximately Optimal decentralized solutions,” Available at *https://arxiv.org/abs/1609.00056*, 2016.

[65] J. Arabneydi and A. Mahajan, “Team-optimal Solution of Finite Number of Mean-field Coupled LQG Subsystems,” in Proceedings of the *54th IEEE Conference on Decision and Control*, Dec. 2015, pp. 5308 – 5313.

[66] M. Baharloo, J. Arabneydi, and A. G. Aghdam, “Near-optimal Control Strategy in Leader-follower Networks: A case Study for Linear Quadratic Mean-field Teams,” in Proceedings of the *57th IEEE Conference on Decision and Control*, Dec. 2018, pp. 3288–3293.

[67] M. Baharloo, J. Arabneydi, and A. G. Aghdam, “Minmax Mean-field Team Approach for a Leader-follower Network: A Saddle-point Strategy,” *IEEE Control Systems Letters*, Jun. 2019, vol. 4, no. 1, pp. 121–126.

[68] J. Arabneydi and A. G. Aghdam, “Optimal Dynamic Pricing for Binary Demands in Smart Grids: A Fair and Privacy-preserving Strategy,” in Proceedings of the *American Control Conference*, 2018, pp. 5368–5373.

[69] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, P. L. Bartlett, and M. J. Wainwright, “Derivative-free Methods for Policy Optimization: Guarantees for Linear Quadratic Systems,” *Journal of Machine Learning Research*, Apr. 2019, vol. 21, no. 21, pp. 1–51.

[70] B. Polyak, “Gradient Methods for Solving Equations and Inequalities,” *USSR Computational Mathematics and Mathematical Physics*, 1964, vol. 4, no. 6, pp. 17–32.

[71] S. Lojasiewicz, “A Topological Property of Real A Analytic subsets,” *Coll. du CNRS, Les equations aux derivees partielles*, 1963, pp. 87–89.

[72] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, “Online Convex Optimization in the Bandit Setting: Gradient Descent without a Gradient,” in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2005, pp. 385–394.

[73] L. C. Evans, “An Introduction to Mathematical Optimal Control Theory," *UC Berkeley Lecture notes, Department of Mathematics*

[74] A. Majumdar, M. Pavone, “How Should a Robot Assess Risk? Towards an Axiomatic Theory of Risk in Robotics," *Robotics Research*, 2020, pp. 75-84.

[75] A. A. Pereira, J. Binney, G. A. Hollinger, and G. S. Sukhatme, “Risk-Aware Path Planning for Autonomous Underwater Vehicles using Predictive Ocean Models,” *Journal of Field Robotics*, Sep. 2013, vol. 30, no. 5 pp. 741–762.

[76] C. S0 Tapiero, “Applied Stochastic Models and Control for Finance and Insurance," *Springer Science & Business Media*, Dec. 2012.

[77] Z. Wu, “Using Machine Learning Approach to Evaluate the Excessive Financialization Risks of Trading Enterprises," *Computational Economics*. Apr. 2022, 59(4):1607-1625.

[78] I. Zografopoulos, J. Ospina, X. Liu, C. Konstantinou, “Cyber-Physical Energy Systems Security: Threat Modeling, Risk Assessment, Resources, Metrics, and Case Studies," *IEEE Access*. Feb. 2021, 9:29775-29818.

[79] S. Bruno, S. Ahmed, A. Shapiro, and A. Street, “Risk-Neutral and Risk-Averse Approaches to Multistage Renewable Investment Planning under Uncertainty,” *European Journal of Operational Research*, vol. 250, no. 3, May 2016, pp. 979–989.

[80] N. Li, X. Li, Y. Shen, Z. Bi, M. Sun, “Risk Assessment Model Based on Multi-Agent Systems for Complex Product Design," *Information Systems Frontiers*, Apr. 2015,17(2):363-385.

[81] M. Roudneshin, J. Arabneydi, A. G. Aghdam, "Reinforcement Learning in Nonzero-sum Linear Quadratic Deep Structured Games: Global Convergence of Policy Optimization," in Proceedings of the *59th IEEE Conference on Decision and Control*, Dec. 2020, pp. 512-517.

[82] J. Arabneydi and A. G. Aghdam, "Deep Teams: Decentralized Decision Making with Finite and Infinite Number of Agents," *IEEE Transactions on Automatic Control*, DOI: 10.1109/TAC.2020.2966035, 2020.

[83] A. Tsiamis, D. S. Kalogerias, L. F. Chamon, A. Ribeiro,G. J. Pappas, "Risk-constrained Linear-Quadratic Regulators," in Proceedings of the *59th IEEE Conference on Decision and Control*. Dec. 2020, pp. 3040-3047.

[84] F. Zhao, K. You, T. Basar, "Infinite-horizon Risk-constrained Linear Quadratic Regulator with Average Cost," in Proceedings of the *60th IEEE Conference on Decision and Control*, Dec. 2021, pp. 390-395.

[85] F. Zhao, K. You, T. Basar, "Global Convergence of Policy Gradient Primal-dual Methods for Risk-constrained LQRs," *arXiv preprint arXiv:*2104.04901. Apr. 2021.

[86] Y. Nesterov, "Introductory Lectures on Convex Optimization: A Basic Course," *Springer Science & Business Media*, 2013, vol. 87.

[87] S. P. Boyd, and L. Vandenberghe, "Convex Optimization," *Cambridge University Press*, 2004.

[88] E. E. Vlahakis, L. D. Dritsas, G. D. Halikias, "Distributed LQR Design for Identical Dynamically Coupled Systems: Application to Load Frequency Control of Multi-area Power Grid," in Proceedings of the *IEEE 58th Conference on Decision and Control*. Dec. 2019, pp. 4471-4476.