Decentralized Control of Complex Systems: Managing Uncertainties and Multi-objective Optimization with Multiple Controllers

Misra, Rahul

DOI (link to publication from Publisher):
10.54337/aau763801042

Publication date:
2024

Document Version
Publisher's PDF, also known as Version of record

DECENTRALIZED CONTROL OF COMPLEX SYSTEMS

MANAGING UNCERTAINTIES AND MULTI-OBJECTIVE OPTIMIZATION WITH MULTIPLE CONTROLLERS

BY
RAHUL MISRA

PhD Thesis 2024

AALBORG UNIVERSITY
DENMARK

Decentralized Control of Complex Systems: Managing Uncertainties and Multi-objective Optimization with Multiple Controllers

PhD Thesis 2024

Rahul Misra

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7C
DK-9220 Aalborg

Submitted:	October 2024
Main Supervisor:	Professor Rafal Wisniewski Aalborg University
Co-supervisor:	Professor Carsten Skovmose Kallesøe Aalborg University
Assessment:	Associate Professor Henrik Schiøler (chair) Aalborg University, Denmark
	Professor Johan Karlsson KTH Royal Institute of Technology, Sweden
	Professor Vianney Perchet Institut Polytechique de Paris, France
PhD Series:	Technical Faculty of IT and Design, Aalborg University
Department:	Department of Electronic Systems

ISSN: 2446-1628
ISBN: 978-87-85239-50-1

Published by:
Aalborg University Open Publishing
Kroghstræde 1-3
DK – 9220 Aalborg Øst
aauopen@aau.dk

Abstract

The rapid advancement of technology and urban population growth has made critical systems like electrical grids, communication networks, and water distribution networks increasingly complex. These systems involve multiple decision-makers and are subject to variable factors such as fluctuating demand, unforeseen faults, and seasonal variations, making optimal control challenging. Managing these systems can be seen as a sequential decision-making process in an uncertain environment, where decision-makers may be cooperative, adversarial, or a mix of both. Traditional model-based control methods struggle to keep up with the evolving complexity and scale of these systems, often rendering their models inaccurate or outdated. In contrast, learning-based adaptive control, particularly model-free control, offers a solution by adjusting control strategies in real-time based on system data, eliminating the need for precise system models.

This dissertation explores how learning-based adaptive controllers can be designed to operate in environments with other controllers, meeting multiple objectives while remaining robust to uncertainties and disturbances. The central hypothesis is that optimal control for large, complex systems can be broken down into a series of decentralized optimization problems, solved independently by each controller at each time step. Under appropriate assumptions, these controllers can learn solutions that ensure system safety, optimality, and resilience to disturbances.

A key feature of the decentralized learning-based controllers developed in this work is their scalability and resilience to faults and cyber-attacks. Instead of relying on direct coordination between controllers, the control problem is modeled using game theory, treating the interactions as a non-cooperative game in which controllers operate independently. This game-theoretic approach is combined with reinforcement learning to handle uncertainty and interactions between controllers. Multi-objective optimization is addressed through game-theoretic and constrained optimization techniques. The practical application of these decentralized control algorithms is demonstrated in the context of water distribution networks with uncertain consumption demands. These methods are implemented and validated in the Smart Water Lab at Aalborg University.

Resumé

Den hurtige udvikling af teknologi og urbaniseringen har gjort vigtige infrastrukturer som elnet, kommunikationsnetværk og vandforsyningen mere komplekse. Disse systemer har mange beslutningstagere og påvirkes af forskellige faktorer, som svingende efterspørgsel, uforudsete fejl og sæsonmæssige variationer, hvilket gør det svært at styre dem optimalt. Styringen af disse systemer kan ses som en proces, hvor beslutningstagerne kan samarbejde med hinanden, modarbejde hinanden eller en kombination af disse. Traditionelle modelbaserede kontrolmetoder har svært ved at følge med disse systemers voksende kompleksitet og skala, og deres modeller bliver ofte upræcise eller forældede. Læringsbaseret adaptiv kontrol, især modelløs kontrol, tilbyder derimod en løsning ved at justere kontrolstrategier i realtid baseret på systemdata, så detaljerede modeller ikke er nødvendige.

Denne afhandling undersøger, hvordan læringsbaserede adaptive controllere kan designes til at fungere sammen med andre controllere, opfylde flere mål og samtidig være robuste over for usikkerheder og forstyrrelser. Den centrale hypotese er, at optimal kontrol for store, komplekse systemer kan opdeles i en række decentraliserede optimeringsproblemer, som løses uafhængigt af hver controller ved et hvert tidspunkt. Under passende forudsætninger kan disse controllere lære løsninger, der sikrer systemets sikkerhed, optimalitet og modstandsdygtighed over for forstyrrelser.

En vigtig egenskab ved de decentraliserede læringsbaserede controllere, der er udviklet i denne afhandling, er deres evne til at skaleres og modstå fejl og cyberangreb. I stedet for at kræve direkte koordinering mellem controllerne bliver kontrolproblemet modelleret ved hjælp af spilteori. Interaktion mellem controllerne ses som et ikkesamarbejdende spil, hvor hver controller opererer selvstændigt. Denne spilteoretiske tilgang kombineres med reinforcement learning for at håndtere usikkerheder og interaktioner mellem controllerne. Multi-objektiv optimering håndteres gennem spilteoretiske og begrænsede optimeringsmetoder. De praktiske anvendelser af disse decentraliserede kontrolalgoritmer demonstreres i vandforsyningsnetværk, hvor algoritmerne styrer forsyningsnetværk med usikre forbrugsbehov. Metoderne er implementeret og

testet i Smart Water Lab ved Aalborg Universitet.

Contents

Thesis Details

Thesis Title:	Decentralized Control of Complex Systems: Managing Uncertainties and Multi-objective Optimization with Multiple Controllers
Ph.D. Student:	Rahul Misra
Supervisors:	Prof. Rafał Wisniewski, Aalborg University
	Prof. Carsten Skovmose Kallesøe, Aalborg University

The main body of this thesis consists of the following papers.

[A] Rahul Misra, Rafał Wisniewski, Carsten S. Kallesøe, "Approximating the model of a water distribution network as a Markov decision process." *10th Vienna International Conference on Mathematical Modelling (MATHMOD)*, IFAC-PapersOnLine, vol. 55, no. 20, pp. 271–276, 2022.

[B] Rahul Misra, Rafał Wisniewski, Carsten S. Kallesøe, "Approximating solution of stochastic differential games for distributed control of a water network." *18th IFAC Workshop on Control Applications of Optimization (CAO), Gif sur Yvette, France, 18–22 July*, vol. 55, no. 16, pp. 110–115, 2022.

[C] Rahul Misra, Rafał Wisniewski, "On principle of optimality for safety-constrained Markov Decision Process and p-Safe Reinforcement Learning" *26th International Symposium on Mathematical Theory of Networks and Systems (MTNS), 19-23 August 2024.*

[D] Rahul Misra, Carsten S. Kallesøe, Rafał Wisniewski, "Decentralized control of a water distribution network using Repeated Games." *2023 27th International Conference on Methods and Models in Automation and Robotics (MMAR).*, IEEE, pp. 181–186, 2023.

[E] Rahul Misra, Rafał Wisniewski, Carsten Skovmose Kallesøe, Manuela L. Bujorianu, "Robust Correlated Equilibrium: Definition and Computation" *Automatica.*

In addition to the main papers, the following publications have also been made during the time period of the PhD project but have not been included in the PhD thesis.

[1] Rahul Misra, Rafał Wisniewski, and Alexander Zuyev, "Attitude stabilization of a satellite having only electromagnetic actuation using oscillating controls." *Aerospace*, vol. 9, no. 8, pp. 444, 2022.

[2] Rahul Misra, Rafał Wisniewski, and Özkan Karabacak, "Sum-of-Squares based computation of a Lyapunov function for proving stability of a satellite with electromagnetic actuation." *21st IFAC World Congress 2020, Germany*, vol. 53, no. 2, pp. 7380–7385, 2020.

[3] Saruch Satishkumar Rathore, Rahul Misra, Carsten S. Kallesøe, Rafał Wisniewski, "Leakage diagnosis with a contamination mitigation control framework using a graph theory based model." *Annual Reviews in Control*, vol. 55, pp. 498–519, 2023.

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted or published scientific papers which are listed above. Parts of the papers are used directly or indirectly in the extended summary of the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Department. The thesis is not in its present form acceptable for open publication but only in limited and closed circulation as copyright may not be ensured.

Preface

This PhD dissertation presents the author's work as part of the Smart Water Infrastructure (SWIft) project, which was funded by the Poul Due Jensen Foundation (Grundfos Foundation). The dissertation is organized as a collection of papers and is divided into two main parts. The first part includes five chapters:

- Chapter 1 introduces the state-of-the-art, hypodissertation, and research questions. The summary of contributions is listed at the end of this chapter.
- Chapter 2 summarizes key concepts from Game theory and Reinforcement learning that have been used in this dissertation.
- Chapters 3-5 summarize the key contributions of this dissertation, with additional details available in the associated papers.

The second part contains five self-contained papers that elaborate on the contributions of this dissertation in greater detail.

This PhD dissertation would not have been possible without the support of many individuals. Firstly, I would like to thank my supervisors, Prof. Rafał Wisniewski and Industrial Prof. Carsten Skovmose Kallesøe, for their guidance and support throughout my PhD journey. The extensive meetings with Rafał significantly shaped my development as a researcher, teaching me how to conceptualize my ideas within a proper mathematical framework. Complementing Rafał's theoretical expertise, Carsten provided invaluable practical knowledge about water distribution networks and assisted me in simulating various systems. Both Rafał and Carsten nominated me for the prestigious EliteForsk travel grant, which supported my research stays abroad.

I am also grateful to Research Director Eitan Altman for inviting me for a research stay at INRIA (French Institute for Research in Computer Science and Automation), Sophia Antipolis, France. The members of the INRIA Network Engineering and Operations team were very welcoming, and I especially thank researcher Samir M. Perlaza for introducing me to the concepts of regret matching and empirical risk minimization.

I would like to extend my gratitude to Research Fellow Manuela L. Bujorianu (UCL, London) for sharing her expertise in stochastic processes and collaborating with me on one of the papers. I also appreciate Prof. Alexander Zuyev (MPI, Magdeburg) for sharing his knowledge of nonlinear control and Lie brackets, as well as for collaborating on one of the papers.

In addition to my supervisors and external collaborators, I would like to thank all my colleagues, both former and present, in the Section of Automation and Control. A special thanks to Saruch, Henrik, Jorge, Mohammad, Salahuddin, Simon, Mirhan, Shahab, John, Özkan, and Aysegul. Their presence created a lively social environment at work, and they were always available for discussions about research topics. Saruch, who worked on the same project, assisted me with lab implementation and collaborated on one of the papers.

Finally, I want to express my heartfelt gratitude to my parents, who have always believed in me—even when I doubted myself—and have provided unwavering support. This dissertation is dedicated to them.

Rahul Misra
Aalborg University, October 1, 2024

Part I

Summary

Chapter 1

Introduction

Rapid progress in technology and urban population growth has led to increasing complexity in essential systems for the present society such as electrical networks, communications, and water networks. Owing to the presence of large number of decision-makers, and variable factors (for example, changing demands in consumption), the optimal operation of such systems can be seen as sequential control (or sequential decision-making) in an uncertain or unknown environment in the presence of other decision-makers. The other decision-makers in the environment may be cooperative, adversarial, or a mixture of two (as they might have multiple objectives and they could agree to cooperate on some objectives and are antagonistic to each other on the remaining objectives). This is especially true for large-scale societal systems with complex dynamics [5, 7]. Examples of such systems can be smart grids supplying electricity to consumers with variable demands [27], wireless networks [65], or water distribution networks where pumping stations supply water to consumers with variable demands [78]. Modeling complex dynamics using fundamental physics principles requires substantial cost and effort and therefore, control theorists have long desired to develop *learning based adaptive control* techniques [8]. Such controllers act on the real-time data from the considered system and either update an estimated model of the system (thus, indirectly updating the model-based controller) or directly update the controller based on the observed data. This control system design is referred to as model-free control and is in stark contrast to systems requiring modeling of the complex dynamics with subsequent design of controller (model-based control design). Another major drawback of model-based control design for complex systems is that such systems are usually large-scale and they continuously evolve (i.e. new components are added or removed from the system over the system's lifetime), which can invalidate a model-based control design [102]. For example, new controllers or sensors might be added or removed over the lifetime of the system, thereby requiring a redesign of the control scheme. This redesigning of the control scheme is

unacceptable since it involves extra costs due to the remodeling of the system, retuning of the control system parameters, and downtime of the system when installing the new control algorithm. Therefore, the control systems community has long desired an adaptable control design that can accommodate such changes on the fly. This is also referred to as *Plug and play* control scheme [102].

In this dissertation, we address the aforementioned challenges by designing decentralized control algorithms for large-scale systems. The reasoning behind choosing decentralized control algorithms is that it improves the scalability of the control system design for large-scale systems. This is because we can simply add new controllers as the scale of the considered system increases. Decentralized controllers are more resilient to faults in the controllers relative to centralized controllers, as we only need to repair one decentralized controller instead of the entire centralized control system. On a similar note, decentralization also makes the overall control system more resilient to cyber-attacks [28]. However, decentralization naturally requires the design of a coordination mechanism such that the decentralized controllers can reach some sort of consensus [79] or jointly converge to a desired equilibrium solution [47] and this is the price we pay for decentralization. Instead of designing a coordination mechanism, we propose to use the mathematical framework of Game theory and formulate the decentralized control problem as a non-cooperative game. The game is non-cooperative since the controllers are not allowed to communicate or coordinate amongst themselves. Furthermore, since we desire model-free or learning based adaptive controls (for bypassing the need for modeling of complex dynamics), we require a mathematical framework for learning in the presence of other decision-makers. Fortunately, Game theory is closely linked to the subject of Reinforcement learning [38]. Another aspect concerning control of complex large-scale systems is balancing multiple objectives simultaneously (for example guaranteeing the safety of the system while ensuring optimality). This aspect can also be modeled using Game theory [84] or Constrained Optimization (where the optimization problem involves optimizing a single objective subject to constraints that represent the other objectives) [67]. It turns out that constrained optimization is closely linked to the concept of 2-player zero-sum games. From a practical point of view, we consider the control of a water distribution network as a representative example of the aforementioned critical complex system for society. We consider arbitrary water distribution networks which consist of pumping stations (that are the controllers), and multiple consumers with uncertain consumption demands. The pumping stations and the consumers are connected via an extensive piping network. Decentralized control algorithms are designed for such systems that are robust to uncertain consumption demands. The practical implementation of the decentralized control algorithm is done in the Smart Water Lab at Aalborg University. We now present the current state of the art, summarizing the latest results on the relevant topics.

1 State of the Art

1.1 Optimization with multiple objectives

Optimization is a broad field of study and [6, 22] provide a good introduction. Multi-objective optimization problems (MOOP) (also referred to as multiple criteria optimization, multiple attributes decision making or multiple criteria decision-making, vector optimization) have been studied extensively by economists, engineers, and biologists [31, 67, 68, 70, 81]. Computer scientists and biologists approach MOOP using evolutionary methods inspired by natural selection. Simulated individuals represent candidate solutions, whose quality is evaluated by a fitness function. These algorithms are known as genetic or evolutionary algorithms [2, 31, 35]. These algorithms are based on an assumption that the evolution of species is 'always' optimal which has been criticized by some researchers [40, 67, 81]. Linear matrix inequalities (LMI's) are used to pose matrix constraints in an optimization problem. They have also been used to study MOOP in control theory [21, 90, 91]. In this context, the design objectives were a mix of H_∞ performance, H_2 performance, passivity, asymptotic disturbance rejection, time-domain constraints, and constraints on the closed-loop pole location. Another approach is to use Lexicographic Optimization which prioritizes objectives and solve for the objective priority-wise while keeping other objectives constrained close to their optimum value respectively. The number of objectives in the constraints increases as we go to the lower-priority objectives. This technique has been used to design multi-objective Model Predictive Controls (MPC) in [4, 55, 99].

Polynomial optimization and Moment methods

Polynomial Optimization deals with minimizing (or maximizing) a polynomial over certain semi-algebraic sets (i.e. sets defined by the intersection of finitely many polynomial inequalities). A common use of polynomial optimization is verifying if a polynomial is non-negative over a semi-algebraic set (see the book [60] for practical applications in finance, optimal control, and game theory). Checking the non-negativity of a high-dimensional polynomial over a semi-algebraic set is computationally intractable. Instead, researchers strive to check whether a polynomial has a Sum-of-Squares (SoS) decomposition (i.e. the polynomial can be written as a sum of squares of smaller polynomials or monomials) that implies non-negativity of the polynomial. Finding the SoS decomposition of a polynomial is equivalent to solving a Semidefinite Program (SDP) [82]. Applications of SoS optimization include computing Lyapunov functions and computing barrier certificates for safety-critical control systems [72, 80, 86, 113]. The major drawback of SoS optimization is the exponential growth in the size of the SDP as the number of variables or degree of the polynomial is increased [60, 61]. This is being addressed by exploiting sparsity in the optimization problem for reducing the size of SDP [57, 80, 110] and is an active area of research.

1.2 Dynamical systems theory and Optimal Control

Dynamical systems is a broad field of mathematics that deals with differential equations (which encode states evolving with time). The primary focus is to study whether the solutions of such differential equations (time-dependent state trajectories) converge to an equilibrium point, or an invariant set [56]. Such a dynamical system is called a *Stable* system. A dynamical system is said to be *Controllable* if we can design a control algorithm that ensures the convergence of all trajectories (due to any initial condition) to the desired equilibrium point, or to the desired invariant set [8]. The central idea behind control theory is to measure the state of the dynamical system and introduce *Feedback* based on the measurements for controlling the dynamical system [37]. The presence of feedback introduces desirable properties in the control system such as robustness to disturbances. However, feedback can also destabilize a stable system (for example, positive feedback system). Since there can be many state trajectories depending on the initial conditions, we instead associate an 'energy' function and study whether the energy of the system dissipates along the dynamics of the system. This energy function is called the Lyapunov function. This technique is extremely useful for studying asymptotic convergence of dynamical systems and applications include controlled systems such as robots, wind turbines, and water networks but can also be used to prove convergence of optimization algorithms as done by Polyak for "heavy ball" optimization [85, 112]. Optimal control theory is concerned with finding an optimal path to control a given dynamical system and is referred to as an Optimal Control Problem (OCP) in literature [41, 93]. Solving an OCP is difficult (or might be impossible) in general due to the presence of dynamical system constraints. However, numerical techniques exist for solving OCP which can broadly be classified as indirect methods or direct methods. Indirect methods such as Pontryagin's maximum principle rely on incorporating the dynamical system constraint in the objective by using time-varying Lagrange multipliers and thereafter solving the associated system at each time instant. In contrast, the direct methods rely on directly 'lifting' the OCP into an optimization problem (i.e. at each stage of optimization the discretized differential equation is included as a constraint in the lifted optimization problem) or directly solving the Hamilton-Jacobi-Bellman equation [59]. Arguably, the most celebrated technique for finding an approximate solution of an OCP directly is the Model Predictive control and the books [25, 64] provide a good introduction to the same.

1.3 Game theory

Borel initiated the study of game theory [19], while Cournot introduced the concept of Cournot Equilibrium [29]. Other early contributors include Pareto [39]. A formal theory of games with applications to economics, politics, and military decision-making was first developed by Von Neumann and Morgenstern in the seminal book [108]. The celebrated minimax theorem, zero-sum and nonzero-sum games, equilibrium points, ran-

domized (or mixed) strategies, and the concept of expected utilities were introduced in the same [108]. Thereafter, Nash characterized equilibrium for n-person games in [73, 76] and proved the requirement of randomized strategies for ensuring the existence of equilibrium points. The primary tool used by Nash for proving the existence of equilibrium points was Brouwer's fixed point theorem (see Appendix of [23]). For this fundamental contribution regarding the existence of equilibrium points, the equilibrium in games are referred to as Nash equilibrium. Nash also introduced cooperative games in [74] with bargaining problem [75]. Parthasarathy extended Von Neumann's minimax theorem to games with continuous action spaces where only one player needs to follow a mixed strategy (continuous distribution over a Lebesgue measure in the case of games with continuous action spaces) for the equilibrium to exist [83]. Sion also proposed an extension of Von Neumann's minimax theorem with cost being a real-valued function with respect to the player's decision variables that belong to convex (concave) and compact sets for the minimizer (maximizer). Stackelberg introduced the concept of Stackelberg equilibrium [109] where one of the players is a "leader" and announces his/her strategy to the other players or "followers". The "followers" optimize in response to the announced strategy of the "leader". Rosen introduced the concept of Generalized Nash Equilibrium (GNE), that is basically Nash equilibrium points of games where the decision space of each player is constrained under some coupling constraints [89]. The existence and uniqueness of GNE for n-player concave games was studied by Rosen in [89]. A significant contribution of this work was the diagonal concavity condition on the payoffs that ensure the existence of a unique equilibrium point for games with payoffs satisfying the aforementioned diagonal concavity conditions. Furthermore, Rosen also designed a gradient descent-based rule that allowed convergence to the equilibrium points for such games. The concept of Nash equilibrium in general n-person games (without any assumptions of Rosen) was found to be computationally hard in [30] and incompatible with the Bayesian interpretation of Probability theory. Specifically, it was argued in [10] that there is no reason why all players should choose decisions to converge to the Nash equilibrium. To resolve these issues, Aummann proposed the concept of Correlated Equilibrium in [9, 10]. It should be noted that all the notions of game-theoretic equilibrium mentioned so far are "static" in nature meaning that they do not take into account how individual agents or decision-makers reach such equilibrium points.

Blackwell studied Von Neumann's Minimax theorem from the point of view of repeated game setting wherein he considered the following question: Suppose a 2-player static game is repeated T times, then is it possible that player 1 can asymptotically ensure that the time-average payoff converges to a predefined set no matter what player 2 does? Von Neumann's minimax theorem (and equivalently Sion's Minimax theorem) answers this question if the payoff of player 1 is a scalar. Unlike standard Von Neumann's theorem, in [18] Blackwell considers a vector of payoffs and provides sufficient conditions under which the time-average of vector payoffs approaches the target set. For convex

sets, Blackwell provides necessary and sufficient conditions to ensure approachability. Furthermore, the concept of Approachability can be seen as optimizing a multi-objective optimization problem in a game theoretic sense (as player 2 is trying to ensure that the desired set is not approachable by player 1). Shortly, after Blackwell's paper, Hannan introduced the concept of regret (i.e. in hindsight, how much better payoff could have been obtained if a different action was chosen) for repeated games in [44]. It was noted by Blackwell that strategies that ensure approachability also have no regret in Hannan's sense. Hart and Mas-Colell designed an algorithm that minimizes internal regret in [45] and proved its convergence using Blackwell's Approachability theorem. Subsequently, Abernethy showed that Approachability and minimizing internal regret are equivalent in the paper [1]. See paper [84] for a survey on Approachability and the related results on Regret Minimization and Calibration.

1.4 Learning in Games

Since the dawn of the theory of games, economists and decision-makers have always argued why any independent rational decision-makers should converge to a game-theoretic notion of an equilibrium such as Nash equilibrium. Therefore, research has been focused on finding adaptive learning rules that each decision-maker can follow independently (i.e. their decisions are uncoupled from each other). The first adaptive learning rule was the "fictitious play" rule proposed in [24, 88] in 1951. However, it was discovered that fictitious play does not guarantee convergence to the Nash equilibrium in nonzero-sum games [95] or games with more than 2 actions or more than 2 players [52]. In fact, careful analysis by Benaim and Hirsch on fictitious play dynamics in [15] showed that the learning dynamics become chaotic and there is no hope for convergence to equilibrium in general games. It has been argued that the concept of game theoretic equilibrium would only make sense in real-world applications such as markets if each agent can find simple adaptive learning rules that guarantee convergence to the Nash equilibrium as mentioned in the famous quote: *"If your laptop cannot find it, then neither can the market"* - Kamal Jain [77]. Unfortunately, despite more than 70 years of research, economists could not find simple adaptive learning rules that decision-makers can use for learning Nash equilibrium in general [47]. In fact, it was proven in [46] that it is impossible for agents to follow simple uncoupled adaptive learning rules which lead to convergence to Nash Equilibrium in general. In stark contrast to negative results regarding the convergence of simple adaptive learning rules to Nash equilibrium, there exist simple adaptive learning rules that converge to the set of Correlated Equilibrium or the weaker notion of the Hannan set [44]. One such simple adaptive learning rule is Regret-Matching introduced in [45] that guarantees convergence to the set of Correlated Equilibrium. Another algorithm is Calibration introduced in [36] which was initially introduced for predicting occurrences of a random event. Such algorithms minimize a notion of regret and hence are called "No-regret" Algorithms [43] and recently they have applied for bidding in

Smart grids in [54]. The books [26, 38] summarize nicely most of the adaptive learning rules in game theory. Following the work of Rosen in [89], there have been developments in learning equilibrium for convex/concave games with coupling constraints and continuous action spaces under the restrictions of strict monotonicity in [106, 107]. Note that these learning algorithms are dependent on the constraints introduced by Rosen on the payoffs in [89] that ensure the existence of a unique equilibrium point. Designing learning algorithms for games without Rosen's payoff assumptions (that ensure the uniqueness of Nash equilibrium points) is still challenging as such games have multiple Nash equilibrium points. Furthermore, there needs to be some sort of coordination towards the equilibrium point among players. This can be either via partial information exchange or a correlating signal [13, 97].

1.5 Games with states and links with Reinforcement learning

Machine Learning can be broadly classified as supervised learning (learning from labeled data), unsupervised learning (learning from unlabeled data), or Reinforcement Learning (learning from experiences). In Reinforcement Learning (RL), the agent (or the decision-maker) receives a reward if a favorable action is taken and the goal of the agent is to maximize reward based on experiences in accumulating rewards over time either during training or during online implementation [103]. RL methods can be broadly classified as Direct RL (which directly computes either the value function or control policy) or Indirect RL (which first computes the transition dynamics and then the value function or the control policy) [16, 104]. Direct RL can be further subclassified as value-based (ex: Q-learning [111], SARSA [98]), policy-based (ex: Proximal Policy Optimization [92]), or a combination of value and policy (ex: Actor-Critic [58]). MDP is the underlying mathematical framework behind RL [17, 66] and has been studied extensively since the pioneering work of Bellman [14] and Shapley [96] in 1950's. The book [87] and the lecture notes [53] are a good reference on this topic. The central result in MDP literature is the Bellman's Optimality Equations and the associated concepts of Dynamic Programming and the "Priciple of Optimality" that allow a large optimization problem to be broken down into smaller optimization problems that can be solved sequentially. MDPs with constraints have also been studied extensively and the book [3] covers the topic well. Some RL algorithms developed for constrained MDPs are Actor-Critic [20] and natural policy gradient [32]. Safety-constrained MDPs are studied in [114] and since the problem is inherently multichain, dynamic programming algorithms (which underline most of RL algorithms) cannot be used directly. This was demonstrated by a simple counterexample in [48, 71].

2-player Zero-sum Markov games were introduced by Shapely in the seminal paper [96] (he referred to them as Stochastic games). The essential difference between Markov games and repeated games is that in Markov games each player not only affects the

costs incurred by all players but also the transition to the next state. The concept of Markov games was extended to n-player Nonzero-sum Markov games in [34, 105]. Markov games with correlated equilibrium were studied in [100, 101]. However, unlike MDP's, one cannot use Linear programs even for 2-player Zero-sum Markov games [33]. Despite this negative result, Markov games have recently been extensively used as the underlying framework for Multi-agent Reinforcement Learning (MARL) [62]. Some of the earliest MARL algorithms were Minimax Q-learning [62], Friend-or-Foe Q-learning [63], Nash Q-learning [50] and Correlated Q-learning [42]. Despite these early successes in MARL algorithms, solving Nonzero sum Markov games remains an open challenge as pointed out in [116] (where it was observed that algorithms like Nash Q learning and Correlated Q learning were cycling between a mixture of policies instead of converging to an equilibrium policy) and more recently in the survey papers [49, 115]. Another class of games involving states is Differential Games (DG) introduced by Issacs in the 1960s [51] where each player's control actions affect the immediate cost as well as the underlying dynamics of the system [11, 12, 23, 94]. They can be seen as a generalization of the OCP to multiple controllers. Depending on the measurement feedback to the controllers, the solution concepts for DG are classified as open-loop solutions or feedback (or closed-loop) solutions. Computing feedback solutions such as (feedback Nash or feedback Stackelberg strategies) of DG's is challenging in general as the dynamics are perturbed due to any perturbation in the strategy of any player [23, 94]. Despite these fundamental challenges, researchers have made attempts to apply Machine learning and RL techniques for finding feedback solutions for DG's. However, as shown by the paper [69], gradient-based learning in even Linear Quadratic games almost surely misses the Nash equilibrium points.

2 Project Hypothesis and Research Objectives

The aim of this PhD project is to design control algorithms for large-scale systems with the following desirable properties:

- The controllers should be able to learn the optimal control strategies despite the presence of uncertainties in the system and the presence of other learning-based controllers.
- The controllers should be able to learn to coordinate amongst themselves with minimum interaction between themselves.
- The controllers should be able to learn to balance trade-offs between multiple objectives (such as safety and optimality).
- The control algorithms should be computationally efficient and easy to implement with minimal commissioning time and modifications to the existing system.

Since a large-scale system has underlying dynamics that will be affected by the control actions taken by all the controllers at any given time, DG's seems to be the most suitable mathematical framework for this project. However, as mentioned in the State-of-the-Art, DG's are notoriously difficult to solve in general owing to the presence of dynamical constraints (even when we have the perfect model of the system). Markov games offer relatively more hope (compared to DG's), for obtaining tractable control algorithms. However, as mentioned in the State-of-the-Art, algorithms for finding correlated equilibrium for Markov games can cycle between a mixture of policies instead of converging to an equilibrium policy. Thus, the central research question that this dissertation seeks to answer is as follows,

How can we design controllers for such systems involving states in a computationally efficient way with the aforementioned desirable properties?

One of the most celebrated results that unify OCP literature with RL and Game theory is the Bellman equations (as Hamilton-Jacobi-Bellman equations in OCP, Shapley equations in Markov games, and as Hamilton-Jacobi-Issacs equations in Differential games). They provide necessary and sufficient conditions for optimality [12, 33] and have the attractive feature of being able to break down a large-scale optimization problem into small sub-problems with optimal sub-structure thanks to Bellman's Principle of optimality. As DG's and Markov Games are computationally intractable in general, we hope to approximate the original problem in such a way that we can sequentially solve smaller optimization problems or static games. This forms the primary hypothesis behind this PhD project, which is stated next.

Hypothesis 2.1. *Optimal Control of large complex systems can be formulated as a sequence of optimization problems that need to be solved at each time instant by individual controllers in a decentralized setting. Furthermore, under suitable assumptions, the controllers can learn the desired solution that is robust to disturbances and can also satisfy multiple objectives such as safety and optimality.*

Based on the Hypothesis 2.1, we have formulated the following research objectives, that we seek to address during the PhD project. The research findings of the dissertation seek to fulfill the following research objectives and the research findings can be found in the associated papers and the introductory chapters.

Research Objective 1

Develop a method for discretizing differential games to facilitate obtaining a solution by sequentially solving static games.

This research objective is addressed in the Papers A and B. Chapter 3 of this dissertation provides a summary of the Markov chain Approximation (MCA) technique. We

consider a generic large-scale drinking water distribution network with two controllers and apply MCA to obtain the associated Markov chain (more specifically a Markov game in this case due to the presence of multiple controllers) at each time instant. As the controllers do not communicate and each controller has a different objective function, we have a nonzero-sum game. Since no efficient algorithms exist for finding a Nash equilibrium for general nonzero-sum games, we resort to the Minimax solution. However, as mentioned in the State-of-the-Art, there are no Linear programs but instead Bi-linear programs for solving 2-player Zero-sum Markov Games (this is discussed in the subsection 2.4). We approximate the associated Bi-linear program by combining the approximate Value iteration algorithm with a Linear program (see subsection 2.1 for more details). Essentially, we construct a sequence of smaller optimization problems that need to be solved at each time instant by each controller for convergence to the Minimax solution in a decentralized setting.

Research Objective 2

Develop a method for the controllers to learn an optimal policy in a decentralized setting despite the presence of disturbances and uncertainties in the environment

This research objective is primarily addressed in the Papers D and E and the associated Chapter 4 of the dissertation. We consider the steady-state operation of a generic large-scale drinking water distribution network. The reason for considering the steady state operation is because in practice the flow dynamics of the water distribution evolve on a faster time scale compared to the measurements (pressure) and therefore, we only consider the steady state of the system with stochastic consumption being treated as the disturbances to the system. Paper D deals with the practical implementation of decentralized control on the Smart Water Lab at Aalborg University. We emulate a practical water distribution network in the lab and each player solves a sequence of Linear programs corresponding to the Minimax solution. Each player constructs an approximate model of the water distribution network so as to obtain cost matrices for the aforementioned Linear program. The proposed decentralized Minimax solution for a 2-player nonzero-sum game (where players do not know the cost functions of their opponents) is compared to the standard centralized nonlinear program based solutions in Chapter 4 of this dissertation. This comparison is done for an arbitrary water distribution network and the simulation results show that the decentralized Minimax solution obtains a similar result to the centralized solver.

In contrast to Paper D, Paper E considers a model-free setting where the controllers do not have any prior knowledge about the water distribution network, nor do they know about the disturbances due to the consumption dynamics or the presence of other controllers. Another point of departure, compared to the previous papers is that we focus on the notion of Correlated Equilibrium instead of Minimax solution in Paper E. We formalize the aforementioned problem as a sequence of perturbed static games

that need to be solved at each time instant and introduce equilibrium notions for the same in Paper E. A learning algorithm that is robust to the effect of disturbances due to consumption dynamics is proposed in Paper E. The proof of convergence of the same required a modification of the existing notion of Approachability and Blackwell's Approachability theorem and this is discussed in Paper E.

Research Objective 3

Formulate a state-dependent multi-objective optimization problem as a constrained optimization problem and learn the best possible solution of the same

This research objective is addressed in the Paper C and Chapter 5 of the dissertation. Unfortunately, it turns out that Bellman's Equations do not hold for constrained Markov Decision process (in general cases) and one needs to solve a potentially large Linear program that considers all the states and control actions together as it solves the entire constrained MDP. The complexity of this Linear program grows as the number of states or actions increases. Furthermore, this approach is not amenable to iterative learning methods and therefore, we attempt to resolve this challenge by reformulating the problem as a 2-player zero-sum game and searching for subgame equilibrium.

3 Summary of Contributions

The summary of contributions of this dissertation are listed below as follows,

- In Paper E, the standard notion of Correlated equilibrium is found to be not robust to the unobserved perturbations or shocks to the costs (as shown by a simple example). We instead propose the stricter notion of Correlated Equilibrium referred to as Robust Correlated Equilibrium that is robust to time-varying shocks on the costs.
- In Paper E, an algorithm that can be seen as an extension to the Regret matching algorithm is proposed for learning the Robust Correlated Equilibrium in a decentralized setting. The convergence analysis of this algorithm required a modification to the celebrated Blackwell's Approachability Theorem for the considered case as the standard definition of Approachability is valid only for the time-average of costs.
- In Paper B, Markov Chain Approximation is used to approximate a nonzero-sum differential game as a corresponding Markov game. The associated Markov game is solved via a novel combination of value iteration and linear programming instead of solving a bilinear program or imposing restrictive assumptions on the controllers.

- In Paper A, a Monte Carlo simulation based algorithm is proposed as an alternative to Markov chain Approximation for estimating the value function of a stochastic optimal control problem.
- In Paper C, an algorithm is proposed for safety-constrained MDP's subject to random stopping times. Convergence of the algorithm is studied and minimal bounds on the run time for approximate convergence to the optimal Lagrangian are obtained.
- The mathematical programs for solving nonzero-sum games, involve solving complex centralized nonlinear programs that cannot be extended beyond 2-players. To that end, we propose to implement decentralized linear programs that solve a minimax problem corresponding to the original game in Paper D. The simulation results as well as the practical implementations in the lab, show that the Minimax solution is reasonably good as the desired set point is achieved by the controllers.

References

[1] J. Abernethy, P. L. Bartlett, and E. Hazan, "Blackwell approachability and no-regret learning are equivalent," in *Proceedings of the 24th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, 2011, pp. 27–46.

[2] A. Abraham and L. Jain, *Evolutionary multiobjective optimization*. Springer, 2005, edited by Evolutionary Multiobjective Optimization Springer London.

[3] E. Altman, *Constrained Markov decision processes: stochastic modeling*. Routledge, 1999.

[4] M. Anilkumar, N. Padhiyar, and K. Moudgalya, "Lexicographic optimization based mpc: Simulation and experimental study," *Computers & Chemical Engineering*, vol. 88, pp. 135–144, 2016.

[5] A. M. Annaswamy, K. H. Johansson, G. J. Pappas *et al.*, "Control for societal-scale challenges: Road map 2030," *IEEE Control Systems Society Publication: Piscataway, NJ, USA*, 2023.

[6] A. Antoniou and W.-S. Lu, *Practical optimization*. Springer, 2007.

[7] P. J. Antsaklis and K. M. Passino, *An introduction to intelligent and autonomous control*. Kluwer Academic Publishers, 1993.

[8] K. J. Aström, P. Albertos, M. Blanke, A. Isidori, W. Schaufelberger, and R. Sanz, *Control of complex systems*. Springer Science & Business Media, 2011.

[9] R. J. Aumann, "Subjectivity and correlation in randomized strategies," *Journal of mathematical Economics*, vol. 1, no. 1, pp. 67–96, 1974.

[10] ——, "Correlated equilibrium as an expression of bayesian rationality," *Econometrica: Journal of the Econometric Society*, pp. 1–18, 1987.

[11] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1998.

[12] T. Başar and G. Zaccour, *Handbook of dynamic game theory*. Springer, 2018.

[13] G. Belgioioso, P. Yi, S. Grammatico, and L. Pavel, "Distributed generalized nash equilibrium seeking: An operator-theoretic perspective," *IEEE Control Systems Magazine*, vol. 42, no. 4, pp. 87–102, 2022.

[14] R. Bellman, "Dynamic programming," *Princeton University Press, New Jersey*, 1957.

[15] M. Benaım and M. W. Hirsch, "Mixed equilibria and dynamical systems arising from fictitious play in perturbed games," *Games and Economic Behavior*, vol. 29, no. 1-2, pp. 36–72, 1999.

[16] D. Bertsekas, *Reinforcement learning and optimal control*. Athena Scientific, 2019.

[17] D. P. Bertsekas, *Abstract dynamic programming*. Athena Scientific, 2022.

[18] D. Blackwell, "An analog of the minimax theorem for vector payoffs." *Pacific Journal of Mathematics*, vol. 6, no. 4, pp. 1–8, 1956.

[19] E. Borel, "Game theory and the integral equations of its symmetric kernel," *Reports of the Academy of Sciences*, vol. 173, no. 1304-1308, p. 58, 1921.

[20] V. S. Borkar, "An actor-critic algorithm for constrained markov decision processes," *Systems & control letters*, vol. 54, no. 3, pp. 207–213, 2005.

[21] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[22] S. Boyd and L. Vandenberghe, "Convex optimization, cambridge univ," *Press, UK*, 2004.

[23] A. Bressan, "Noncooperative differential games," *Milan Journal of Mathematics*, vol. 79, no. 2, pp. 357–427, 2011.

[24] G. W. Brown, "Iterative solution of games by fictitious play," *Act. Anal. Prod Allocation*, vol. 13, no. 1, p. 374, 1951.

[25] E. F. Camacho and C. Bordons, *Model predictive control. 2. ed.* London: Springer, 2004.

[26] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[27] P. Chakraborty, "Optimization and control of flexible demand and renewable supply in a smart power grid," Ph.D. dissertation, University of Florida, 2016.

[28] S. Chen, Z. Wu, and P. D. Christofides, "Cyber-security of centralized, decentralized, and distributed control-detector architectures for nonlinear processes," *Chemical Engineering Research and Design*, vol. 165, pp. 25–39, 2021.

[29] A. A. Cournot, *Researches into the Mathematical Principles of the Theory of Wealth [Original published title: Recherches sur les Principes Mathématiques de la Théorie des Richesses]*. New York: Macmillan Company, 1927 [c1897], 1897.

[30] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou, "The complexity of computing a nash equilibrium," *Communications of the ACM*, vol. 52, no. 2, pp. 89–97, 2009.

[31] K. Deb, "Multi-objective optimization," in *Search methodologies*. Boston, MA: Springer, 2014, pp. 403–449.

[32] D. Ding, K. Zhang, T. Basar, and M. Jovanovic, "Natural policy gradient primal-dual method for constrained markov decision processes," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8378–8390, 2020.

[33] J. Filar and K. Vrieze, *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

[34] A. M. Fink, "Equilibrium in a stochastic n-person game," *Journal of science of the hiroshima university, series ai (mathematics)*, vol. 28, no. 1, pp. 89–93, 1964.

[35] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multiobjective optimization: Formulation discussion and generalization," *Icga*, vol. 93, pp. 416–423, July 1993.

[36] D. P. Foster and R. V. Vohra, "Asymptotic calibration," *Biometrika*, vol. 85, no. 2, pp. 379–390, 1998.

[37] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and J. D. Powell, *Feedback control of dynamic systems*. Prentice hall Upper Saddle River, 2002, vol. 4.

[38] D. Fudenberg and D. K. Levine, *The theory of learning in games*. MIT Press, 1998, vol. 2.

[39] G. Gambarelli and G. Owen, "The coming of game theory," *Essays in Cooperative Games: In Honor of Guillermo Owen*, pp. 1–18, 2004.

[40] S. J. Gould and R. C. Lewontin, "The spandrels of san marco and the panglossian paradigm: a critique of the adaptationist programme," *Proceedings of the royal society of London. Series B. Biological Sciences*, vol. 205, no. 1161, pp. 581–598, 1979.

[41] D. Grass *et al.* (2008) Optimal control of nonlinear processes: With applications in drugs, corruption, and terror. Berlin, Heidelberg: Springer Berlin Heidelberg.

[42] A. Greenwald, K. Hall, and R. Serrano, "Correlated q-learning," in *ICML*, vol. 3, 2003, pp. 242–249.

[43] A. Greenwald and A. Jafari, "A general class of no-regret learning algorithms and game-theoretic equilibria," in *COLT*, vol. 3, 2003, pp. 2–12.

[44] J. Hannan, "Approximation to bayes risk in repeated play," *Contributions to the Theory of Games*, vol. 3, pp. 97–139, 1957.

[45] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[46] ——, "Stochastic uncoupled dynamics and nash equilibrium," *Games and economic behavior*, vol. 57, no. 2, pp. 286–303, 2006.

[47] ——, *Simple adaptive strategies: from regret-matching to uncoupled dynamics*. World Scientific, 2013, vol. 4.

[48] M. Haviv, "On constrained markov decision processes," *Operations research letters*, vol. 19, no. 1, pp. 25–28, 1996.

[49] P. Hernandez-Leal, B. Kartal, and M. E. Taylor, "A survey and critique of multiagent deep reinforcement learning," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 6, pp. 750–797, 2019.

[50] J. Hu and M. P. Wellman, "Nash q-learning for general-sum stochastic games," *Journal of machine learning research*, vol. 4, no. Nov, pp. 1039–1069, 2003.

[51] R. Isaacs, *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization*. Courier Corporation, 1965.

[52] J. S. Jordan, "Three problems in learning mixed-strategy nash equilibria," *Games and Economic Behavior*, vol. 5, no. 3, pp. 368–386, 1993.

[53] L. Kallenberg, "Markov decision processes," *Lecture Notes. University of Leiden*, vol. 428, 2011.

[54] O. Karaca, P. G. Sessa, A. Leidi, and M. Kamgarpour, "No-regret learning from partially observed data in repeated auctions," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14–19, 2020.

[55] E. C. Kerrigan and J. M. Maciejowski, "Designing model predictive controllers with prioritised constraints and objectives," in *Proceedings. IEEE International Symposium on Computer Aided Control System Design. IEEE, 2002*, 2002.

[56] H. K. Khalil, *Nonlinear control*. Pearson New York, 2015.

[57] M. Kojima and M. Muramatsu, "A note on sparse sos and sdp relaxations for polynomial optimization problems over symmetric cones," *Computational Optimization and Applications*, vol. 42, no. 1, pp. 31–41, 2009.

[58] V. R. Konda and V. S. Borkar, "Actor-critic–type learning algorithms for markov decision processes," *SIAM Journal on control and Optimization*, vol. 38, no. 1, pp. 94–123, 1999.

[59] H. J. Kushner and P. G. Dupuis, *Numerical methods for stochastic control problems in continuous time*. Springer Science & Business Media, 2001, vol. 24.

[60] J.-B. Lasserre, *Moments, positive polynomials and their applications*. World Scientific, 2010, vol. 1.

[61] M. Laurent, "Sums of squares, moment matrices and optimization over polynomials," in *Emerging applications of algebraic geometry*. New York, NY: Springer, 2009, pp. 157–270.

[62] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Machine learning proceedings 1994*. Elsevier, 1994, pp. 157–163.

[63] M. L. Littman *et al.*, "Friend-or-foe q-learning in general-sum games," in *ICML*, vol. 1, no. 2001, 2001, pp. 322–328.

[64] J. M. Maciejowski, *Predictive control with constraints*. Harlow: Prentice Hall, 2002.

[65] A. B. MacKenzie and L. A. DaSilva, *Game theory for wireless engineers*. Springer Nature, 2022.

[66] S. Mahadevan, "Optimality criteria in reinforcement learning," in *Proceedings of the AAAI Fall Symposium on Learning Complex Behaviors in Adaptive Intelligent Systems*. Citeseer, 1996.

[67] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization*, vol. 26, no. 6, pp. 369–395, 2004.

[68] V. K. Mathur, "How well do we know pareto optimality?" *The Journal of Economic Education*, vol. 22, no. 2, pp. 172–178, 1991.

[69] E. Mazumdar, L. J. Ratliff, and S. S. Sastry, "On gradient-based learning in continuous games," *SIAM Journal on Mathematics of Data Science*, vol. 2, no. 1, pp. 103–131, 2020.

[70] K. Miettinen, *Nonlinear multiobjective optimization*. Springer Science & Business Media, 2012, vol. 12.

[71] R. Misra, R. Wisniewski, and C. S. Kallesøe, "On bellman's principle of optimality and reinforcement learning for safety-constrained markov decision process," *arXiv preprint arXiv:2302.13152*, 2023.

[72] R. Misra, R. Wisniewski, and Ö. Karabacak, "Sum-of-squares based computation of a lyapunov function for proving stability of a satellite with electromagnetic actuation," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 7380–7385, 2020.

[73] J. Nash, "Non-cooperative games," *Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951. [Online]. Available: http://www.jstor.org/stable/1969529

[74] ——, "Two-person cooperative games," *Econometrica: Journal of the Econometric Society*, pp. 128–140, 1953.

[75] J. F. Nash Jr, "The bargaining problem," *Econometrica: Journal of the econometric society*, pp. 155–162, 1950.

[76] ——, "Equilibrium points in n-person games," *Proceedings of the national academy of sciences*, vol. 36, no. 1, pp. 48–49, 1950.

[77] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press, 2007.

[78] C. Ocampo-Martinez, D. Barcelli, V. Puig, and A. Bemporad, "Hierarchical and decentralised model predictive control of drinking water networks: Application to barcelona case study," *IET control theory & applications*, vol. 6, no. 1, pp. 62–71, 2012.

[79] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[80] A. Papachristodoulou and S. Prajna, "A tutorial on sum of squares techniques for systems analysis," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005, pp. 2686–2700.

[81] G. A. Parker and J. M. Smith, "Optimality theory in evolutionary biology," *Nature*, vol. 348, no. 6296, pp. 27–33, 1990.

[82] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, 2000.

[83] T. Parthasarathy, "On games over the unit square," *SIAM Journal on Applied Mathematics*, vol. 19, no. 2, pp. 473–476, 1970.

[84] V. Perchet, "Approachability, regret and calibration: Implications and equivalences," *Journal of Dynamics and Games*, vol. 1, no. 2, pp. 181–254, 2014.

[85] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR computational mathematics and mathematical physics*, vol. 4, no. 5, pp. 1–17, 1964.

[86] S. Prajna, "Barrier certificates for nonlinear model validation," *Automatica*, vol. 42, no. 1, pp. 117–126, 2006.

[87] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 1994.

[88] J. Robinson, "An iterative method of solving a game," *Annals of mathematics*, pp. 296–301, 1951.

[89] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica: Journal of the Econometric Society*, pp. 520–534, 1965.

[90] C. Scherer, P. Gahinet, and M. Chilali, "Multiobjective output-feedback control via lmi optimization," *IEEE Transactions on automatic control*, vol. 42, no. 7, pp. 896–911, 1997.

[91] C. Scherer and S. Weiland, "Linear matrix inequalities in control," Tech. Rep., 2000.

[92] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[93] S. P. Sethi and G. L. Thompson. (2000) Optimal control theory applications to management science and economics. New York, NY: Springer US.

[94] S. P. Sethi, "Differential games," in *Optimal Control Theory*. Springer, 2019, pp. 385–407.

[95] L. Shapley, "Some topics in two-person games," *Advances in game theory*, vol. 52, pp. 1–29, 1964.

[96] L. S. Shapley, "Stochastic games," *Proceedings of the national academy of sciences*, vol. 39, no. 10, pp. 1095–1100, 1953.

[97] Y. Shoham, K. Leyton-Brown *et al.*, "Multiagent systems," *Algorithmic, Game-Theoretic, and Logical Foundations*, 2009.

[98] S. Singh, T. Jaakkola, M. L. Littman, and C. Szepesvári, "Convergence results for single-step on-policy reinforcement-learning algorithms," *Machine learning*, vol. 38, pp. 287–308, 2000.

[99] S. Siniscalchi-Minna, F. D. Bianchi, and C. Ocampo-Martinez, "Predictive control of wind farms based on lexicographic minimizers for power reserve maximization," in *2018 Annual American Control Conference (ACC). IEEE*, 2018.

[100] E. Solan, "Characterization of correlated equilibria in stochastic games," *International Journal of Game Theory*, vol. 30, pp. 259–277, 2001.

[101] E. Solan and N. Vieille, "Correlated equilibrium in stochastic games," *Games and Economic Behavior*, vol. 38, no. 2, pp. 362–399, 2002.

[102] J. Stoustrup, "Plug & play control: Control technology towards new challenges," *European Journal of Control*, vol. 15, no. 3-4, pp. 311–330, 2009.

[103] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[104] C. Szepesvári, *Algorithms for reinforcement learning*. Springer Nature, 2022.

[105] M. Takahashi, "Equilibrium points of stochastic non-cooperative n-person games," *Journal of Science of the Hiroshima University, Series AI (Mathematics)*, vol. 28, no. 1, pp. 95–99, 1964.

[106] T. Tatarenko and M. Kamgarpour, "Learning nash equilibria in monotone games," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 3104–3109.

[107] ——, "Bandit learning in convex non-strictly monotone games," *arXiv preprint arXiv:2009.04258*, 2023.

[108] J. Von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton university press, 1944.

[109] H. Von Stackelberg, "Marktform and gleichgewicht springer-verlag," 1934.

[110] H. Waki, S. Kim, M. Kojima, and M. Muramatsu, "Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity," *SIAM Journal on Optimization*, vol. 17, no. 1, pp. 218–242, 2006.

[111] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[112] A. Wilson, *Lyapunov arguments in optimization*. University of California, Berkeley, 2018.

[113] R. Wisniewski and L.-M. Bujorianu, "Safety of stochastic systems: An analytic and computational approach," *Automatica*, vol. 133, p. 109839, 2021.

[114] R. Wisniewski and M. L. Bujorianu, "Probabilistic safety guarantees for markov decision processes," *IEEE Transactions on Automatic Control*, 2023.

[115] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *arXiv preprint arXiv:1911.10635*, pp. 321–384, 2019.

[116] M. Zinkevich, A. Greenwald, and M. Littman, "Cyclic equilibria in markov games," *Advances in neural information processing systems*, vol. 18, 2005.

Chapter 2

Game theory and Reinforcement Learning

Summary *This chapter provides an overview of fundamental concepts related to Game Theory—specifically focusing on Equilibrium concepts, Repeated games, Markov games, and computational methods—as well as their connections to Reinforcement Learning within the game-theoretic framework described. While this chapter is not intended to comprehensively cover every aspect of game theory and reinforcement learning, it does address the primary topics relevant to this dissertation, focusing on those directly utilized or investigated during the research period. In this dissertation, game theory is studied from the perspective of decentralized control. We motivate the application of solution concepts by first formulating a decentralized optimization problem which we hope to solve by using solution concepts from game theory (such as Nash equilibrium). We discuss solution concepts such as Minimax solution, Nash equilibrium, Best response solution, Correlated equilibrium, and Coarse correlated equilibrium. The minimax solution concept is used for designing decentralized controls for a water distribution network in Paper D [55]. We also introduce Robust Correlated Equilibrium that is one of the main contributions of this dissertation in Paper E [58]. Thereafter, we introduce the notion of repeated games that formalize learning in this setting and introduce the notion of Approachability and the related Blackwell's Approachability theorem that was used in Paper E. We also introduce the concept of Stochastic (or Markov) games that extend repeated games with the notion of state. Computation procedures for solving Static games, as well as Markov games, are summarized as well including the Linear program-based solution for Static games used in Paper D.*

In the following sections, we explore the related mathematical framework for Markov games and highlight that these games lack the ordered field property, rendering linear

programming alone inadequate. We delve into the origin of the nonlinearity (specifically bi-linearity) and suggest a sequence of approximating Linear programs. This method is utilized in Paper B [56]. Additionally, we examine dynamic programming techniques (specifically those based on Shapley's equation), such as the Hoffman-Karp Algorithm and the Pollatschek-Avi-Itzhak Algorithm. A modified Q-learning version of the Hoffman-Karp algorithm is used in Paper C [57]. Finally, we introduce learning algorithms like fictitious play and regret matching for repeated games, followed by an overview of value iteration-based reinforcement learning for Markov games.

1 Notation

- For a finite set A, $\mid A \mid$ represents the total number of elements in A.
- $\mathbb{P}[(\cdot)]$ represents the probability of event $(\cdot)$ occurring.
- $\mathbb{E}_{(\cdot)}$ represents the expectation operator corresponding to the probability measure $(\cdot)$ indicated in the subscript.
- $\Delta(A)$ represents a probability simplex in $\mathbb{R}^{|A|}$ where A is a finite set. Formally

$$\Delta(A) := \left\{ x \in \mathbb{R}_+^{|A|} : \sum_{a \in A} x(a) = 1 \right\}.$$

- $\mathbf{a} \in \mathbb{R}^m$ represents a vector of reals with components $(a_1, \cdots, a_m)$.
- $[a]_+$ represents positive part of the real number a i.e. $[a]_+ = \max\{a, 0\}$ and for a vector of reals $\mathbf{a} = (a_1, \cdots, a_m)$, represents $([a_1]_+, \cdots, [a_m]_+)$ and similarly for a matrix A of reals, $[A]_+$ represents matrix consisting of entries $[a_{ij}]_+$.
- $\|\mathbf{a}\|_p$ denotes the p-norm of the vector $\mathbf{a} \in \mathbb{R}^n$. If no subscript is present it represents the 2-norm.
- For two vectors $\mathbf{a}$ and $\mathbf{b}$, $\mathbf{a} \cdot \mathbf{b}$ represents the scalar product between them.
- $\mathbf{I}_{n \times n}$ represents a $n \times n$ Identity matrix and $\mathbf{1}_n$ is a vector (dimension n) of 1's.

2 Game theory

Game theory is a collection of mathematical frameworks, used for modeling multiple decision-makers wherein the cost/reward (or utility) of each decision-maker is affected by the decisions taken by all decision-makers i.e. the costs of all decision-makers are coupled. Depending on decision times, and the presence of states or decision nodes,

common game theoretic models are Static Games (or 1-shot games), Repeated games, Bayesian Games, Markov Games (or Stochastic games), Extensive form games, Differential games, Evolutionary Games, Mean field games and many more. Games can also be classified based on the information available to each decision-maker: Full-information games, Partial Information Games, Unknown games, and Leader-Follower games (or Stackelberg Games) [7]. They can also be classified as finite-dimensional or infinite-dimensional based on whether the decision-makers have finite control actions or a continuum of control actions. Games with a continuum of decision-makers are analyzed in Mean Field Game theory [48].

Remark 2.1. *The terms decision-maker(s), agent(s), and player(s) represent the controller(s). The terms action(s) and pure strategy(s) represent the finite control action(s) available to the controller(s). The terms mixed strategy and randomized strategy are also equivalent, meaning that the control strategy is a probability distribution over finite control actions.*

2.1 Static and Perturbed games

Consider a game with finite players, where U^i is the set of m finite control actions available to player i. A Static Game is defined by a tuple $\Gamma := (N, U, C)$, where

- N is the finite number of players,
- $U := \prod_{i \in N} U^i := U^1 \times U^2 \times \cdots U^N$ is the combined control input set,
- $C := \{c^1, c^2, \cdots, c^N\}$ is the set of costs incurred by the players (indexed by superscripts), where $c^i : U \mapsto \mathbb{R}$ i.e. cost incurred by player i is dependent on combined control input by all the players.

We define the product set U^{-i} as

$$U^{-i} := \prod_{j \in \{1, \cdots, N\}, j \neq i} U^j,$$

which represents the joint control action space of all players except player i. In the sequel, $u \in U$ will represent a joint control action i.e. $u = (u^i, u^{-i})$. Each decision-maker solves the following optimization problem independent of other decision-makers

$$\min_{u^i} \; c^i\left(u^i, u^{-i}\right) \tag{2.1a}$$

$$\text{s.t.} \quad u^i \in U^i \tag{2.1b}$$

Since the variables u^{-i} are unknown (ahead of decision-making time), the optimization problem (2.1) is ill-posed as of now. Since $u^{-i} \in U^{-i}$ is a set of finite control actions, any reasonable solution should be parameterized by u^{-i}. Before discussing the solutions offered by game theory, we state the following underlying assumption on all players.

Assumption 2.1 (Rationality) Each decision-maker aims to solve (2.1) and can use any level of sophistication in their strategies.

Thus the players cannot take vindictive actions against other players if it leads to higher costs for the player taking a vindictive control action. Refinements of this assumption such as Bounded Rationality (i.e. Rational with a bounded computational complexity of strategies) are discussed in [76]. The solution concepts offered by game theory are typically equilibrium solutions in the sense that for any player i, equilibrium is attained when u^i is fixed (or more generally the control policy π^i dictating the selection of u^i is fixed) and deviation from that would only incur additional costs for player i. This leads us to the definition of Pure strategy Nash Equilibrium, where Pure implies deterministic.

Definition 2.1 (Pure Strategy Nash Equilibrium [90]) A joint pure strategy profile $u^\star := (u^{i\star}, u^{-i\star})$ is called a Pure Strategy Nash Equilibrium for the game Γ if, for every player i, the pure strategy $u^{i\star}$ in the joint pure strategy profile belongs to the set defined by the following inequality,

$$\{u^{i\star} \in U^i : c^i(u^{i\star}, u^{-i\star}) \leq c^i(\tilde{u}^i, u^{-i\star}), \quad \forall i, \tilde{u}^i\}, \tag{2.2}$$

where $\tilde{u}^i \in U^i$ is any possible unilateral (i.e. without forming coalitions or agreements with other players) deviation by player i from the joint pure strategy profile $u^\star$.

Intuitively, an equilibrium condition implies that no player has any incentive to deviate unilaterally from the equilibrium strategy profile, and thus, once the players learn or somehow compute the equilibrium strategies the joint profile rests or stays at the equilibrium. Nash was the first to characterize such equilibrium points in [61] and thus, such solutions are named Nash equilibrium in his honor. Since the seminal work on game theory [90], it is well known that an equilibrium solution to (2.1) satisfying (2.2) may not exist.

2.1 Example (Game of Rock-Paper-Scissor [47]) Consider the classic Rock-Paper-Scissor Game with payoffs defined by the following matrix

		Player B		
		Rock	*Paper*	*Scissor*
Player A	*Rock*	$(0,0)$	$(-1,1)$	$(1,-1)$
	Paper	$(1,-1)$	$(0,0)$	$(-1,1)$
	Scissor	$(-1,1)$	$(1,-1)$	$(0,0)$

This is a 2 player game where each player has three actions *Rock*, *Paper*, and *Scissor* i.e. $U^1 = U^2 = \{Rock, Paper, Scissor\}$. Player A is the row player and Player B is the column player. The elements of the payoff matrix indicate the payoff received due to the joint action taken by the row and column player. The first entry is the payoff received by Player A and the second entry is the Payoff received by Player B. Each player would like to maximize their payoffs. Observe that no matter what action is taken by player A, Player B can always choose an action that increases his payoff.

In the Paper [60], Nash showed that while an equilibrium solution may not exist for the game Γ in the space of finite control actions (i.e. there may not exist a Pure Strategy Nash Equilibrium), an equilibrium solution for the game Γ always exists in the space of Mixed Strategies. A mixed strategy for a player $i \in \{1, \cdots, N\}$ is a probability distribution $\pi^i \in \Delta(U^i)$ over player i's pure control actions U^i. They can also be interpreted as a convex combination of pure strategies. A joint distribution (of mixed strategies) is a probability distribution $\pi \in \Delta(U)$ over the joint pure control actions U taken by all the players. Given the joint distribution π, the marginal distribution $\pi^i \in \Delta(U^i)$ can be written as,

$$\pi^i(u^i) = \sum_{u^{-i} \in U^{-i}} \pi(u^i, u^{-i}), \quad \forall u^i \in U^i,$$

and the marginal distribution $\pi^{-i} \in \Delta(U^{-i})$ can be written as,

$$\pi^{-i}(u^{-i}) = \sum_{u^i \in U^i} \pi(u^i, u^{-i}), \quad \forall u^{-i} \in U^{-i}.$$

Recall that $u = (u^i, u^{-i})$ and the domain of cost functions is extended multilinearly from joint control action space U to the joint probability distribution $\Delta(U)$ as follows,

$$c(\pi(u)) := \mathbb{E}_\pi[c(u^i, u^{-i})] = \sum_{u \in U} \pi(u)c(u) = \sum_{u^i \in U^i} \sum_{u^{-i} \in U^{-i}} \pi^i(u^i)c(u^i, u^{-i})\pi^{-i}(u^{-i}).$$

Similarly, we can write the costs associated with the marginal distributions π^i or π^{-i}.

Definition 2.2 (Mixed Strategy Nash Equilibrium [60]) A probability distribution $\pi^\star(u)$ consisting of product of independent marginal distributions i.e. $\pi^\star(u) := \pi^{1\star} \times \cdots \times \pi^{N\star}$ is a Mixed Strategy Nash equilibrium of the game Γ if for any player i, the marginal distribution $\pi^{i\star}$ belongs to the following set

$$\left\{\pi^{i\star}(u^i) \in \Delta(U^i) : c^i(\pi^\star(u)) \leq c^i(\tilde{\pi}(u))\right\}, \tag{2.3}$$

where $\tilde{\pi} = \pi^{1\star} \times \cdots \times \tilde{\pi}^i \times \cdots \times \pi^{N\star}$ is any possible unilateral deviation by player i from the product probability distribution $\pi^\star$.

Thus once the players have reached an equilibrium strategy, they have no incentive to deviate from the equilibrium point. The aforementioned solution concepts are the most commonly used in game theory. However, besides these solution concepts, several other solutions have also been proposed and a non-exhaustive list is as follows.

- Minimax solution (or Security solution)
- Best response solution
- Hannan Consistent (Coarse Correlated Equilibrium)
- Correlated Equilibrium
- Robust Correlated Equilibrium

Minimax Solution

Consider the game Γ, a scalar V, and a player i, then a Minimax solution (or Security Solution) is an independent strategy for i that guarantees the worst-case cost to be at most V. This solution concept provides a worst-case solution to the decentralized optimization problem (2.1) provided that player i knows the possible control actions available to the other players (not to be confused with the actual control action taken by the opponent) and can evaluate the corresponding costs c^i. For readers familiar with classical control theory, the Minimax solution corresponds to the familiar Robust Control [6] and robust MPC formulation [17, 52].

Definition 2.3 (Minimax Solution [6, 90]) A probability distribution $\pi^{i\star}$ is the Minimax solution of the game Γ for player i, if $\pi^{i\star}$ belongs to the following set

$$\left\{\pi^{i\star}(u^i) \in \Delta(U^i) : c(\pi^{i\star}(u^i), u^{-i}) \leq V \leq c(\tilde{\pi}^i(\tilde{u}^i), u^{-i}),\ \forall u^{-i} \in U^{-i}\right\}, \tag{2.4}$$

where $\tilde{\pi}^i(\tilde{u}^i)$ is any possible unilateral deviation by player i from the distribution $\pi^{i\star}$.

A generalization of the Minimax solution to continuous action spaces can be found in [84] and is stated as follows,

Theorem 2.1 (Sion's Minimax theorem [84]).
Given compact and convex sets $U^1 \subset \mathbb{R}^n$, $U^2 \subset \mathbb{R}^m$, and a (semi) continuous function $c : U^1 \times U^2 \mapsto \mathbb{R}$ that is (quasi) convex for any $u^1 \in U^1$ and (quasi) concave for any $u^2 \in U^2$, we have

$$\inf_{u^1 \in U^1} \sup_{u^2 \in U^2} c(u^1, u^2) = \sup_{u^2 \in U^2} \inf_{u^1 \in U^1} c(u^1, u^2). \tag{2.5}$$

For Nonzero-sum games, a Minimax solution may not correspond to an equilibrium solution unless the Nonzero-Sum game is strategically equivalent to a Zero-Sum game. Informally, the concept of strategic equivalence states that: two games are strategically equivalent if the cost function of one game can be obtained from the other game via only linear transformations. A more detailed overview of strategic equivalence can be found in [5, 6, 59].

Best Response Solution

The Best Response solution as the name suggests is the optimal response to the given opponent's strategy (can be a mixed strategy or a pure strategy). More specifically the Best response for player i is a mapping from the opponent's joint strategy to a distribution over player i's actions and is defined as follows.

Definition 2.4 (Best Response Solution [7, 77]) The Best response solution, $BR^i(\pi^{-i}) : \pi^{-i} \mapsto \Delta(U^i)$ is a distribution $\pi^{i\star}$ that belongs to the following set.

$$\left\{\pi^{i\star}(u^i) \in \Delta(U^i) : c(\pi^{i\star}(u^i), \pi^{-i}(u^{-i})) \leq c(\tilde{\pi}^i(\tilde{u}^i), \pi^{-i}(u^{-i}))\right\}, \tag{2.6}$$

where $\tilde{\pi}^i$ is a unilateral deviation by player i from the best response solution $\pi^{i\star}$.

Since the opponent's strategy is usually not known to other players, the best response mapping is used to map an estimate of the opponent's strategy to the optimal distribution. This is the basic idea behind learning algorithms Fictitious play [13, 29] and calibrated learning [27]. In a two-player zero-sum game, if both players play the best response to their opponent's strategies (or empirical frequencies à la fictitious play) then their strategies converge to a Mixed Strategy Nash Equilibrium (or a Minimax Solution). A closely related solution concept is the so-called 'better response' that performs gradient descent based on the observed empirical frequencies of the opponent [25, 77].

Correlated Equilibrium and Coarse Correlated Equilibrium

The solution concepts discussed so far robustify the player's strategy to the worst-case outcome and for zero-sum games, this leads to an "optimal" solution (given a rational opponent). However, nonzero-sum or general-sum games better represent most real-life situations involving multiple decision-makers, and analyzing these games as zero-sum games introduces a well-known cognitive bias called zero-sum thinking [75] that results in solutions that are inferior to Pareto Optimal solutions [73, 74]. Furthermore, all general-sum games have multiple Nash equilibrium [6], and it is not clear which equilibrium point will be chosen by the players and this invalidates any possible use of fixed-point algorithms. Another related concern is ensuring that the players coordinate or correlate their strategies to reach the same equilibrium point. Lastly, the aforementioned solution concepts are incompatible with the Bayesian view on subjective probabilities i.e. players may choose to deviate from an equilibrium point after observing the outcome [4].

2.2 Example (Game of chicken [3]) Consider the following Traffic Intersection Game (sometimes called Game of Chicken) with payoffs defined by the following matrix

		Player B	
		$Wait$	Go
Player A	$Wait$	$(0, 0)$	$(0, 1)$
	Go	$(1, 0)$	$(-100, -100)$

The game consists of two car drivers who reach an intersection on the road from opposite sides. The choices available for them are whether to wait for the other driver to pass or to go and hope that the other diver waits. Both players aim to maximize their payoffs. As indicated by the payoff matrix if they collide they both incur a very negative payoff that represents a potentially catastrophic accident for both of them. This game has two pure strategy Nash equilibrium points: $(Wait, Go)$ and $(Go, Wait)$ where the first action is player A's choice and the second action is player B's choice. Notice that the payoff for one of the players will be 0 in both cases. There also exists a mixed strategy Nash equilibrium point where player A plays action $Wait$ with probability ψ and action Go with probability $1-\psi$. At equilibrium, player B should have no incentive to switch actions which implies that player B's payoff should be the same for both actions i.e.

$$\psi(1) + (1 - \psi)(-100) = 0 \implies \psi = \frac{100}{101}.$$

By symmetricity of the payoff matrix, similar arguments apply for player B as well. Thus both the players get an expected payoff of 0 while also risking a catastrophic accident. From the payoff matrix, it is evident that a better outcome for both players can be if they can coordinate their action profiles such that they rule out the two

"bad" outcomes where their actions are the same and instead randomize uniformly over the "good" outcomes. Specifically choosing $\psi = 0.5$ for the outcome $(Wait, Go)$ and $\psi = 0.5$ for the outcome $(Go, Wait)$.

Note that in the above example, the joint distribution $\psi = (0.5, 0.5)$ over the two outcomes, $(Wait, Go)$ and $(Go, Wait)$ does not exist in the space of Mixed strategy Nash equilibrium despite satisfying the Rationality Assumption. This led Aumann to introduce the concept of Correlated Equilibrium where players assign subjective probabilities to outcomes [3, 4] and is defined as follows. Similar to the presentation in [26], we first introduce the correlating device and then the equilibrium condition.

Definition 2.5 (Correlating Device [26]) Consider a finite set of correlating signals U, a partition P^i of U for every player $i \in \{1, \cdots, N\}$ and a probability simplex over the finite set of correlating signals denoted by $\pi^\star \in \Delta(U)$. Then a correlating device is defined by the tuple $\mathcal{CD} = (U, (P^i)_{i\in\{1,\cdots,N\}}, \pi^\star)$.

In the sequel, we shall work with the conditional distribution $\pi^\star(u \mid P^i(u))$ and the expectation $c(\pi^\star(u \mid P^i(u)))$ will refer to the standard conditional expectation i.e. expectation with respect to the conditional distribution $\pi^\star(u \mid P^i(u))$.

Definition 2.6 (Correlated Equilibrium [26]) Given a correlating device $\mathcal{CD}$, a distribution $\pi^\star \in \Delta(U)$ conditioned on $P^i(u)$ is a correlated equilibrium of the game Γ if for every player i, $\pi^\star$ belongs to the following set

$$\big\{\pi^\star(u \mid P^i(u)) \in \Delta(U) : c(\pi^\star(u^i, u^{-i} \mid P^i(u))) \leq c(\pi^\star(\tilde{u}^i, u^{-i} \mid P^i(u))),\ \forall u^i, \tilde{u}^i \in U^i, \text{ where } u^i \sim \pi^\star(u \mid P^i(u))\big\}. \quad (2.7)$$

Correlated Equilibrium can be interpreted as a generalization of Mixed strategy Nash Equilibrium since it considers a distribution over joint control action space (instead of independent product distributions of control actions for each player). Thus, all Nash equilibria are Correlated equilibria but the reverse is not true. Referring to the previous Traffic Intersection game, Correlated Equilibrium can also be considered as a convex combination of Nash equilibrium points where both players receive an independent random signal before the beginning of the game. Based on this signal both players choose one of the outcomes $(Wait, Go)$ or $(Go, Wait)$. An example of this signal can be a traffic light that can be either Red or Green. Suppose player A sees a Red signal, then it can

be concluded by player A that player B must have seen a Green signal, and therefore the optimal action is to $Wait$. Symmetrical arguments apply to player B. It should be noted that the signal received by both players need not be a traffic light, it can be something arbitrary such as whether the sky is Sunny or Cloudy. The correlating device can be implicitly found by players (such as the history of play or the empirical frequency of the states visited) as in [35]. Note that Correlated Equilibrium (2.7) is robust to any conditional unilateral deviation by player i after sampling $u^i \sim \pi^\star(u \mid P^i(u))$ where the unilateral deviation is conditioned on what player i infers about the strategy of other players based on his/her private recommendation. A relaxation of Correlated Equilibrium which guarantees robustness only to unilateral (unconditional) deviations by player i is the Coarse Correlated Equilibrium of the game Γ [59] or the Hannan set [31] of the game Γ and is defined as follows.

Definition 2.7 (Coarse Correlated Equilibrium [73]) A joint distribution $\pi^\star$ is the coarse correlated equilibrium of the game Γ if for every player i, $\pi^\star$ belongs to the following set

$$\left\{\pi^\star(u) \in \Delta(U) : c^i(\pi^\star(u^i, u^{-i})) \leq c^i(\pi^\star(\tilde{u}^i, u^{-i})), \quad \forall u^i, \tilde{u}^i \in U^i\right\} \tag{2.8}$$

Unlike the definition of Correlated Equilibrium, Coarse Correlated Equilibrium is defined in terms of expectations (and not conditional expectations) with respect to the joint distribution $\pi^\star(u)$.

Perturbed Games and Robust Correlated Equilibrium [58]

We now consider an extension of Static Games defined by Γ where we introduce perturbations on the costs incurred by the players due to exogenous disturbances. As the scope of the work so far is finite games, we consider the possible perturbations due to exogenous disturbances to be a finite number and we shall refer to this extended game as a Perturbed game. Perturbed games and Robust Correlated Equilibrium are two of the primary contributions of this dissertation and are summarized in Paper E. Formally a Perturbed game is defined by a tuple $\Gamma' = (N, U, C_D)$, where

- N is the finite number of players,
- $U := \prod_{i \in N} U^i := U^1 \times U^2 \times \cdots U^N$,
- D be the finite number of perturbed costs due to exogenous disturbance,
- $C_D := \{c^1, \cdots, c^N\}$ is the set of costs incurred by the players that are indexed by the superscripts, where each $c^i : U \mapsto \mathbb{R}^D$. The components of c^i are where $d \in \{1, \cdots, D\}$.

In the perturbed game Γ', the correlated equilibrium might not be robust to disturbances and this is demonstrated via the following example taken from Paper E [58].

2.3 Example (Game of Irrigation between farms [58]) Consider two farms that use water from a common irrigation channel. Each farm has an automatic controller that directs water from the irrigation channel to the farm. Assume for simplicity, that both the farms continuously require water. The controller can either *open* the valve (action O) or keep it *closed* (action C). Under all conditions, the water in the irrigation channel can supply only one farm at a time. If both farms try to use water from the irrigation channel simultaneously it will be drained for a long time. The cost incurred by the controllers for opening the valve simultaneously is 8 units. The cost incurred by the controller for opening the valve while the other controller's valve is closed is 0 units and the cost incurred for keeping the valve closed while the other farm consumes the water is 5 units. Lastly, if both the controllers do not consume water they pay 1 unit each. The water available in the common irrigation supply depends on the weather conditions (specifically whether there is a drought). Under drought conditions, it becomes more expensive for the farm to not consume water while the other farm consumes it and this is reflected in the costs incurred by the controllers as the cost incurred by a controller for keeping the valve closed, while the other controller opens its valve is increased to 7.5 units form 5 units. The following cost matrices summarize the game.

$d =$ Normal condition

		Control 2	
		C	O
Control 1	C	$(1,1)$	$(5,0)$
	O	$(0,5)$	$(8,8)$

$d =$ Drought condition

		Control 2	
		C	O
Control 1	C	$(1,1)$	$(7.5,0)$
	O	$(0,7.5)$	$(8,8)$

Let $\psi_{CC}, \psi_{CO}, \psi_{OC}$, and ψ_{OO} represent the probabilities of the recommender suggesting the actions $\{C,C\}$, $\{C,O\}$, $\{O,C\}$, and $\{O,O\}$ to controller 1 and 2 respectively. Consider the correlated equilibrium where the public recommendation to both players is as follows,

$$\psi_{CC} = \frac{1}{3}, \quad ,\psi_{CO} = \frac{1}{3}, \quad \psi_{OC} = \frac{1}{3}, \quad \psi_{OO} = 0. \tag{2.9}$$

Consider the normal conditions i.e. (d = Normal condition), and suppose that the control 1 is recommended action C, than control 1 can infer that control 2 must have received the recommendation C or O with equal probability and therefore the expected cost for control 1 is,

$$\frac{1}{2}(1) + \frac{1}{2}(5) = 3, \text{ if control 1 follows } C,$$
$$\frac{1}{2}(0) + \frac{1}{2}(8) = 4, \text{ if control 1 switches to action } O,$$

and under recommendation O, the control 1 infers that control 2 must have been recommended to choose C with probability 1, and therefore cost incurred by control 1 is 0 for following the recommendation and 1 unit for not following the recommendation. Thus, the distribution (2.9) is a correlated equilibrium. Now consider the drought conditions i.e. (d = Drought), and suppose that the control 1 is recommended action C, than control 1 can infer that control 2 must have received the recommendation C or O with equal probability and therefore the expected cost for control 1 is,

$$\frac{1}{2}(1) + \frac{1}{2}(7.5) = 4.25, \text{ if control 1 follows } C,$$
$$\frac{1}{2}(0) + \frac{1}{2}(8) = 4, \text{ if control 1 switches to action } O,$$

and the costs remain the same if the recommendation to control 1 was O. Clearly, it is better for control 1 to not follow the recommendation in the case recommendation is C during drought conditions and this demonstrates that the correlated equilibrium distribution (2.9) is not robust to the disturbance d.

The condition on a finite number of disturbances can be relaxed to a continuum of disturbances provided that the disturbances are bounded. Robust Correlated Equilibrium seeks to robustify the definition of Correlated Equilibrium to these exogenous disturbances and is defined as follows.

Definition 2.8 (Robust Correlated Equilibrium [58]) Given a correlating device $\mathcal{CD}$, a joint distribution $\pi^\star(u \mid P^i(u))$ is called a Robust Correlated Equilibrium for the Perturbed Game Γ' if for every player i, it is an element of the following set

$$\Big\{\pi^\star(u \mid P^i(u)) \in \Delta(U) : c_d^i(\pi^\star(u^i, u^{-i} \mid P^i(u))) \leq c_d^i(\pi^\star(\tilde{u}^i, u^{-i} \mid P^i(u))),$$
$$\forall u^i, \tilde{u}^i \in U^i, \text{ where } u^i \sim \pi^\star(u \mid P^i(u)), \text{ and } \forall d \in \{1, \cdots, D\}\Big\}. \quad (2.10)$$

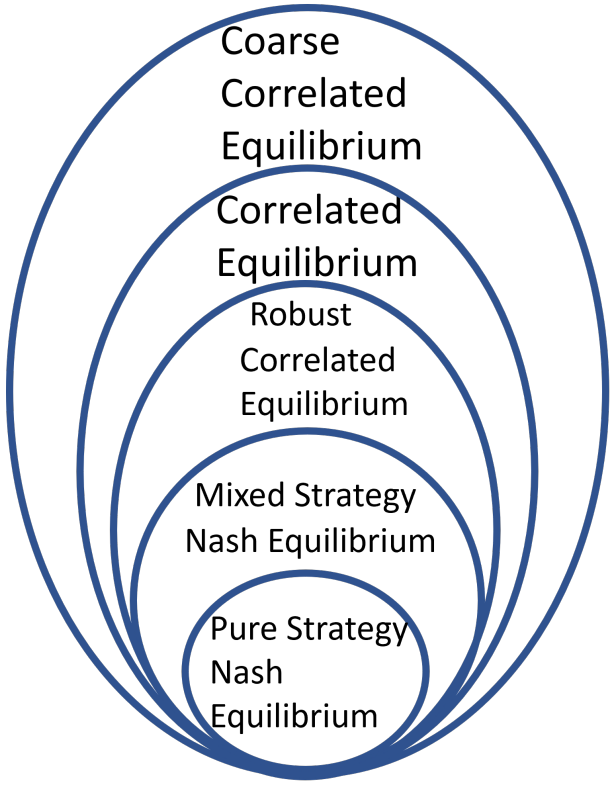


Fig. 2.1: Summary of solution concepts.

The existence of Robust Correlated Equilibrium is proven using the concept of Blackwell's Approachability theorem which shall be discussed in the following section on Repeated Games. While Robust Correlated Equilibrium can serve as a good approximation for the games with states wherein the disturbances d can be thought of as states, however, players do not control which states are visited. Thus, we shall also consider dynamic games that involve states in the later section on Markov Games and Differential Games. The solution concepts discussed so far are summarized in the following figure 2.1.

2.2 Repeated Games and Blackwell's Approachability theorem

Repeated games are essentially repetitions of the Static game Γ or the Perturbed game Γ' wherein the concept of discrete time is introduced. This gives rise to the Theory of Learning in games [29] (that has been researched extensively since the 1950s [10, 13, 31]

and is still an active field of research [16, 19]) since players can now try out different actions and the fundamental question arises whether the joint action profile converges to any of the equilibrium solutions. Consider the game Γ (or equivalently Γ') with joint action profile $u = (u^i, u^{-i})$ and let the partial history of play (i.e. history from player i's perspective) up to time t be denoted by h_t^i and defined as follows,

$$h_t^i := (u_0^i, c_1^i, \cdots, u_{t-1}^i, c_t^i). \tag{2.11}$$

The full history of play up to time t is the product of partial histories up to time t,

$$h_t := \prod_{i \in \{1,\cdots,N\}} h_t^i. \tag{2.12}$$

The empirical distribution of joint actions played is defined as

$$\bar{\pi}_t(u) := \frac{1}{t} \mid \{\tau \leq t : u_\tau = u\} \mid . \tag{2.13}$$

We can now define the problem of learning in games as follows.

Definition 2.9 (Learning in Repeated games) Consider the game Γ (or equivalently Γ') and for each player $i \in \{1, \cdots, N\}$, the learning problem is defined as follows: find π^i such that given the partial history (2.11), the empirical distribution of joint action profile (2.13) converges to the desired equilibrium solution (example: Pure Strategy Nash Equilibrium (2.2), Mixed Strategy Nash equilibrium (2.3), Coarse Correlated Equilibrium (2.8), Correlated Equilibrium (2.7), or Robust Correlated Equilibrium (2.10)).

Many algorithms have been designed to solve the problem of learning in repeated games such as fictitious play, smooth calibration, and no-regret learning [29]. Furthermore, any real-life problems involving online decision-making can be cast as a repeated game and solved using these algorithms (for example weather forecasting [27], decentralized power control [49], auctions [46], decentralized control of water networks [55]). In this dissertation, we specifically focus on the concepts of Correlated Equilibrium and related concepts such as Robust Correlated Equilibrium since learning Nash equilibrium points has been proven to be computationally intractable [20, 65]. We now introduce the concept of Approachability for learning strategies for multi-objective games.

Approachability for games with vector costs

The concept of approachability introduces a mathematical framework for approaching a target set that satisfies multiple objectives or multiple criteria simultaneously despite

the presence of an adversarial opponent. The approachability theorem was originally introduced by Blackwell in [10] with his motivation being to extend the celebrated von Neumann's Minimax theorem to games with vector costs. In his seminal work [10], Blackwell characterized both approachable and excludable sets. He provided both sufficient and necessary conditions for convex target sets and sufficient conditions for arbitrary non-convex target sets. An overview of Approachability and related literature can be found in [66]. Furthermore, Approachability inspired some of the key concepts in Machine Learning such as Online Optimization [18, 37, 66, 93] and the related concepts of no-regret learning [1, 38, 66] and calibration [27, 28, 35, 66]. We shall now formally define approachability and present Blackwell's Approachability theorem. Consider a 2-player zero-sum game defined via the tuple $\Gamma_z = (2, U^1 \times U^2, \mathbf{c}, -\mathbf{c})$, with the m-dimensional cost $\mathbf{c} : U^1 \times U^2 \mapsto \mathbb{R}^m$. Consider a closed convex set $\mathcal{A} \subset \mathbb{R}^m$ and a point $a \in \mathbb{R}^m$. The distance between the point a and the set $\mathcal{A}$ is defined as

$$dist_{\mathcal{A}}(a) = \inf_{b \in \mathcal{A}} \|a - b\| .$$

For a $\delta > 0$, let $\mathcal{A}^\delta$ denote the δ open neighborhood of $\mathcal{A}$ i.e.

$$\mathcal{A}^\delta := \{b \in \mathbb{R}^m \mid dist_{\mathcal{A}^\delta} < \delta\}.$$

For a given time instant t, define $\mathbf{c}_t := \mathbf{c}(u_t^1, u_t^2)$ and consider a discrete time sequence of costs $\mathbf{c}_1, \mathbf{c}_2, \cdots$. Let $\bar{a}_t$ denote the time average of the aforementioned sequence i.e.

$$\bar{\mathbf{a}}_t := \frac{\sum_{s=1}^{t} \mathbf{c}_s}{t}.$$

A control strategy for player i, where $i \in \{1, 2\}$ is a mapping from the set of histories to distribution over actions i.e. $\pi_t^i : h_t^i \mapsto \Delta(U^i)$. As per Kolmogorov's extension theorem and standard product topology [87], the joint distribution (π^1, π^2) induces a probability distribution $\mathbb{P}_{\pi^1,\pi^2}$ over h_t (set of all possible infinite histories of the game Γ_z) [66].

Definition 2.10 (Approachability and Exludability [10, 66]) A convex and compact set $\mathcal{A} \in \mathbb{R}^m$ is *Approachable* by player 1 if there exists a strategy π^1 such that for every $\varepsilon > 0$, there exists a natural number $T(\varepsilon)$ such that for every possible strategy π^2 of player 2, the following condition holds

$$\sup_{t > T(\varepsilon)} \mathbb{E}_{\pi^1,\pi^2}[dist_{\mathcal{A}}(\bar{\mathbf{a}}_t)] \leq \varepsilon \text{ and } \mathbb{P}_{\pi^1,\pi^2}\left(\sup_{t > T(\varepsilon)} \mathbb{E}_{\pi^1,\pi^2}[dist_{\mathcal{A}}(\bar{\mathbf{a}}_t)] \geq \varepsilon \right) \leq \varepsilon. \quad (2.14)$$

The set $\mathcal{A}$ is *Excludable* by player 2 if player 2 can approach the complement of set $\mathcal{A}^\delta$ for some $\delta > 0$.

If the costs associated with the game Γ_z were scalar i.e. $c : U^1 \times U^2 \mapsto \mathbb{R}$, then Approachability is the same as optimizing (in an asymptotic sense) against worst-case strategy of the opponent as the set $\mathcal{A} \subset \mathbb{R}$ and therefore Approachability implies that the time-average of worst-case costs are asymptotically bounded. The following example demonstrates the equivalence (or lack of it) between Minimax theorem and Approachability.

2.4 Example (Minimax theorem and Approachability) Consider a 2-player zero sum game, where player 1 is the minimizer and player 2 is the maximizer with control action spaces $U^1 = U^2 = \{0, 1\}$ and the cost $c = u^1 u^2$, where $u^1 \in U^1$ and $u^2 \in U^2$. Can player 1 ensure that the worst-case cost is 0 i.e. can player 1 *Approach* the set $\mathcal{A} = (-\infty, 0]$ irrespective of what player 2 does? By Von Neumann's Minimax theorem [90], we have

$$\max_{u^2 \in U^2} \min_{u^1 \in U^1} u^1 u^2 = 0 = \min_{u^1 \in U^1} \max_{u^2 \in U^2} u^1 u^2$$

Thus if player 1 chooses $u^1 = 0$, than irrespective of what player 2 does, player 1 can ensure that the cost is no more than 0. This is why we can interchange the $\min(\cdot)$ and $\max(\cdot)$ operations as it does not matter which player optimizes first. Now, we consider the vector case where the 2-player zero sum game is same as before except the cost is a vector in $\mathbb{R}^2$ and is defined as follows, $\mathbf{c} = [u^1, u^2]$. Observe now that the $\min(\cdot)$ and $\max(\cdot)$ cannot be directly used as we have not defined any notion of minimality or maximality of a vector. However the notion of Approachability (to a predefined set) can still be used and we pose the following question. Let player 1's goal be to *Aprroach* $\mathcal{A}$ and player 2's goal be to *Exclude* player 1 from $\mathcal{A}$. Can player 1 *Approach* the set $\mathcal{A}$ defined as follows: $\{u \in U^1, u \in U^2 : [u, u] \in \mathbb{R}^2\}$? Observe that unlike the scalar case, it matters which player chooses their control action first as if player 1 chooses $u = 0$, player 2 can choose $u = 0$ and exclude player 1 from set $\mathcal{A}$. Blackwell showed that while player 1 might not be able to Approach the set in a static or 1-shot game, player 1 can ensure Approachability of time-average of costs for a repeated game asymptotically. A similar example for compact control spaces is presented in [1].

Given $\bar{\mathbf{a}}_t$, let $\Pi(\bar{\mathbf{a}}_t)$ be the projection of $\bar{\mathbf{a}}_t$ on $\mathcal{A}$ i.e.

$$\Pi(\bar{\mathbf{a}}_t) := \left\{ b' \in \mathcal{A} \mid b' \in \arg\min_{b \in \mathcal{A}} \|\bar{\mathbf{a}}_t - b\| \right\}.$$

Let $\lambda(\bar{\mathbf{a}}_t)$ be the vector pointing from $\bar{\mathbf{a}}_t$ to $\Pi(\bar{\mathbf{a}}_t)$. Observe that due to the convexity of $\mathcal{A}$, it will be normal to the half-space $\mathcal{H}(\bar{\mathbf{a}}_t)$ that also includes $\Pi(\bar{\mathbf{a}}_t)$ (see optimality

of projection in [12]). We can now state the celebrated Blackwell's Approachability Theorem [10] that states the necessary and sufficient conditions under which player 1 can ensure approachability of $\bar{\mathbf{a}}_t$ to $\mathcal{A}$.

Theorem 2.2 (Blackwell's Approachability Theorem [10, 66]). *The closed convex set $\mathcal{A} \subset \mathbb{R}^m$ is Approachable by player 1 as per (2.14) if and only if for every $\mathbf{c} \notin \mathcal{A}$, there exists a strategy $\pi^1_{\mathcal{H}(\bar{\mathbf{a}}_t)} \in \Delta(U^1)$ such that,*

$$(\bar{\mathbf{a}}_t - \Pi(\bar{\mathbf{a}}_t)) \cdot (\mathbf{c}_{t+1}(\pi^1_{\mathcal{H}(\bar{\mathbf{a}}_t)}, \pi^2) - \Pi(\bar{\mathbf{a}}_t)) \leq 0, \quad \forall \pi^2 \in \Delta(U^2). \tag{2.15}$$

Furthermore, the following procedure for player 1 guarantees that $\bar{\mathbf{a}}_t$ approaches $\mathcal{A}$ as per (2.14),

$$\pi^1 = \begin{cases} \pi^1_{\mathcal{H}(\bar{\mathbf{a}}_t)} & \text{if } \bar{\mathbf{a}}_t \notin \mathcal{A} \\ \text{arbitrary} & \text{if } \bar{\mathbf{a}}_t \in \mathcal{A} \end{cases} \tag{2.16}$$

If condition (2.15) is not satisfied by player 1, than there exists a strategy such that the set $\mathcal{A}$ is Excludable by player 2.

In words Blackwell's Approachability theorem states that player 1 can approach the set $\mathcal{A}$, if there exists a strategy such that player 1 can approach the half-space $\mathcal{H}(\bar{\mathbf{a}}_t)$ irrespective of what player 2 does i.e. half-space $\mathcal{H}(\bar{\mathbf{a}}_t)$ is Enforceable by player 1. Thus, the problem of approaching any arbitrary set is reduced to enforcing that the next instance of cost vector $\mathbf{c}_{t+1}$ is on the other side of $\mathcal{H}(\bar{\mathbf{a}}_t)$ irrespective of what the other player does. A natural way to achieve this is to choose π^i_{t+1} such that $\lambda(\bar{\mathbf{a}}_t)$ is minimized. This is the basis for learning algorithms such as regret matching [32] (where $\lambda(\bar{\mathbf{a}}_t)$ is equivalent to regret given that $\mathcal{A} = \{x \leq 0 \mid x \in \mathbb{R}^m\}$), online gradient descent [93] and other online convex optimization techniques [1, 37, 38]. The following figure 2.2 visually showcases Blackwell's Approachability Theorem in the m-dimensional space of $\mathbf{c}$. Note that Blackwell's definitions of Approachability (2.14) and the Approachability theorem consider specifically the time-average dynamics of the cost $\bar{a}_t$. Can we consider dynamics beyond the time-average, such as a weighted average? What about dynamics that incorporate not only the time average of past costs but also the past two iterations, to enhance robustness against perturbations in perturbed games? The former is addressed in [22], showing faster convergence in repeated games. The latter is explored in this dissertation, specifically in Paper E, where definition 5.2 and Theorem 5.2 demonstrate robustness to bounded disturbances and convergence to Robust Correlated Equilibrium in perturbed games. Another application of Blackwell's Approachability theorem is for designing Counterfactual regret minimization algorithms for Extensive form Games [14]. Extensive form games are a generalization of Markov games to partial information setting and are analogous to Partially Observable Markov Decision Process [63] and are outside the scope of this dissertation. Similar to how Sion's Minimax theorem

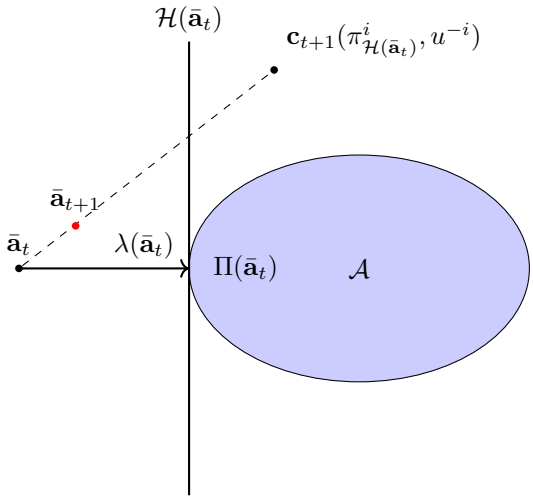


Fig. 2.2: Approachability towards $\mathcal{A}$ by player 1. $\bar{\mathbf{a}}_t$ is the time-average of cost vector at time t. $\Pi(\bar{\mathbf{a}}_t)$ is the projection of $\bar{\mathbf{a}}_t$ on $\mathcal{A}$. $\mathcal{H}_t$ is the half-space intersecting $\Pi(\bar{\mathbf{a}}_t)$ and $\lambda(\bar{\mathbf{a}}_t)$ is normal to $\mathcal{H}_t$. Since player 1 has a strategy that enforces the next immediate cost $\mathbf{c}_{t+1}$, the time average $\bar{\mathbf{a}}$ Approaches $\mathcal{A}$.

extends Von Neumann's Minimax theorem, Blackwell's Approachability theorem can be extended to the case where the control action spaces are convex, compact spaces. As noted in [1, 66] for a given sequence $\mathbf{c}_t$, Approachability 2.10 does not require observation of opponents control actions and nor does it require finiteness of control action spaces (provided convexity and compactness of control action spaces U^1, U^2 is guaranteed and the considered cost function $\mathbf{c} : U^1 \times U^2 \mapsto \mathbb{R}^m$ satisfies the continuity and convex-concave in first and second arguments conditions as stated in Theorem 2.1).

Theorem 2.3 (Blackwell's Theorem for compact control action spaces [66]). *The closed convex set $\mathcal{A} \subset \mathbb{R}^m$ is Approachable by player 1 as per (2.14) if and only if for every $\mathbf{c} \notin \mathcal{A}$, there exists a strategy $\pi^1_{\mathcal{H}(\bar{\mathbf{a}}_t)} \in \mathbb{R}^{p_1}$ such that,*

$$\inf_{u^1 \in \mathbb{R}^{p_1}} \sup_{u^2 \in \mathbb{R}^{p_2}} (\bar{\mathbf{a}}_t - \Pi(\bar{\mathbf{a}}_t)) \cdot (\mathbf{c}_{t+1}(\pi^1_{\mathcal{H}(\bar{\mathbf{a}}_t)}, u^2) - \Pi(\bar{\mathbf{a}}_t)) \leq 0. \tag{2.17}$$

Furthermore, the following procedure for player 1 guarantees that $\bar{\mathbf{a}}_t$ approaches $\mathcal{A}$ as per (2.14),

$$\pi^1 = \begin{cases} \pi^1_{\mathcal{H}(\bar{\mathbf{a}}_t)} & \text{if } \bar{\mathbf{a}}_t \notin \mathcal{A} \\ \text{arbitrary} & \text{if } \bar{\mathbf{a}}_t \in \mathcal{A} \end{cases} \tag{2.18}$$

If condition (2.17) is not satisfied by player 1, than there exists a strategy such that the set $\mathcal{A}$ is Excludable by player 2.

The primary difference between Theorems 2.2 and 2.3 is that the control spaces are no longer probability spaces and that we modify the Blackwell condition with inf sup over control action spaces instead of requiring the condition to hold for any finite control action $u^2 \in U^2$.

2.3 Markov Games

Markov Games were introduced in the seminal Paper by Shapley [78] and are a generalization of both the Markov Decision process and Static Games. They generalize Markov Decision processes as they include more than one decision maker and they generalize static games as now player's decisions affect not only the immediate costs but also the transition to future states and therefore the future costs. They are formally defined as a tuple $\mathcal{G} = (N, X, U, \mathbb{P}, C, \beta, \tau)$, where

- N is the finite number of players,
- $X := \{x_1, \cdots, x_n\}$ is the finite number of states,
- X can be partitioned into a set of absorbing target states T, and set of transient states E
- $U := \prod_{i \in N} U^i := U^1 \times U^2 \times \cdots U^N$ is the set of m finite state-dependent actions available to players, where $U^i : X \mapsto \{u_1^i, \cdots u_m^i\}$,
- $\mathbb{P} : X \times U \times X \mapsto [0, 1]$ is the transition matrix that maps probability of reaching state x_{next} given $x_{previous}$ and control action u,
- $C := \{c^1, c^2, \cdots, c^N\}$ is the set of costs incurred by the players that are indexed by the superscripts, where $c^i : X \times U \mapsto \mathbb{R}$,
- $\beta \in \Delta(\mathrm{E})$ is the initial distribution over the transient states,
- τ is the random stopping time indicating exit from transient states E.

Define $t = 1, 2, \cdots$ as the discrete time, and consider a discrete-time stochastic process $\{x_t\}_{t=0}^{\tau}$ with randomized stopping time τ defined on the set of finite states X. Each player chooses actions $u_t^i \in U^i$. The joint control action at time t is $u_t = (u_t^i, u_t^{-i})$. Let the space of sample paths generated by $\{x_t\}$, $\{u_t\}$ denote the canonical sample space Ω of the considered stochastic process. The random stopping time τ is defined as follows

$$\tau := \left\{ \inf_t x_t \in \mathrm{T} \mid x_0 \in \mathrm{E} \right\}$$

Given a player i, we consider two classes of control policies π^i (Markovian policies) and μ^i (History-dependent policies) which are defined as follows. Let

$$h_t^i = (x_0, u_0^i, c_1^i, x_1, u_1^i, c_2^i \cdots, u_{t-1}^i, x_t, c_t^i),$$

represent the partial history of stochastic process X_t up to time t.

Definition 2.11 (History-dependent policies μ_t^i [23]) Let H_t^i represent the set of all possible partial histories up to time t. Than $\mu_t^i : H_t^i \to \Delta(U^i)$ is defined as

$$\mu_t^i(h_t^i) := \left[\mu^i(u_1^i \mid h_t^i), \cdots, \mu^i(u_{|U^i|}^i \mid h_t^i)\right] \in \Delta(U^i). \tag{2.19}$$

History-dependent policies are a richer class of policies compared to Markov policies, however often they are more expensive to compute since they require memory to keep track of the time-dependent history. Therefore, we will attempt to restrict our attention to memoryless or Markov policies whenever possible which are defined as follows.

Definition 2.12 (Markov policies π^i [23]) A Markov policy $\pi^i : X \to \Delta(U^i)$ is a stationary (i.e. independent of time) control policy that is dependent only on the given state and is defined as

$$\pi^i(x) = \left[\pi^i(u_1^i \mid x), \cdots, \pi^i(u_{|U^i|}^i \mid x)\right] \in \Delta(U^i). \tag{2.20}$$

Although Markov policies are more appealing in terms of computation, in many cases we might be forced to consider history-dependent policies as an "optimal" policy might not exist in the space of Markov policies [36, 45] or would be intractable (as is in the case of finding coarse correlated equilibrium policies for Markov games in the space of stationary Markov policies [21]). In the sequel, we will consider stationary Markov policies unless explicitly specified otherwise. The goal of each player is to minimize their expected costs (in a decentralized setup) given the initial distribution of states and formally this is defined as follows,

Definition 2.13 (Decentralized Markov games) Let $\pi^i : X \mapsto \Delta(U^i)$ represent the control policy for player i and $\pi : X \mapsto \Delta(U)$ represent the joint control policy for all players. Every joint control policy π together with an initial state $x \sim \beta$ induces a distribution $\mathbb{P}_{\pi,x}$ on $\{\Omega\}$. For systems with finite states and control actions, the corresponding transition probabilities $p(j \mid i, u)$ can be obtained from the distribution $\mathbb{P}_{\pi,x}$ as follows,

$$p(j \mid i, u) = \mathbb{P}_{\pi,x}[x_{t+1} = j \mid x_t = i, u_t = u],$$

and the corresponding expectation operator is denoted by $\mathbb{E}_{\pi,x}$. We extend the definition of $c^i(\pi^i, \pi^{-i})$ from Static games to the Markov games setting by defining

$c^i(x, \pi^i(x), \pi^{-i}(x))$ as follows,

$$c^i(x, \pi^i(x), \pi^{-i}(x)) := \mathbb{E}_{\pi,x}[c^i(x, u^i, u^{-i})] = \mathbb{E}_{\pi^i,\pi^{-i},x}[c^i(x, u^i, u^{-i})] \tag{2.21}$$

For a given player i and a given joint policy π, the value of starting at state x is denoted by $V^i_\pi(x)$ and is defined as follows,

$$V^i_\pi(x) : = \mathbb{E}_{\pi,x}\left[\sum_{t=0}^{\tau} c^i_t(x_t, u^i_t, u^{-i}_t)\right],$$

The goal of each player i is to find an optimal policy π^i such that

$$\min_{\pi^i} V^i_\pi(x) = \min_{\pi^i} \mathbb{E}_{\pi,x}\left[\sum_{t=0}^{\tau} c^i_t(x_t, u^i_t, u^{-i}_t)\right], \quad \forall x \in X. \tag{2.22}$$

The value vector is defined as $\mathbf{V}^i_\pi = [V^i_\pi(x_1), \cdots, V^i_\pi(x_n)]$ with entries corresponding to each initial state.

Generally the optimization problem (2.22) is formulated as an infinite horizon optimization problem as for the finite horizon problems, the optimal policy will be time-dependent [23]. However, for an infinite horizon optimization problem, the summand

$$\sum_{t=0}^{\infty} c^i_t(x_t, u^i_t, u^{-i}_t),$$

will not be summable. Thus, discount factor $\gamma \in [0, 1)$ is commonly introduced for such problems as the summand becomes a convergent geometric series (with γ being the common ratio between the terms [23]). More precisely,

$$\sum_{t=0}^{\infty} \gamma^t c^i_t(x_t, u^i_t, u^{-i}_t) = (\mathbf{I}_{n\times n} - \gamma \mathbb{P}_\pi)^{-1} c^i_t(x_t, u^i_t, u^{-i}_t) < \infty. \tag{2.23}$$

Note that due to the presence of stopping time τ, the value vector will be summable (i.e. the entries of $\mathbf{V}^i_\pi < \infty$) as the matrix $\mathbb{P}_\pi$ will be sub-stochastic with the sum of all its entries less than 1. The stopping time τ can be interpreted as an artificial discount factor [23] as follows. Since

$$\sum_{x'=x_1}^{x_n} p(x' \mid x, u) < 1, \quad \forall x \in \mathrm{E}, u \in U(x),$$

we can define the stopping probability for states $\tilde{x} \in \mathrm{T}$ as

$$p(\tilde{x} \mid x, u) := 1 - \sum_{x'=x_1}^{x_n} p(x' \mid x, u) > 0.$$

Define the discount factor $\gamma' := 1 - p(\tilde{x} \mid x, u)$ as the probability of continuation (i.e. not stopping). This leads to a summable value vector for each player i and for any joint policy π analogous to the regular discount factor-based model (2.23) where we replace γ by γ'.

Remark 2.2. *The most important feature of Markov Games is that for each player i, the immediate cost $c^i(x, u^i, u^{-i})$ and the transition to the next state $p(x' \mid x, u^i, u^{-i})$ is affected by the joint control action taken by all the players.*

Similar to Definition 2.2, we can define the Mixed Strategy Nash Equilibrium for Markov games as follows,

Definition 2.14 (Mixed Strategy Nash Equilibrium [23]) A product probability distribution $\pi^\star := \pi^{1\star} \times \cdots \times \pi^{N\star}$ is the Mixed Strategy Nash equilibrium of the game $\mathcal{G}$ if for any player i and any initial state $x \in X$, the marginal distribution $\pi^{i\star}$ belongs to the following set

$$\left\{\pi^{i\star} \in \Delta(U^i(x)) : V^i_{\pi^\star}(x) \leq V^i_{\tilde{\pi}}(x), \quad \forall x \in X, i \in \{1, \cdots, N\}\right\}, \tag{2.24}$$

where $\tilde{\pi} = \pi^{1\star} \times \cdots \times \tilde{\pi}^i \times \cdots \times \pi^{N\star}$ is any possible unilateral deviation by player i from the product probability distribution $\pi^\star$.

In the sequel, the operator val[·] shall indicate the value of a game in the sense of a specified equilibrium condition. It is a generalization of the standard optimization operators min (or the max) in a game theoretic sense. For example, the val[·] operator for a 2-player zero-sum game is precisely the value V of the objective function obtained after applying the min max operator with π^1, π^2 as decision variables and for a general mixed strategy Nash equilibrium as per (2.14), the val[·] operator for player i is the optimal $V^i_{\pi^\star}$ in the sense of Nash equilibrium profile $\pi^\star$. Similarly, we can define the val[·] operator for the Correlated Equilibrium condition. We can now conjecture an "Optimality Equation" for Markov games that is similar to the idea of the "Principle of Optimality" for the Markov Decision Process.

Definition 2.15 (Optimality Equation for Markov games [23]) The optimal value vector for each player i, denoted by $\mathbf{V}^i_{\pi^\star}$ satisfies the following optimality equation component-wise,

$$V^i_{\pi^\star}(x) = \text{val}\left[c^i(x, u^i, u^{-i}) + \sum_{x' \in X} p(x' \mid x, u^i, u^{-i}) V^i_{\pi^\star}(x')\right], \quad \forall x \in X, \tag{2.25}$$

where c^i is the instantaneous cost incurred by player i for being at state x and taking action u^i given u^{-i}.

The existence of stationary strategies $\pi^\star$ satisfying Definition 2.15 for zero-sum games was proven by Shapely in his seminal Paper on Markov games [78]. This solution is also referred to as Markov perfect Equilibrium in literature [7]. As discussed in the previous section on static games, when it comes to nonzero-sum games, there are multiple Nash equilibrium points and it is unclear as to which equilibrium point will be chosen by the players. Furthermore, the concerns about the inferiority of Nash equilibrium points to Pareto Optimal solutions and the incompatibility of Nash equilibrium solutions with Bayesian rationality naturally extends to Markov Games as well. Therefore, the concept of correlated equilibrium was also extended to Markov games in [62, 85]. The existence of correlated equilibrium was proved for Markov games having state and action spaces represented as continuous Borel spaces in [62] (see [44] for a summary of these developments). We consider the restricted setting of finite state and finite action spaces and the definitions presented here are based on [85]. We begin by defining the extended correlating device as follows.

Definition 2.16 (Extended Correlating Device [85]) Given time $t \in \mathbb{N}$, consider a finite set of correlating signals $U = \prod_{i=1}^N U^i$ and the partitions P^i_t of U for every player $i \in \{1, \cdots, N\}$, and a probability simplex over the finite set of correlating signals denoted by $\mu^\star_t : \prod_{i=1}^N h^i_t \times P^i_t(u) \to \Delta(U)$. Then a extended correlating device is defined by the tuple $\mathcal{ECD} = (U, \prod_{i=1}^N P^i_t, \mu^\star_t)$. The extended correlating device is said to be stationary if it is independent of t.

Consider the Markov game $\mathcal{G}$ along with the extended correlation device $\mathcal{ECD}$. We refer to such game as $\mathcal{G}(P)$ where at the beginning of each time instant t of the game, each player i observes the correlating signal $P^i_t(U)$ from the Extended Correlating device $\mathcal{ECD}$. Therefore the history of the game from player i's perspective is appended with the signals P^i_t as follows,

$$H^i_t := (x_0, P^i_0, u^i_0, c^i_1, \cdots, x_{t-1}, P^i_{t-1}, u^i_{t-1}, c^i_t, x_t). \tag{2.26}$$

The full history of play up to time t is the product of partial histories up to time t,

$$H_t := \prod_{i \in \{1, \cdots, N\}} H_t^i. \tag{2.27}$$

We consider History-dependent policies that take into account H_t^i as defined previously in Definition 2.11. Every joint policy $\mu : H_t \mapsto \Delta(U)$ and starting state $x \sim \beta$ induces a distribution over $\{\Omega\}$ and we denote that by $\mathbb{P}_{\mu,x}$ with the corresponding expectation operator $\mathbb{E}_{\mu,x}$. Since we consider systems with finite states and finite actions, the corresponding transition probabilities are obtained as

$$p(j \mid i, u) = \mathbb{P}_{\mu,x}[x_{t+1} = j \mid x_t = i, u_t = u],$$

and the corresponding value function is

$$V_\mu^i(x) := \mathbb{E}_{\mu,x}\left[\sum_{t=0}^{\tau} c_t^i(x_t, u_t^i, u_t^{-i})\right],$$

The correlated equilibrium can now be defined as follows,

Definition 2.17 (Correlated Equilibrium [85]) Given an extended correlating device $\mathcal{ECD}$, a conditional distribution $\mu^\star$ is the correlated equilibrium of the game $\mathcal{G}(P)$ if for every player $i \in \{1, \cdots, N\}$, $\mu^\star$ belongs to the following set

$$\left\{\mu^\star : H_t \mapsto \Delta(U(x)) : V_{\mu^\star}^i(x) \leq V_{\tilde{\mu}}^i(x),\ \forall x \in X\right\}, \tag{2.28}$$

where $\tilde{\mu} := \mu^{1\star} \times \cdots \times \tilde{\mu}^i \times \cdots \times \mu^{N\star}$ is any possible unilateral deviation by player i from the product probability distribution $\mu^\star$.

The Paper [85] showcases that every Markov game with a correlation device $\mathcal{ECD}$ admits a correlated equilibrium. The optimality equation for a correlated equilibrium can be obtained by simply replacing the val operator in (2.25) with the conditional distribution $\mu^\star$ as follows,

$$V_{\mu^\star}^i(x) = \sum_{u \in U} \mu^\star(x, u \mid P_t^i(u)) \left[c^i(x, u) + \sum_{x' \in X} p(x' \mid x, u) V_{\mu^\star}^i(x')\right], \quad \forall x \in X. \tag{2.29}$$

Thus, we have established the equations that need to be solved for solving Static games and Markov games. We shall now discuss the computation procedures for solving the aforementioned equations in the following subsections.

2.4 Computation procedures for game-theoretic solution

In this section, we shall summarize some techniques for computing solutions for Static and Markov games. We shall now endeavor to compute an equilibrium solution for all the players in a centralized setting. Furthermore, players know the functional form of the cost function. Generally, decentralized computation involving unknown costs of game-theoretic solutions usually involves some iterative procedure akin to learning, which will be elaborated on in the next subsection. This subsection is based on the books [79], [23], and the excellent survey paper [71].

Solution for static games

Solutions for zero-sum games were developed jointly with mathematical programming and optimization theory and are closely associated with duality theory (and Karush-Kuhn-Tucker (KKT) conditions) in optimization literature [12]. This is because the solution of a zero-sum game is a saddle point (in the space of player 1 and player 2's mixed strategies) and the solution of a constrained optimization problem such as a Linear Program is also represented via a saddle point (in the space of the decision variable and the Lagrange multipliers). The following Linear program [79] computes the solution of a 2-player zero-sum game or the Minimax solution for N-player game,

$$\min_{\pi^i, V^i} \quad V^i \tag{2.30a}$$

$$\text{s.t.} \quad \sum_{u^i \in U^i} c^i(u^i, u^{-i})\pi^i(u^i) \leq V^i, \forall u^{-i} \in U^{-i}, \tag{2.30b}$$

$$\sum_{u^i} \pi^i(u^i) = 1, \tag{2.30c}$$

$$\pi^i(u^i) \geq 0, \ \forall u^i \in U^i, \tag{2.30d}$$

where V^i is the equilibrium value of the game for player i. Note that player i does not need to know the costs of other players or their strategies. The only information required is the possible actions that can be taken by other players. Note that the objective function in a zero-sum game remains the same for both the players up to a sign as one player is minimizing the objective function, whereas the other player is maximizing the objective function (hence the sum of both the player's objective functions is zero which gives the name zero-sum game). This is what makes it relatively much easier to solve zero-sum games compared to nonzero-sum games even in a 2-player case as in nonzero-sum games, the objective functions of players are fundamentally different, and the solution(s) belong to a set that is characterized as follows. Consider a 2-player nonzero-sum game, then as per Definition 2.2, a Mixed strategy Nash equilibrium

solution $(\pi^{1\star}, \pi^{2\star})$ belongs to the following set,

$$\left\{\pi^{1\star} \in \Delta(U^1) : c^1(\pi^{1\star}, \pi^{2\star}) \leq c^1(\tilde{\pi}^1, \pi^{2\star})\right\}, \tag{2.31}$$

$$\left\{\pi^{2\star} \in \Delta(U^2) : c^2(\pi^{1\star}, \pi^{2\star}) \leq c^2(\pi^{1\star}, \tilde{\pi}^2)\right\}, \tag{2.32}$$

where $\tilde{\pi}^1, \tilde{\pi}^2$ are unilateral deviations from $\pi^\star$ by player 1 and player 2 respectively. We can now present a generalization of the optimization program (2.30) for 2-player Nonzero sum games as proposed in [53] based on KKT conditions.

Lemma 2.1 (Necessary and Sufficient condition for Equilibrium point [53]). *Define scalars V^1, V^2. A necessary and sufficient condition that $(\pi^{1\star}, \pi^{2\star})$ be an equilibrium point of (2.31) is that it is a solution of the following bilinear programming problem*

$$\min_{\pi^1, \pi^2, V^1, V^2} \quad \sum_{u^2 \in U^2} \sum_{u^1 \in U^1} \pi^1(u^1)(c^1(u^1, u^2) + c^2(u^1, u^2))\pi^2(u^2) - V^1 - V^2 \tag{2.33a}$$

$$s.t. \quad \sum_{u^2 \in U^2} c^1(u^1, u^2)\pi^2(u^2) - V^1 \leq 0, \tag{2.33b}$$

$$\sum_{u^1 \in U^1} c^2(u^1, u^2)\pi^1(u^1) - V^2 \leq 0, \tag{2.33c}$$

$$\sum_{u^1 \in U^1} \pi^1(u^1) = 1, \pi^1(u^1) \geq 0, \ \forall u^1 \in U^1, \tag{2.33d}$$

$$\sum_{u^2 \in U^2} \pi^2(u^2) = 1, \pi^2(u^2) \geq 0, \ \forall u^2 \in U^2. \tag{2.33e}$$

Furthermore for the equilibrium pair $(\pi^{1\star}, \pi^{2\star})$ and equilibrium payoffs $(V^{1\star}, V^{2\star})$,

$$\sum_{u^2 \in U^2} \sum_{u^1 \in U^1} \pi^{1\star}(u^1)(c^1(u^1, u^2) + c^2(u^1, u^2))\pi^{2\star}(u^2) - V^{1\star} - V^{2\star} = 0.$$

N-player nonzero-sum games are solved using nonlinear complementarity programs which become impractical to solve as the number of players or the number of actions are increased [79]. As mentioned earlier, it was proven that no efficient algorithm exists for solving N-player nonzero-sum games [20]. However, a sample Correlated equilibrium point (among the set of Correlated Equilibrium points) can be efficiently computed

using a simple Linear feasibility program as follows,

$$\text{Find} \quad \pi \tag{2.34a}$$

$$\text{s.t.} \quad \sum_{u^{-i} \in U^{-i}} \pi(u)[c^i(u) - c^i(\tilde{u}^i, u^{-i})] \leq 0, \ \forall u^i, \tilde{u}^i \in U^i, \forall i \in \{1, \cdots, N\}, \tag{2.34b}$$

$$\sum_u \pi(u) = 1, \tag{2.34c}$$

$$\pi(u) \geq 0, \ \forall u \in U, \tag{2.34d}$$

An objective function can be added to the feasibility program (as long as it is Linear in π). A common objective function choice is the maximum welfare or minimal sum of costs to players i.e. $\min_\pi \sum_{u \in U} \pi(u) \sum_{i \in \{1, \cdots, N\}} c^i(u)$ or one can choose to minimize any convex combination of all players costs. Another can be to minimize the maximal cost attained among all players.

We shall now turn our attention to solving Markov games and an excellent overview is provided in the book [23] and the survey paper [71]. The computational solutions for such games can be classified broadly as a Mathematical program (such as a Linear program or a Nonlinear program) or an iterative solution scheme based on Dynamic programming (such as Value iteration i.e. Shapley's Algorithm). Generally, researchers in the past preferred Linear program as the ideal solution technique as by solving a single optimization problem, one can obtain the optimal value vector as well as the optimal policy. However, iterative methods are preferred in modern literature as they are more amenable to reinforcement learning methods (two almost contradictory statements regarding whether the linear program or dynamic programming should be used for solving Markov decision problems can be found on page 22 of the book [23] and page 87-88 of [86]). In the sequel, for a simplified exposition, we assume that the number of actions available at each state for each player is equal without loss of generality.

Mathematcical programming based procedures

We begin by stating an optimization problem for solving zero-sum Markov games that can be seen as an extension of (2.30). Unfortunately, unlike the optimization problem (2.30), the optimization problem for solving zero-sum Markov games is a nonlinear (specifically bilinear) program. Thereafter, we shall summarize iterative methods such as Shapley's Algorithm which form the foundations for both Markov games and Reinforcement learning (as Value iteration and policy iteration routines are based on this

algorithm). Zero-sum Markov games are solved using the following nonlinear program,

$$\min_{\pi^i(x,u^i),V^i(x)} \quad \sum_{i=1}^{|X|} V^i(x), \tag{2.35a}$$

$$\text{s.t.} \quad \sum_{u^i \in U^i} \pi^i(x,u^i) \left[c^i(x,u^i,u^{-i}) + \sum_{x' \in X} p(x' \mid x,u^i,u^{-i}) V^i(x') \right] \leq V^i(x), \quad \forall u^{-i} \in U^{-i}, \forall x \in X, \tag{2.35b}$$

$$\sum_{u^i \in U^i} \pi^i(x,u^i) = 1, \ \forall x \in X, \tag{2.35c}$$

$$\pi^i(x,u^i) \geq 0, \ \forall u^i \in U^i, \forall x \in X. \tag{2.35d}$$

Notice that the constraint (2.35b) imposes the optimality equation (2.25) and is the source of nonlinearity as it is bilinear in decision variables $\pi^i(x,u^i)$ and $V^i(x)$. Under certain conditions, the bilinear term can be avoided such as in so-called single controller discounted games where the transition dynamics are only affected by one player i.e. $p(x' \mid x,u^i,u^{-i}) = p(x' \mid x,u^{-i})$. This reduces the constraint (2.35b) to

$$\sum_{u^i \in U^i} \pi^i(x,u^i) \left[c^i(x,u^i,u^{-i}) \right] + \sum_{x' \in X} p(x' \mid x,u^{-i}) V^i(x') \leq V^i(x),$$

which is linear in decision variables $\pi^i(x,u^i)$ and $V^i(x)$. Other classes of games for which this is possible are the separable costs (rewards) state-independent transition stochastic games and switching controller stochastic games. Such games obey the ordered field property that can be roughly stated as, given the data of a stochastic game belonging to an ordered field (such as the field of rational numbers), the solution lies in the same ordered field (see references [23] and [71]). The significance of games having the ordered field property is that their solutions can be computed using a finite set of elementary operations (addition, subtraction, multiplication, and division by a nonzero number) as the field is closed under elementary operations. Therefore, numerical procedures can be developed for computing solutions of such games [83]. For nonzero-sum games, the mathematical program is more complicated as we cannot use saddle point inequalities, and therefore both the player's strategies and value functions are computed using a single optimization problem that can be viewed as an extension of (2.33). Consider the following decision variables $\eta = (\pi^i(x,u^i), \pi^{-i}(x,u^{-i}), V^i(x), V^{-i}(x))$ of dimensions

$| U^i | + | U^{-i} | +2 | X |$ and let $\pi = (\pi^i(x, u^i), \pi^{-i}(x, u^{-i}))$ denote the joint policy.

$$\min_{\eta} \sum_{i=1}^{N} \sum_{j=1}^{|X|} \left[V_\pi^i(x) - \pi \left[c^i(x, u^i, u^{-i}) + \sum_{x' \in X} p(x' \mid x, u^i, u^{-i}) V_\pi^i(x') \right] \right], \tag{2.36a}$$

$$\text{s.t. } \pi^i(x, u^i) \left[c^i(x, u^i, u^{-i}) + \sum_{x' \in X} p(x' \mid x, u^i, u^{-i}) V_\pi^i(x') \right] \leq V^i(x), \forall x \in X, \tag{2.36b}$$

$$\left[c^i(x, u^i, u^{-i}) + \sum_{x' \in X} p(x' \mid x, u^i, u^{-i}) V_\pi^i(x') \right] \pi^{-i}(x, u^{-i}) \leq V^{-i}(x), \forall x \in X, \tag{2.36c}$$

$$\sum_{u^i \in U^i} \pi^i(x, u^i) = 1, \pi^i(x, u^i) \geq 0, \ \forall u^i \in U^i, \forall x \in X \tag{2.36d}$$

$$\sum_{u^{-i} \in U^{-i}} \pi^{-i}(x, u^{-i}) = 1, \pi^{-i}(x, u^{-i}) \geq 0, \ \forall u^{-i} \in U^{-i}, \forall x \in X. \tag{2.36e}$$

A class of games known as the Additive reward (cost) Additive transitions obey the ordered field property [72] and iterative methods can be used to find a solution for them although it is not known whether a linear program can be used to obtain the value vector as well as the policy using a single optimization procedure.

Dynamic programming based iterative procedures

The optimality equation can also be solved iteratively using Shapley's Algorithm and that forms the basis for Value Iteration schemes. In the sequel, ε represents the error

Algorithm 1 Shapley's Value Iteration scheme for Markov games [71, 78]

Input: Given player $i \in \{1, \cdots, N\}$, initialize $V_0^i(x)$ for all $x \in X$
while $\left\| \mathbf{V}_t^i - \mathbf{V}_{t-1}^i \right\| > \varepsilon$ **do**
  for each $x \in X$ **do**
    $V_t^i(x) \leftarrow \text{val} \left[c_t^i(x, u_t^i, u_t^{-i}) + \sum_{x' \in X} p(x' \mid x, u_t^i, u_t^{-i}) V_{t-1}^i(x') \right]$
    $t \leftarrow t + 1$
  end for
end while
Obtain $\pi^{i^\star}(x) \leftarrow \arg\min_{\pi^i} V^i(x)$ for all $x \in X$

tolerance parameter that governs the termination condition for the algorithm. Recall that $\mathbf{V}$ represents the value vector with components representing the value for each given initial state and that the val$[\cdot]$ operator represents the value of the game and the

Optimality Equation defined in (2.25). Using Algorithm 1, players can calculate their value vectors in a decentralized setting given that they know the transition probability matrix, and their cost matrix, and can observe the states and actions of all players. However, it should be noted that Algorithm 1 does not utilize optimal strategy at each iteration and therefore can be quite slow in convergence [68, 70]. This limitation was removed in the following Algorithm due to Hoffman and Karp [41] by extending the Policy Iteration algorithm for the Markov decision process [42] to Markov games by using the Best Response solution (2.6) at each iteration on the approximately optimal policy by the opponent. However, this algorithm is only applicable to 2-player zero-sum Markov games. A Q-learning variant of Algorithm 2 is proposed by the author for safety-

Algorithm 2 Hoffman and Karp Algorithm for Markov games [41, 71]

1: **Input**: Given player 1, initialize $V_0^1(x)$ and $\pi^1(x) = \frac{1}{|U^1|}$ for all $x \in X$.
2: **while** $\left\|\mathbf{V}_t^1 - \mathbf{V}_{t-1}^1\right\| > \varepsilon$ **do**
3: **for** each $x \in X$ **do**
4: $\pi_t^2(x) \leftarrow \arg\max_{\pi^2} V_{t-1}^1(x)$
5: Calculate best response value,

$$V_t^1(x) \leftarrow \min_{\pi^i} \left[c_t^1(x, u_t^1, \pi_t^2(x)) + \sum_{x' \in X} p(x' \mid x, u_t^1, u_t^2) V_{t-1}^1(x') \right],$$
$$\text{where } c_t^1(x, u_t^1, \pi_t^2(x)) = \mathbb{E}_{\pi_t^2(x)} \left[c_t^1(x, u_t^1, u_t^2) \right]$$

6: **end for**
7: $t \leftarrow t + 1$
8: **end while**

constrained Markov Decision Process in Paper C. The author considers the Lagrange multiplier associated with the safety constraint as the maximizing player and the control policy as the minimizing player. It should be noted that the minimizing player in Algorithm 2 essentially solves a degenerate Markov Decision Process (parameterized by the strategy of player 2) at each iteration. This can obviously lead to high computational costs depending on the number of states and control actions in the Markov game. This challenge was resolved in the following algorithm by Pollatschek and Avi-Itzhak in [69] which can be viewed as a generalization of the Policy iteration Algorithm. We overload the notion $\text{val}_\pi[\cdot]$ by specifying the joint policy with whose respect to the value of the game is calculated by val[·] operator. While Algorithm 3 is significantly faster compared to Algorithms 1, 2, the primary drawback of Algorithm 3 is that the convergence is not always guaranteed [9, 71, 89] and [89] proposed a counterexample for Algorithm 3. Modifications to Algorithm 3 that ensure convergence in all cases while preserving the speed of convergence were proposed in [89] where the val[·] operator is calculated up

Algorithm 3 Pollatschek and Avi-Itzhak Algorithm for Markov games [69, 71]

Input: For each player $i \in \{1, \cdots, N\}$, initialize $V_0^i(x)$ and $\pi^i(x) = \frac{1}{|U^i|}$ for all $x \in X$.
while $\left\|\mathbf{V}_t^i - \mathbf{V}_{t-1}^i\right\| > \varepsilon$ **do**
  for each $x \in X$ **do**
    $V_t^i(x) \leftarrow \text{val}_{\pi_{t-1}}\left[c_t^i(x, u_t^i, u_t^{-i}) + \sum_{x' \in X} p(x' \mid x, u_t^i, u_t^{-i}) V_{t-1}^i(x')\right]$
    $\pi_t^i(x) = \arg\min_{\pi^i} V_t^i(x)$
  end for
  $t \leftarrow t + 1$
end while

to certain time steps in the future at each iteration (the so-called Generalized Policy iteration in Reinforcement learning literature [86] which is quite similar to the idea of Receding Horizon Control). The paper [24] presented a thorough analysis of these algorithms and the main idea is as follows. Recall that, Shapley's Algorithm 1 essentially finds the fixed point of the Optimality equation (2.25) and thus, Algorithm 3 essentially minimizes the distance between the optimal val[·] operator and the value function at a given iteration (or the so-called Bellman residual in the literature [67]). More precisely, let

$$T(V^i, x) := \text{val}\left[c^i(x, u^i, u^{-i}) + \sum_{x' \in X} p(x' \mid x, u^i, u^{-i}) V_{\pi^\star}^i(x')\right], \quad \forall x \in X. \tag{2.37}$$

and $\mathbf{T}(\mathbf{V}^i, x)$ be the operator (2.37) for all starting states $x \in X$. Then, finding the fixed point of val[·] operator for all starting states is essentially equivalent to finding the roots of the following equation (or minimizing Bellman residual),

$$\Phi(\mathbf{V}^i, x) := \mathbf{T}(\mathbf{V}^i, x) - \mathbf{V}^i.$$

Under certain standard assumptions on uniqueness and continuity of the partial derivatives of $\Phi(\mathbf{V}^i, x)$ with respect to $\mathbf{V}^i$, it is shown in [23, 24] that the Algorithm 3 essentially minimizes $\Phi(\mathbf{V}^i, x)$ and is equivalent to classical Newton's method for finding roots. In order to ensure convergence, in all cases, Filar and Tolwinski propose an algorithm that minimizes $J(V^i, x) = 0.5[\Phi(\mathbf{V}^i, x)]^T \Phi(\mathbf{V}^i, x)$ with the step-size selected such that $J_{t+1}(V^i, x) < J_t(V^i, x)$ instead of just minimizing $\Phi(\mathbf{V}^i, x)$ as done in Algorithm 3 and call it Modified Newton's Method [24], it turns out that in the case of 2-player zero-sum Markov games the algorithm by Filar and Tolwinski [24] obtains a global solution as proved in [24]. Further details on this algorithm and the associated Reinforcement learning algorithms can be found in [67].

3 Reinforcement Learning and Learning in Game-theoretic setting

In this section, we shall summarize some reinforcement learning techniques and learning techniques so that players can compute optimal strategies in the sense of a game theoretic solution. In contrast to the previous section on the computation of a game-theoretic solution, the players will not have access to the functional form of their cost functions (or utilities) i.e. the payoff matrix is not accessible to the players furthermore in the case of Markov games, players shall not have access to the underlying transition dynamics of the system i.e. they cannot plan which states to visit in future. In recent years, Reinforcement Learning has emerged as a dominant paradigm in machine learning owing to the success of AlphaGo [82] which is a 2-player zero-sum Markov Game. However, learning in the presence of other players (who are also learning) is a fundamentally very challenging problem owing to the non-stationarity of the environment from the perspective of any given player (owing to the presence of other learners in the environment). We introduce the following assumption that is similar to the rationality assumption 2.1 on the rationality of the players.

Assumption 3.1 (No binding commitment) Each decision-maker seeks to learn the equilibrium strategy and is not allowed to commit to one control action irrespective of what the opponent does.

The assumption 3.1 is required to ensure that during the learning phase, each player does not *bully* other players to a desirable equilibrium point by being a *poor* learner. This is illustrated by the following example.

3.1 Example (Learning in game of chicken [80]) Consider the game of chicken from the previous section 2.2 (with the game matrix repeated here for convenience) and recall the two Pure strategy Nash equilibrium points $(Wait, Go)$ and $(Go, Wait)$.

		Player B	
		$Wait$	Go
Player A	$Wait$	$(0, 0)$	$(0, 1)$
	Go	$(1, 0)$	$(-100, -100)$

Assume now that the game is repeated T times with the task of players being to learn the correlated equilibrium distribution. Suppose that player A chooses to make a binding commitment to take the action Go irrespective of what player B does. Then after finite repetitions of the game player B will learn to always take action

Wait as the outcome (Go, Go) is disastrous for player B. Observe that in this case the expected payoff of player A will be 1 which is higher than any expected payoff obtained by using correlated equilibrium distribution. Thus, by essentially being a *poor* learner, player A essentially *bullies* player B to an outcome that is more favorable to player A.

3.1 Learning in Repeated games

This subsection will summarize some of the techniques for learning in repeated games and is based on the books [29, 35] and papers [8, 13, 28, 33, 54, 66, 77]. We consider the game $\Gamma = (N, U, C)$ as defined in subsection 2.1 and the game is played repeatedly by the players. Generally, learning in games can be classified as model-based or model-free [80]. The model-based setting involves creating an empirical model of the opponent's (unknown) strategy based on the empirical history of play and then playing the best response solution (see definition 2.4) to the empirical model of the opponent's strategy. The model-free setting involves direct computation of the strategy based on the empirical history without explicitly building a model of the opponent's (unknown) strategy. We illustrate these two approaches by stating two well-known algorithms from both the model-based and model-free approaches. These two algorithms are Fictitious play and Regret minimization respectively and are the subject of the following subsections.

Fictitious play in Repeated Games

Consider a 2-player repeated game setup where each player can observe the past actions of the opponent but is unaware of the cost function and the mixed strategy of the opponent. The fictitious play setup assumes that each player can learn the empirical distribution of the opponent's mixed strategy (or the beliefs over pure actions) by keeping a running average of past actions (so-called empirical beliefs over pure actions). This assumption is true if the opponent has a fixed mixed strategy. However, what happens if both players assume that the other player is keeping a constant mixed strategy? Do the empirical mixed strategies converge to any meaningful Nash equilibrium of the game? These questions are studied in the literature on fictitious play [2, 8, 29, 77]. In the sequel, we shall consider player i's perspective (by symmetricity players are interchangeable). Given the history of the play,

$$h_t^i = [u_0^i, u_0^{-i}, c^i(u_0^i, u_0^{-i}), \cdots, c^i(u_t^i, u_t^{-i})],$$

let ω_t^{-i} denote the running average of the past control actions u^{-i} of player 2 and it is updated as follows,

$$\omega_{t+1}^{-i}(u_t^{-i}) = \omega_t^{-i}(u_t^{-i}) + \frac{1}{t+1}(u_t^{-i} - \omega_t^{-i}(u_t^{-i})). \tag{2.38}$$

Based on the update (2.38), player i updates the mixed strategy π^i_{t+1} by taking the best response (see definition 2.4) to the distribution (2.38) i.e. $BR^i(\omega^{-i}_{t+1})$ by solving the following linear program,

$$\min_{\pi^i, V^i} \quad V^i \tag{2.39a}$$

$$\text{s.t.} \quad \sum_{u^i \in U^i} \sum_{u^{-i} \in U^{-i}} \pi^i(u^i) c^i(u^i, u^{-i}) \omega^{-i}_{t+1}(u^{-i}) \leq V^i, \tag{2.39b}$$

$$\sum_{u^i} \pi^i(u^i) = 1, \tag{2.39c}$$

$$\pi^i(u^i) \geq 0, \ \forall u^i \in U^i, \tag{2.39d}$$

From (2.38) and linear program (2.39), it is evident that Fictitious play can be used as a decentralized way for players to compute their optimal strategies. Unfortunately, the following theorem shows that the empirical strategies generated by this procedure fail to converge to a Nash equilibrium unless we restrict our attention to very specific repeated games.

Theorem 3.1 (Convergence requirement for Fictitious play [5, 8, 29]). *The empirical strategies generated by (2.38) and linear program (2.39) converge to the Nash equilibrium if the game Γ satisfies one of the following properties*

- Γ *is a 2-player zero-sum game (or strategically equivalent to one) i.e.* $\Gamma = (2, U^1, U^2, c, -c)$,
- Γ *is a cooperative game where all players are cooperatively trying to minimize the global cost function* $c : U \mapsto \mathbb{R}$, *i.e.* $\Gamma = (N, U^1, \cdots, U^N, c)$,
- Γ *is a 2-player nonzero-sum game where the control actions available to at least one player is almost 2 i.e.* $\Gamma = (2, U^1, 2, c^1, c^2)$ *or* $\Gamma = (2, 2, U^2, c^1, c^2)$.

The papers [2, 77] suggest modifications to fictitious play for ensuring convergence by interpreting it as a stochastic approximation scheme and stabilizing the associated continuous time differential equation. However, it is not discussed whether the modified scheme is applicable to all classes of games, particularly nonzero-sum games. Furthermore, given the negative result by [34] regarding the nonexistence of simple, general, and uncoupled (or decentralized) dynamics for ensuring convergence to Nash equilibrium for any general game, it seems unlikely that the scheme will work for general nonzero-sum games. Therefore, we focus our attention to approachability based learning rules such as regret matching and calibration that ensure convergence to the notion of correlated equilibrium instead of Nash equilibrium.

Approachability-based learning in Repeated games: Regret Matching

Consider the game Γ which is repeated with each player observing partial histories (2.11), and let us define the regret matrix of dimensions $\mid U^i \mid \times \mid U^i \mid$ for player i, as $\left[IR_t^i(a,b)\right]_{a=1,\cdots\mid U^i\mid, b=1,\cdots\mid U^i\mid}$, with elements defined as follows,

$$IR_t^i(a,b) = \begin{cases} c_t^i(b, u_t^{-i}) - c_t^i(a, u_t^{-i}), & \text{if } u_t^i = a, \\ 0, & \text{otherwise.} \end{cases} \tag{2.40}$$

Suppose $u_t^i = a$, than let $IR_t^i(a,b)_+ := \max\{IR_t^i(a,b), 0\}$ denote the positive regrets with respect to a and $\mu := \sum_{b \in U^i} IR_t^i(a,b)_+$ be the sum of all positive regrets. The regret matching algorithm due to [32] proposes that given the partial history h_t^i, each player i at time $t+1$ updates the mixed strategy π_{t+1}^i for all actions $b \in U^i$ in proportion to the past regrets as follows,

$$\pi_{t+1}^i(b) = \begin{cases} \frac{1}{\mu} IR_t^i(a,b), & \text{if } a \neq b, \\ 1 - \sum\limits_{b' \neq a} \frac{1}{\mu} IR_t^i(a,b'), & \text{if } a = b, \\ \frac{1}{\mid U^i \mid}, & \text{if } \mu = 0 \end{cases} \tag{2.41}$$

Observe that the regret-matching algorithm (2.40), (2.41) minimizes the internal regret as the matrix $IR_t^i(a,b)$ quantifies the counterfactual difference in costs incurred by player i if he/she had played action b every time he/she played action a. The regrets are counterfactual since we are assuming that all other players would have played the same sequence of actions despite player i switching actions based on past regrets. Despite this shortcoming in the notion of regret, [32] proved that if every player uses the regret-matching algorithm then the joint distribution will converge to the set of correlated equilibrium of the game Γ (2.7). The proof relied on Blackwell's Approachability theorem where the regrets were shown to approach the negative orthant in the action space $\mathbb{R}^{\mid U^i \mid}$. Note that the regret matching algorithm does not have any assumptions on the strategies of the opponents unlike fictitious play and this allows for a decentralized implementation of the same and consequently, it is easily scalable for large systems. The regret matching algorithm is a simple and effective decentralized scheme for ensuring convergence to a correlated equilibrium in repeated games but, it does not guarantee convergence to a specific Nash equilibrium among the set of correlated equilibrium. However, in the case of the existence of a unique Nash equilibrium point such as the zero-sum game of Rock-paper-scissor, the regret matching algorithm converges to the mixed strategy Nash equilibrium (see fig. 2.3, 2.4). A weaker notion of regret that is commonly studied is the so-called external regret where player i compares the action chosen at time t to all other fixed actions i.e. instead of the matrix (2.40), we minimize the following vector of dimension $\mid U^i \mid$,

$$ER_t^i(a) := [c_t^i(1, u_t^{-i}) - c_t^i(a, u_t^{-i}), \cdots, c_t^i(\mid U^i \mid, u_t^{-i}) - c_t^i(a, u_t^{-i})]. \tag{2.42}$$

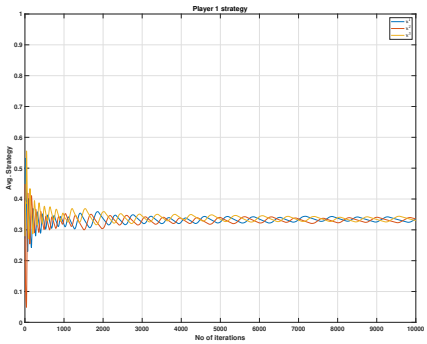


Fig. 2.3: Mixed strategies for Player 1 obtained from Regret matching algorithm (2.40), (2.41).

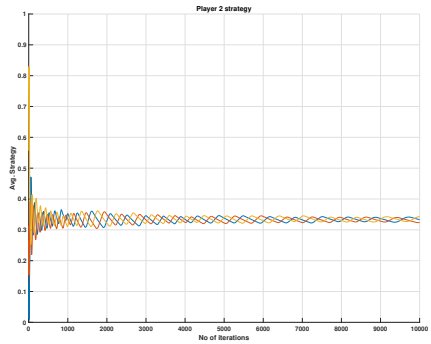


Fig. 2.4: Mixed strategies for Player 2 obtained from Regret matching algorithm (2.40), (2.41).

If all players minimize their respective external regrets then the joint distribution converges to the set of coarse-correlated equilibrium (2.8) [18, 31]. Lastly, we note that the regret matching algorithm can be interpreted as sequential predictions of future costs based on the past behavior of the opponents, and this interpretation connects the regret matching algorithm to calibration [66].

3.2 Learning in Markov games

There is a vast literature [15, 39, 40, 92] on this topic owing to the recent success of AlphaGo (and other general variants that learn the rules of the game as well) [81, 82] and we do not cover all the algorithms in this chapter. However, we state a general principle behind (most of) the algorithms as follows. Consider the optimality equation for Markov games (2.25) and off-policy setting for reinforcement learning where given a baseline joint policy (that is not the equilibrium or optimal policy but just a sufficiently exploratory policy), each player aims to learn their ε-equilibrium policy in some finite number of episodes (parametrized by desired approximation ε; see [21, 64, 92] for finite bounds). The basic idea is to iteratively solve (or learn) this equation by trying out different actions. More precisely, given some baseline joint policy π, the optimality equation can be reformulated as,

$$\hat{V}^i_{t+1}(x) = c^i(x_t, u^i_t, u^{-i}_t) + \sum_{x' \in X} p(x' \mid x_t, u^i_t, u^{-i}_t)\hat{V}^i_t(x'), \quad \forall x \in X, \quad u^i_t, u^{-i}_t \sim \pi, \tag{2.43}$$

where $\hat{V}^i_{t+1}(x)$ is the estimate of the value vector at time $t+1$. In the reinforcement learning setting, the transition kernel $p(x' \mid x, u^i, u^{-i})$ is unknown, and therefore it is proposed in [11, 88, 91] to approximate the value vector $\hat{V}^i_{t+1}(x)$ for each control

action using stochastic approximation. This is done by augmenting the value vector $\hat{V}^i_{t+1}(x)$ with control action arguments by introducing the notation $Q^i_{t+1}(x, u^i, u^{-i})$ and updating it using the following scheme,

$$Q^i_{t+1}(x_t, u^i_t, u^{-i}_t) = (1-\alpha_t)Q^i_t(x_t, u^i_t, u^{-i}_t) + \alpha_t(c^i(x_t, u^i_t, u^{-i}_t) + \hat{V}^i_t(x_{t+1})), \tag{2.44}$$

$$\hat{V}^i_t(x_{t+1})) = \text{val}[Q^i_t(x_{t+1}, u^i_t, u^{-i}_t)], \tag{2.45}$$

where α_t is the learning rate. The following assumption is required for ensuring convergence of the scheme (2.44) such that the argument of the val$l[\cdot]$ operator should satisfy the desired equilibrium condition for every player.

Assumption 3.2 (Convergence requirements) The learning rate α_t must satisfy Robbins-Monroe assumptions [11] i.e.,

$$\sum_{t=0}^{\infty} \alpha_t = \infty, \quad \sum_{t=0}^{\infty} \alpha_t^2 < \infty. \tag{2.46}$$

Furthermore, the baseline policy π needs to ensure that every state-joint action pair is visited sufficiently often (again parameterized by ε; see [21, 64, 92] for more details) often.

Depending on the definition of the val$l[\cdot]$ operator, different equilibrium solutions can be obtained as the val$l[\cdot]$ operator is essentially the solution of a static game. For example, if the val$l[\cdot]$ operator corresponds to the min max operator then we have the Minimax solution of the game (the so-called Minimax Q-learning and Friend or Foe Q-learning algorithms [50, 51]), if it corresponds to Nash equilibrium, then we have the so-called Nash Q-learning algorithm [43] and lastly, if the val$l[\cdot]$ operator corresponds to the Correlated equilibrium as in (2.29), then we have the so-called Correlated Q-learning algorithm [30].

4 Conclusion

This chapter provides a technical background of most concepts studied during the PhD project. It provides a snapshot of Game Theory and Learning in the presence of other players. We motivate the reason for using Game theory as a generalization of a optimization problems to multiple decision makers where the cost incurred by any given decision maker is affected not only by his/her decisions but also by the decisions of every other decision maker as well. We highlight some of the solution concepts offered

by Game theory that have been used in this PhD thesis. This by no means an exhaustive list of Game theoretic solutions as we have not covered solution concepts such as Bayesian Nash Equilibrium or Shapley value for Potential games and many more. The solution concepts of particular focus in this PhD thesis are Minimax solution, Correlated Equilibrium and Blackwell's Approachability theorem for multi-objective games. We also focus on Markov games that have recently gained a lot of attention from Reinforcement learning community as a framework for multi-agent Reinforcement Learning and we highlight the fundamental challenges involved with learning in the presence of other players such as bullying by weak learners.

References

[1] J. Abernethy, P. L. Bartlett, and E. Hazan, "Blackwell approachability and no-regret learning are equivalent," in *Proceedings of the 24th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, 2011, pp. 27–46.

[2] G. Arslan and J. S. Shamma, "Distributed convergence to nash equilibria with local utility measurements," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 2. IEEE, 2004, pp. 1538–1543.

[3] R. J. Aumann, "Subjectivity and correlation in randomized strategies," *Journal of mathematical Economics*, vol. 1, no. 1, pp. 67–96, 1974.

[4] ——, "Correlated equilibrium as an expression of bayesian rationality," *Econometrica: Journal of the Econometric Society*, pp. 1–18, 1987.

[5] T. Basar *et al.*, "Lecture notes on non-cooperative game theory," *Game Theory Module of the Graduate Program in Network Mathematics*, pp. 3–6, 2010.

[6] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1998.

[7] T. Başar and G. Zaccour, *Handbook of dynamic game theory*. Springer, 2018.

[8] M. Benaım and M. W. Hirsch, "Mixed equilibria and dynamical systems arising from fictitious play in perturbed games," *Games and Economic Behavior*, vol. 29, no. 1-2, pp. 36–72, 1999.

[9] D. Bertsekas, "Distributed asynchronous policy iteration for sequential zero-sum games and minimax control," *arXiv preprint arXiv:2107.10406*, 2021.

[10] D. Blackwell, "An analog of the minimax theorem for vector payoffs." *Pacific Journal of Mathematics*, vol. 6, no. 4, pp. 1–8, 1956.

[11] V. Borkar, *Stochastic Approximation: A Dynamical Systems Viewpoint: Second Edition*, ser. Texts and Readings in Mathematics. Hindustan Book Agency, 2022. [Online]. Available: https://books.google.dk/books?id=k_ChEAAAQBAJ

[12] S. Boyd and L. Vandenberghe, "Convex optimization, cambridge univ," *Press, UK*, 2004.

[13] G. W. Brown, "Iterative solution of games by fictitious play," *Act. Anal. Prod Allocation*, vol. 13, no. 1, p. 374, 1951.

[14] N. Brown and T. Sandholm, "Superhuman ai for heads-up no-limit poker: Libratus beats top professionals," *Science*, vol. 359, no. 6374, pp. 418–424, 2018.

[15] L. Buşoniu, R. Babuška, and B. De Schutter, "Multi-agent reinforcement learning: An overview," *Innovations in multi-agent systems and applications-1*, pp. 183–221, 2010.

[16] Y. Cai, C. Daskalakis, H. Luo, C.-Y. Wei, and W. Zheng, "Tractable local equilibria in non-concave games," *arXiv preprint arXiv:2403.08171*, 2024.

[17] P. J. Campo and M. Morari, "Robust model predictive control," in *1987 American control conference*. IEEE, 1987, pp. 1021–1026.

[18] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[19] C. Daskalakis, "Non-concave games: A challenge for game theory's next 100 years," *Cowles Preprints*, 2022.

[20] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou, "The complexity of computing a nash equilibrium," *Communications of the ACM*, vol. 52, no. 2, pp. 89–97, 2009.

[21] C. Daskalakis, N. Golowich, and K. Zhang, "The complexity of markov equilibrium in stochastic games," in *The Thirty Sixth Annual Conference on Learning Theory*. PMLR, 2023, pp. 4180–4234.

[22] G. Farina, C. Kroer, and T. Sandholm, "Faster game solving via predictive blackwell approachability: Connecting regret matching and mirror descent," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 6, 2021, pp. 5363–5371.

[23] J. Filar and K. Vrieze, *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

[24] J. A. Filar and B. Tolwinski, "On the algorithm of pollatschek and avi-ltzhak," in *Stochastic Games and Related Topics: In Honor of Professor LS Shapley*. Springer, 1991, pp. 59–70.

[25] S. D. Flåm, "Equilibrium, evolutionary stability and gradient dynamics," *International Game Theory Review*, vol. 4, no. 04, pp. 357–370, 2002.

[26] F. Forges, "Correlated equilibria and communication in games," *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models*, pp. 107–118, 2020.

[27] D. P. Foster and R. V. Vohra, "Calibrated learning and correlated equilibrium," *Games and Economic Behavior*, vol. 21, no. 1-2, pp. 40–55, 1997.

[28] ——, "Asymptotic calibration," *Biometrika*, vol. 85, no. 2, pp. 379–390, 1998.

[29] D. Fudenberg and D. K. Levine, *The theory of learning in games*. MIT Press, 1998, vol. 2.

[30] A. Greenwald, K. Hall, and R. Serrano, "Correlated q-learning," in *ICML*, vol. 3, 2003, pp. 242–249.

[31] J. Hannan, "Approximation to bayes risk in repeated play," *Contributions to the Theory of Games*, vol. 3, pp. 97–139, 1957.

[32] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[33] ——, "Regret-based continuous-time dynamics," *Games and Economic Behavior*, vol. 45, no. 2, pp. 375–394, 2003.

[34] ——, "Uncoupled dynamics do not lead to nash equilibrium," *American Economic Review*, vol. 93, no. 5, pp. 1830–1836, 2003.

[35] ——, *Simple adaptive strategies: from regret-matching to uncoupled dynamics*. World Scientific, 2013, vol. 4.

[36] M. Haviv, "On constrained markov decision processes," *Operations research letters*, vol. 19, no. 1, pp. 25–28, 1996.

[37] E. Hazan *et al.*, "Introduction to online convex optimization," *Foundations and Trends® in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2016.

[38] E. Hazan and K. Singh, "Introduction to online nonstochastic control," *arXiv preprint arXiv:2211.09619*, 2022.

[39] P. Hernandez-Leal, M. Kaisers, T. Baarslag, and E. M. de Cote, "A survey of learning in multiagent environments: Dealing with non-stationarity," *arXiv preprint arXiv:1707.09183*, 2017.

[40] P. Hernandez-Leal, B. Kartal, and M. E. Taylor, "A survey and critique of multiagent deep reinforcement learning," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 6, pp. 750–797, 2019.

[41] A. J. Hoffman and R. M. Karp, "On nonterminating stochastic games," *Management Science*, vol. 12, no. 5, pp. 359–370, 1966.

[42] R. A. Howard, *Dynamic Probabilistic Systems, Volume I: Markov Models*. Courier Corporation, 2012, vol. 1.

[43] J. Hu and M. P. Wellman, "Nash q-learning for general-sum stochastic games," *Journal of machine learning research*, vol. 4, no. Nov, pp. 1039–1069, 2003.

[44] A. Jaśkiewicz and A. S. Nowak, "Non-zero-sum stochastic games," *Handbook of dynamic game theory*, pp. 1–64, 2018.

[45] L. Kallenberg, "Markov decision processes," *Lecture Notes. University of Leiden*, vol. 428, 2011.

[46] O. Karaca, P. G. Sessa, A. Leidi, and M. Kamgarpour, "No-regret learning from partially observed data in repeated auctions," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14–19, 2020.

[47] A. R. Karlin and Y. Peres, *Game theory, alive*. American Mathematical Soc., 2017, vol. 101.

[48] J.-M. Lasry and P.-L. Lions, "Mean field games," *Japanese journal of mathematics*, vol. 2, no. 1, pp. 229–260, 2007.

[49] M. Le Treust and M. Le Treust, "A repeated game formulation of energy-efficient decentralized power control," *IEEE Transactions on Wireless Communications*, vol. 9, no. 9, pp. 2860–2869, 2010.

[50] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Machine learning proceedings 1994*. Elsevier, 1994, pp. 157–163.

[51] M. L. Littman *et al.*, "Friend-or-foe q-learning in general-sum games," in *ICML*, vol. 1, no. 2001, 2001, pp. 322–328.

[52] J. Löfberg, *Minimax approaches to robust model predictive control*. Linköping University Electronic Press, 2003, vol. 812.

[53] O. L. Mangasarian and H. Stone, "Two-person nonzero-sum games and quadratic programming," *Journal of Mathematical Analysis and applications*, vol. 9, no. 3, pp. 348–355, 1964.

[54] S. Mannor, J. S. Shamma, and G. Arslan, "Online calibrated forecasts: Memory efficiency versus universality for learning in games," *Machine Learning*, vol. 67, pp. 77–115, 2007.

[55] R. Misra, C. S. Kallesøe, and R. Wisniewski, "Decentralized control of a water distribution network using repeated games," in *2023 27th International Conference on Methods and Models in Automation and Robotics (MMAR)*. IEEE, 2023, pp. 181–186.

[56] R. Misra, R. Wisniewski, and C. S. Kallesøe, "Approximating solution of stochastic differential games for distributed control of a water network," *IFAC-PapersOnLine*, vol. 55, no. 16, pp. 110–115, 2022.

[57] ——, "On bellman's principle of optimality and reinforcement learning for safety-constrained markov decision process," *arXiv preprint arXiv:2302.13152*, 2023.

[58] R. Misra, R. Wisniewski, C. S. Kallesøe, and M. L. Bujorianu, "Robust correlated equilibrium: Definition and computation," *arXiv preprint arXiv:2311.17592*, 2023.

[59] H. Moulin and J. P. Vial, "Strategically zero-sum games: the class of games whose completely mixed equilibria cannot be improved upon," *International Journal of Game Theory*, vol. 7, pp. 201–221, 1978.

[60] J. Nash, "Non-cooperative games," *Annals of mathematics*, pp. 286–295, 1951.

[61] J. F. Nash Jr, "Equilibrium points in n-person games," *Proceedings of the national academy of sciences*, vol. 36, no. 1, pp. 48–49, 1950.

[62] A. S. Nowak and T. E. Raghavan, "Existence of stationary correlated equilibria with symmetric information for discounted stochastic games," *Mathematics of Operations Research*, vol. 17, no. 3, pp. 519–526, 1992.

[63] F. Oliehoek, N. Vlassis *et al.*, "Dec-pomdps and extensive form games: equivalence of models and algorithms," *Ias technical report IAS-UVA-06-02, University of Amsterdam, Intelligent Systems Lab, Amsterdam, The Netherlands*, 2006.

[64] A. Ozdaglar, M. O. Sayin, and K. Zhang, "Independent learning in stochastic games," *arXiv preprint arXiv:2111.11743*, 2021.

[65] C. Papadimitriou and T. Roughgarden, "Computing correlated equilibria in multi-player games," *Journal of the ACM (JACM)*, vol. 55, no. 3, pp. 1–29, 2008.

[66] V. Perchet, "Approachability, regret and calibration: Implications and equivalences," *Journal of Dynamics and Games*, vol. 1, no. 2, pp. 181–254, 2014.

[67] J. Pérolat, "Reinforcement learning: The multi-player case," Ph.D. dissertation, Université de Lille 1-Sciences et Technologies, 2017.

[68] J. Perolat, J. Z. Leibo, V. Zambaldi, C. Beattie, K. Tuyls, and T. Graepel, "A multi-agent reinforcement learning model of common-pool resource appropriation," *Advances in neural information processing systems*, vol. 30, 2017.

[69] M. Pollatschek and B. Avi-Itzhak, "Algorithms for stochastic games with geometrical interpretation," *Management Science*, vol. 15, no. 7, pp. 399–415, 1969.

[70] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 1994.

[71] T. Raghavan and J. A. Filar, "Algorithms for stochastic games—a survey," *Zeitschrift für Operations Research*, vol. 35, no. 6, pp. 437–472, 1991.

[72] T. E. Raghavan, S. Tijs, and O. Vrieze, "On stochastic games with additive reward and transition structure," *Journal of Optimization Theory and Applications*, vol. 47, pp. 451–464, 1985.

[73] T. Roughgarden, "Algorithmic game theory," *Communications of the ACM*, vol. 53, no. 7, pp. 78–86, 2010.

[74] ——, "The price of anarchy in games of incomplete information," in *Proceedings of the 13th ACM Conference on Electronic Commerce*, 2012, pp. 862–879.

[75] J. Różycka-Tran, P. Boski, and B. Wojciszke, "Belief in a zero-sum game as a social axiom: A 37-nation study," *Journal of Cross-Cultural Psychology*, vol. 46, no. 4, pp. 525–548, 2015.

[76] A. Rubinstein, *Modeling bounded rationality*. MIT press, 1998.

[77] J. S. Shamma and G. Arslan, "Dynamic fictitious play, dynamic gradient play, and distributed convergence to nash equilibria," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 312–327, 2005.

[78] L. S. Shapley, "Stochastic games," *Proceedings of the national academy of sciences*, vol. 39, no. 10, pp. 1095–1100, 1953.

[79] Y. Shoham, K. Leyton-Brown *et al.*, "Multiagent systems," *Algorithmic, Game-Theoretic, and Logical Foundations*, 2009.

[80] Y. Shoham, R. Powers, and T. Grenager, "If multi-agent learning is the answer, what is the question?" *Artificial intelligence*, vol. 171, no. 7, pp. 365–377, 2007.

[81] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[82] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the game of go without human knowledge," *nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[83] S. Sinha, "Contribution to the theory of stochastic games," Ph.D. dissertation, Indian Statistical Institute-Kolkata, 1989.

[84] M. Sion, "On general minimax theorems." *Pacific J. Math.*, vol. 8, no. 4, pp. 171–176, 1958.

[85] E. Solan and N. Vieille, "Correlated equilibrium in stochastic games," *Games and Economic Behavior*, vol. 38, no. 2, pp. 362–399, 2002.

[86] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[87] T. Tao, *An introduction to measure theory*. American Mathematical Soc., 2011, vol. 126.

[88] J. N. Tsitsiklis, "Asynchronous stochastic approximation and q-learning," *Machine learning*, vol. 16, pp. 185–202, 1994.

[89] J. Van Der Wal, "Discounted markov games: Generalized policy iteration method," *Journal of Optimization Theory and Applications*, vol. 25, pp. 125–138, 1978.

[90] J. Von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton university press, 1944.

[91] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[92] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *arXiv preprint arXiv:1911.10635*, pp. 321–384, 2019.

[93] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th international conference on machine learning (icml-03)*, 2003, pp. 928–936.

Chapter 3

Markov chain approximation based sequential minimax control

Summary *This chapter summarizes the contributions of this dissertation on optimal control schemes with multiple controllers (players or decision makers) for Stochastic Differential Equations based on Dynamic Programming (DP). The papers associated with this chapter are Paper A and Paper B. Stochastic optimal control problems involve systems with continuous state and control action spaces wherein the evolution of the state is governed by a Stochastic Differential Equation (SDE) that evolves on a continuous timescale. The motivation behind studying such systems is that this framework is applied for studying optimal control problems subject to uncertainty such as stochastic reach-avoid problems, viability [1, 5] and numerous problems in economics, evolutionary biology, control systems especially robotics, and finance [2, 8]. If the cost function and the dynamics of each player are affected not only by the control actions taken by the considered controller but also by the cost actions taken by the other controllers, than the problem becomes a stochastic differential game and that is the focus of this chapter. Furthermore, we consider the controllers to be independent and non-communicating i.e. non-cooperative setting. Cooperative setting is studied in the book [17]. In Paper A, we propose a model-free variant of Markov chain Approximation using Monte Carlo simulation for estimating the value function. The estimated value function from the Monte Carlo simulation is compared to the value function obtained from the Markov chain Approximation. In Paper B, we have applied Markov Chain Approximation on the Stochastic Differential Equation to obtain a sequence of state-dependent Linear programs that we use to calculate a Minimax solution for the game. The reasoning behind*

choosing Minimax as the solution concept is to obtain a decentralized control scheme such that players need not communicate with each other. We formulate the problem precisely in the following section followed by a solution scheme using finite difference approximation of Hamilton-Jacobi equations.

1 Problem formulation

We consider a stochastic process $x(t)$ evolving on a continuous time scale $t \in \mathbb{R}_+$. Let $\mathfrak{p} = (\Omega, (\mathcal{F}_t)_{t\geq 0}, \mathbb{P})$ represent a filtered probability space, where Ω represent the sample space, $\mathcal{F}_t$ represent the filtration or the event space up to time t and $\mathbb{P}$ represent the corresponding probability measure. Consider the state variable $x(t)$ and control variable $u(t)$. The control variable $u(t) \in U \subset \mathbb{R}^m$ with m representing the number of players (or controllers). Specifically, each player controls a coordinate of the control vector $u(t)$. The state variable $x(t) \in \bar{\mathcal{X}} \subset \mathbb{R}^n$, where $\bar{\mathcal{X}}$ is the closure of the open subset $\mathcal{X} \subset \mathbb{R}^n$. The evolution of $x(t) \in \bar{\mathcal{X}}$ is governed by the following SDE,

$$dx(t) = f\left(x(t), u(t)\right) dt + \sigma\left(x(t), u(t)\right) dw(t), \tag{3.1}$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a bounded measurable function that represents the drift of the system, $\sigma : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n \times \mathbb{R}^p$ is a bounded measurable function that represents the diffusion with $a(\cdot) = \sigma(\cdot)\sigma(\cdot)^T$ being the corresponding diffusion matrix, w represents the standard Wiener process on $\mathbb{R}^p$. The solution to equation (3.1) satisfy

$$x(t) = x(0) + \int_0^t f\left(x(s), u(s)\right) ds + \int_0^t \sigma\left(x(s), u(s)\right) dw(s) \tag{3.2}$$

where local mean drift and local covariance are given by

$$\begin{aligned} \mathbb{E}[x(t+\Delta t) - x(t) \mid x(t), u(t)] &\approx f(x(t), u(t))\Delta t, \\ \mathrm{Cov}[x(t+\Delta t) - x(t) \mid x(t), u(t)] &\approx a(x(t), u(t))\Delta t. \end{aligned}$$

An SDE is simulated in fig. 3.1. The existence and uniqueness of solutions for an SDE is an important question for studying such a stochastic process as the wiener process $w(t)$ is not differentiable. However, for simplicity, we won't go into details to prove the same, and interested readers can go through the texts [8, 15] where a gentle introduction is given for studying SDE with the wiener process as a driving noise. In this work, we assume the existence and uniqueness in a strong sense as per the following assumption.

Assumption 1.1 [11, 12] We require the existence and uniqueness of solutions of (3.2). Strong existence holds if for a given probability space $\mathfrak{p}$, a filtration $\mathcal{F}_t$, an $\mathcal{F}_t$-Wiener process w and an $\mathcal{F}_0$- measurable initial condition $x(0)$, there exists an

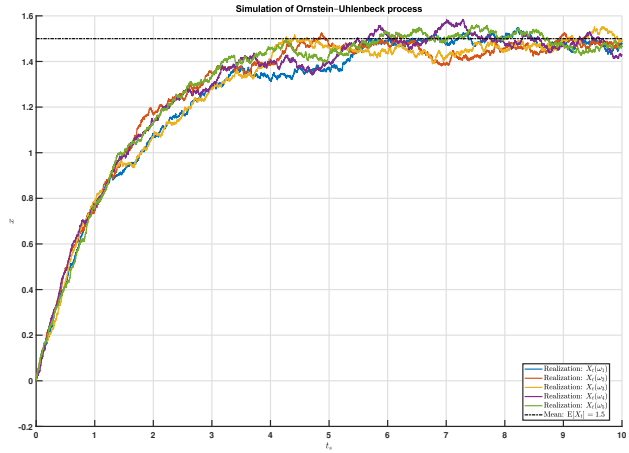


Fig. 3.1: Simulation of five sample paths of Ornstein-Uhlenbeck process with the mean value of 1.5. The stochastic process has been simulated using the Euler-Maruyama scheme.

$\mathcal{F}_t$-adapted process $x(t)$ satisfying (3.1) for all $t \geq 0$. Furthermore, uniqueness holds if for any two sample paths $x_1(t), x_2(t)$, $\mathbb{P}\{x_1(0) = x_2(0)\} = 1 \implies \mathbb{P}\{x_1(t) = x_2(t) \forall t \geq 0\} = 1$.

We also assume that the diagonal terms of the diffusion matrix are dominant over the non-diagonal terms. This assumption will be used later to ensure the well-posedness of approximate Hamilton-Jacobi equations while applying a numerical scheme with finite differences to the Ito operator.

Assumption 1.2 [11, 12] The diagonal terms of the diffusion matrix are dominant over off-diagonal terms as follows

$$a_{ii}(x) - \sum_{j:j\neq i} |a_{ij}(x)| \geq 0.$$

We define $u^{-k}(t) := [u^1(t), \cdots, u^{k-1}(t), u^{k+1}(t), \cdots u^m(t)]$ as the control vector $u(t)$ without the k^{th} players component or the control vector can be partitioned as $u(t) = [u^k(t), u^{-k}(t)]$.

1.1 Optimal Control problem formulation

Given an initial state $x(0) = x$, we define τ as the first hitting time of the solution $\phi(t)$ for (3.1) to the boundary $\bar{\mathcal{X}}$ as follows,

$$\tau(\phi) := \left\{ \inf_t \mid \phi(t) \notin \mathcal{X} \right\}.$$

Let $c^k : \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}$ represent the cost incurred by the k^{th} player, where $k \in \{1, \cdots, m\}$. If $x(t) \in \mathcal{X}$ for all time t, than $\tau = \infty$. This can happen in case, the asymptotic equilibrium point of (3.1) is within $\mathcal{X}$. Therefore, we need to consider discounted costs (with discount factor $\gamma \in (0, 1)$) as the cost might be infinite if $\tau = \infty$. Let $u^k = [u^k(1), \cdots u^k(\tau - 1)]$ be the sequence of control inputs by the k^{th} player and u^{-k} be the corresponding control sequence from all players except the k^{th} player. The optimal control problem from a given player k's perspective is as follows,

$$\inf_{u^k} \mathop{\mathbb{E}}_{x,u(\cdot)} \left(\int_0^\tau \gamma^t c^k \left(x(t), u^k(t), u^{-k}(t) \right) dt + \gamma^\tau c^k_\tau \left(x(\tau) \right) \right) \tag{3.3a}$$

$$\text{s.t. } dx(t) = f \left(x(t), u^k(t), u^{-k}(t) \right) dt + \sigma \left(x(t), u(t) \right) dw(t). \tag{3.3b}$$

The value function for player k is defined as follows,

$$V^k(x) = \inf_{u^k} \underbrace{\mathop{\mathbb{E}}_{x,u(\cdot)} \left(\int_0^\tau \gamma^t c^k \left(x(t), u^k(t), u^{-k}(t) \right) dt + \gamma^\tau c^k_\tau \left(x(\tau) \right) \right)}_{Q^k(x, u^k, u^{-k})}, \tag{3.4}$$

We now require the following assumptions on the drift function f, diffusion σ, and the cost function c^k for any player $k \in \{1, \cdots, m\}$.

Assumption 1.3 [11, 12] The functions f, σ and c^k are bounded, continuous, and Lipschitz continuous in x and uniformly in u. Furthermore, the drift function f and the cost function c^k are additively separable in terms of components due to individual players i.e. we only consider the so-called AR-AT (Additive Reward - Additive transition) games. Specifically, $f \left(x(t), u^k(t), u^{-k}(t) \right) = f^k \left(x(t), u^k(t) \right) + f^{-k} \left(x(t), u^{-k}(t) \right)$ and $c^k \left(x(t), u^k(t), u^{-k}(t) \right) = c^k_k \left(x(t), u^k(t) \right) + c^k_{-k} \left(x(t), u^{-k}(t) \right)$, where the subscript denotes the contribution due to the k^{th} player or due to every other player except the k^{th} player. Furthermore, for each initial state and control, the function $\tau(\cdot)$ is a continuous map from $\mathbb{R}^n \mapsto \mathbb{R}_+$ with probability 1 with respect to the measure induced by (3.2).

Lastly, we assume that each player can observe the history of play i.e. they can observe the following vector,

$$h(t) := [x(0), u^k(1), u^{-k}(1), \cdots, u^k(t-1), u^{-k}(t-1), x(t)],$$

with H being the set of all possible histories. The equilibrium condition for the differential game is based on the one presented in [7] wherein given a state $x(t)$, k^{th} players strategy π^k is a mapping from $H \mapsto U^k$, where $U^k \subset U^m$ and is stated as follows,

$$Q^k(x, \pi^{k\star}, \pi^{-k\star}) \leq Q^k(x, \tilde{\pi}^k, \pi^{-k\star}) + \varepsilon, \tag{3.5}$$

where $\tilde{\pi}^k$ is a unilateral deviation by the k^{th} player and ε is the tolerance error. In the sequel, the val$[\cdot]$ operator will represent the value of the game in the sense of (3.5).

2 Solution using finite-differences scheme on coupled Hamilton-Jacobi-Bellman-Issacs equations

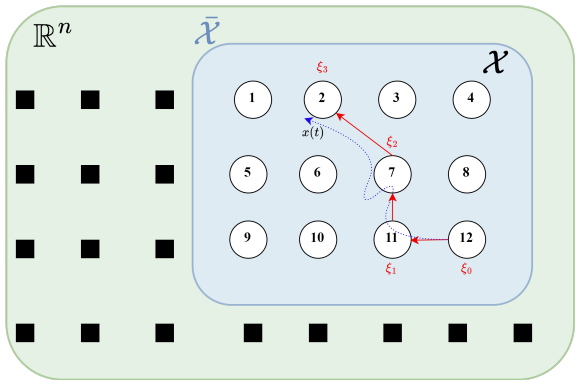


Fig. 3.2: Markov chain Approximation of (3.3) with circles and squares representing discrete states in the continuous state space $\mathbb{R}^n$. Note that the process is stopped once $x(t)$ hits the boundary $\bar{\mathcal{X}}$ of the set $\mathcal{X}$. The evolution of continuous dynamics (3.1) shown by the blue dotted line. The evolution of corresponding discrete dynamics over discrete states (represented by numbered circles) is shown by the red line.

Let V_x^k be the derivative of $V^k(x)$ with respect to x, and $V_{x_i x_j}^k$ represent the partial derivative of $V^k(x)$ with respect to x_i and x_j. The optimal value function $V^k(x)$ for any player $k \in \{1, \cdots, m\}$ must satisfy the following Hamilton-Jacobi-Bellman-Issacs

(HJBI) equations,

$$V^k(x)\ln\gamma + \mathrm{val}\Big[c^k\left(x(t),u^k(t),u^{-k}(t)\right) + V_x^k(x)\cdot f\left(x(t),u^k(t),u^{-k}(t)\right) + \frac{1}{2}\sum_{i,j=1}^{n} a_{ij}V^k_{x_ix_j}(x)\Big] = 0, \quad \text{for } x\in\mathcal{X} \tag{3.6}$$

with the boundary condition,

$$V^k(x) = c^k\left(x(\tau)\right), \quad \text{for } x\in\bar{\mathcal{X}}. \tag{3.7}$$

The derivation of the equations (3.6), (3.7) can be followed from the book [3] and the Paper [16] which gives a concise derivation from the Kolmogorov backward equations. In papers A and B, we have described the Markov chain Approximation scheme which is essentially a sequential finite difference approximation of (3.6) and (3.7). A visual illustration of the discretization is given in fig. 3.2 and in fig. 3.3.

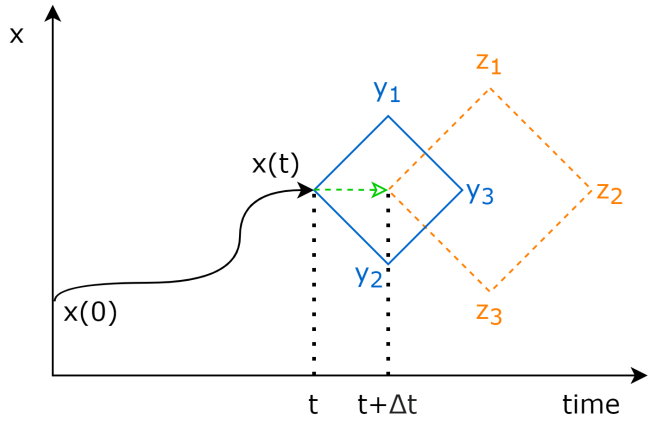


Fig. 3.3: Here we illustrate sequential approximation by MCA for a sample path of (3.1) starting at $x(0)$ and reaching $x(t)$ at time t, y_1, y_2 and y_3 represent the reachable states for $x(t)$ in time interval Δt. Any realized state $x(t+\Delta t)$ (shown by the green arrow) can be found as a convex combination of extremities of blue polygon. Once the next state $x(t+\Delta t)$ is reached the process repeats as shown by orange polygon with reachable states z_1, z_2 and z_3 (Source: [11, 12]).

Given a discretization parameter $h>0$ and coordinate basis $e_1, e_2, \cdots e_n$ of $\mathbb{R}^n$, we

can approximate the partial derivative $V^k_{x_i x_j}$ and gradient V^k_x as follows,

$$\frac{1}{2}\sum_{i,j=1}^{n} a_{ij}(x)\underbrace{\frac{V^k(x+e_ih)+V^k(x-e_ih)-2V^k(x)}{h^2}}_{\approx V^k_{x_ix_j}} + \sum_{i=1}^{n}\left|f(x,u^k,u^{-k})\right|^{+}\underbrace{\frac{V^k(x+e_ih)-V^k(x)}{h}}_{\approx \frac{1}{2}V^k_x} - \sum_{i=1}^{n}\left|f(x,u^k,u^{-k})\right|^{-}\underbrace{\frac{V^k(x)-V^k(x-e_ih)}{h}}_{\approx \frac{1}{2}V^k_x} + \gamma^t c^k(x,u^k,u^{-k}) = 0, \quad (3.8)$$

where $\left|f(x,u^k,u^{-k})\right|^{+} = \max(f(x,u^k,u^{-k}),0)$, approximates the positive part of the drift and $\left|f(x,u^k,u^{-k})\right|^{-} = \max(-f(x,u^k,u^{-k}),0)$ approximates the negative part of the drift. The reason behind the separation of drift according to the sign is that the associated approximate gradient of the Value function should be pointing in the direction corresponding to the drift which can change as the trajectory evolves on the state space [10]. Note that a similar consideration is not required for the partial derivative of the value function as it is associated with the diffusion matrix that is positive semidefinite. For simplicity and a clear presentation let us consider a discrete state space consisting of only 3 discrete states, $\xi = \{\xi_1, \xi_2, \xi_3\}$. Rearranging the terms of (3.8) leads to,

$$V^k(x) = \underbrace{\frac{a_{ii}(x)/2 + h\left|f(x,u^k,u^{-k})\right|^{+}}{\sum_{i=1}^{n}(a_{ii}(x)+h\left|f(x,u^k,u^{-k})\right|)}}_{p(x+e_ih|x,u^k,u^{-k})} V^k(x+e_ih) + \underbrace{\frac{a_{ii}(x)/2 + h\left|f(x,u^k,u^{-k})\right|^{-}}{\sum_{i=1}^{n}(a_{ii}(x)+h\left|f(x,u^k,u^{-k})\right|)}}_{p(x-e_ih|x,u^k,u^{-k})} V^k(x-e_ih) + \underbrace{\frac{h^2}{\sum_{i=1}^{n}(a_{ii}(x)+h\left|f(x,u^k,u^{-k})\right|)}}_{\Delta t}\gamma^t c^k(x,u^k,u^{-k}), \quad (3.9)$$

where $\xi_1 = x$, $\xi_2 = x + e_ih$ and $\xi_3 = x - e_ih$ can be considered as the discrete states and Δt is the time difference between two consecutive approximated Markov Game with initial state ξ_1 (see fig. 3.3). The probability of staying at x is given by, $p(x \mid x, u^k, u^{-k}) = 1 - (p(x + e_ih \mid x, u^k, u^{-k}) + p(x - e_ih \mid x, u^k, u^{-k}))$. Note that, Markov

chain Approximation can be done for more discrete states as well as per the procedure outlined in [8]. The following theorem due to [7] ensures that an ε-equilibrium in the sense of (3.5) is obtained and the value function approximated by the Markov chain Approximation can be made arbitrarily close to the value function of the original system.

Theorem 2.1 ([7]).
Given the assumptions 1.1, 1.2, and 1.3, consider the Markov chain approximation for (3.3) with discretization parameter h and state space ξ. For any $\varepsilon(h) > 0$ corresponding to the $\varepsilon(h)$-equilibrium for the approximated Markov chain, there exists an $\varepsilon > 0$ corresponding to the ε-equilibrium as per (3.5) with $\varepsilon \to 0$ as $h \to 0$.

2.1 Solving the associated Stochastic Game

The associated stochastic game corresponding to the Markov chain Approximation is given by the tuple $\mathcal{G} = (N, \xi, U_d, \mathcal{P}, [c^1], \cdots, [c^N])$, where N is the no. of players, ξ is the finite state space of the game, U_d is the finite control space of all the players obtained by discretizing the control space U, $\mathcal{P}$ is the probability transition matrix with elements obtained from (3.9), $[c^1], \cdots, [c^N]$ are the cost matrices for each player corresponding to U_d of dimensions $\nu \times \mid U_d^k \mid \times \mid U_d^{-k} \mid$. Since we will be approximating the problem (3.3) via (3.9), the instantaneous costs for the associated Stochastic game for the k^{th} player will be defined as follows,

$$c_t^k(\xi, u^i, u^{-i}) := \underbrace{\frac{h^2}{\sum_{i=1}^{n} (a_{ii}(x) + h\,|f(x, u^k, u^{-k})|)}}_{\Delta t} \gamma^t c^k(x, u^k, u^{-k}). \tag{3.10}$$

In order to ensure fully decentralized computation of control strategies, each player computes a minimax control strategy by solving a sequence of Linear programs parameterized by the state reached at any given time (see Paper B). Recall from the previous chapter that there does not exist a Linear program for solving zero-sum Markov games (2.35) with the source of nonlinearity being the multiplication of $\pi^k(x, u^k)$ with $V^k(x)$ in (2.35b). We propose to avoid this nonlinearity by solving a sequence of Linear programs inside a value iteration loop where $V^k(x)$ is replaced by its estimate obtained via Markov chain Approximation in Algorithm 4. Let ν represent the number of discrete states ξ and define the Shapley game matrix for each discrete state ξ attained during a sample path (see fig. 3.2) as follows,

$$M_t^k(\xi, u^k, u^{-k}) = c_t^k(\xi, u^k, u^{-k}) + \sum_{\xi'=1}^{v} p_t(\xi' | \xi, u^k, u^{-k}) \hat{V}_t^k(\xi'), \tag{3.11}$$

where $\hat{V}^k(\cdot)$ is the best estimate of the value vector at a given time using Markov chain Approximation. Note that for $\nu = 3$, the equation (3.11) takes almost the form of (3.9) with the differences being that (3.11) is defined for discrete controls instead of continuous controls in (3.9) and (3.9) does not consider the possibility of staying at the same state. The val[·] operator for a given state in update (3.13) of Algorithm 4 is evaluated by the following Linear program,

$$\min_{\pi^k, \hat{V}_t(\xi)} \quad \hat{V}_t^k(\xi) \tag{3.12a}$$

$$\text{s.t.} \quad \sum_{u^k \in U_d} M_t^k(\xi, u^k, u^{-k}) \pi^k(u^k) \leq \hat{V}_t^k(\xi), \quad \forall u^{-k} \in U_d, \tag{3.12b}$$

$$\sum_{u^k \in U_d} \pi^k(u^k) = 1, \tag{3.12c}$$

$$\pi^k(u^k) \geq 0, \quad \forall u^k \in U_d, \tag{3.12d}$$

Algorithm 4 MCA based Stochastic differential game solver [11]

1: **Input**: Initial state $x(0)$, Δt, h, f, a, U.
2: Initialize $\hat{V}^k(x) = 0$ for all states
3: **while** $t < \tau$ **do**
4: $\xi \leftarrow x(t)$
5: Find the set of possible discrete states corresponding to ξ
6: Construct cost matrix with elements $c_t^k(\xi, u^k, u^{-k})$, for all players
7: Obtain transition probabilities using (3.9)
8: **for** All possible control actions of players $-k$ **do**
9:

$$\hat{V}_{t+\Delta t}^k(\xi) \leftarrow \text{val}\Bigg[\underbrace{c_t^k(\xi, u^k, u^{-k}) + \sum_{\xi'=1}^{v} p_t(\xi' | \xi, u^k, u^{-k}) \hat{V}_t^k(\xi')}_{M_t^k}\Bigg] \tag{3.13}$$

10: **end for**
11: Solve the Shapley game (3.13) using (3.12) for $\pi^k(\xi, u^k)$
12: Sample u^k from $\pi^k(\xi, u^k)$ and apply on the system at time t
13: $t \leftarrow t + \Delta t$
14: **end while**

2.2 Simulation Results for MCA based solver

The following figures 3.4, 3.5 demonstrate the application of the proposed Algorithm 4 for solving Stochastic Differential games approximately. The reference output at both

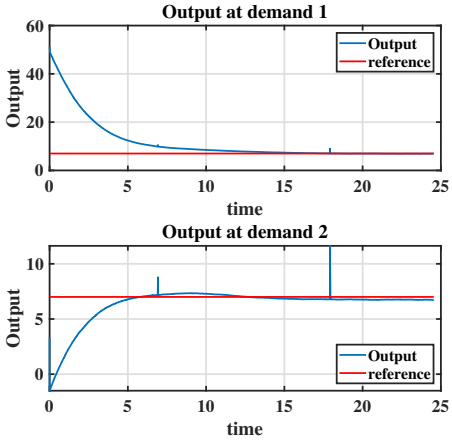


Fig. 3.4: Output pressure is tracked successfully to the reference (Source: [11]).

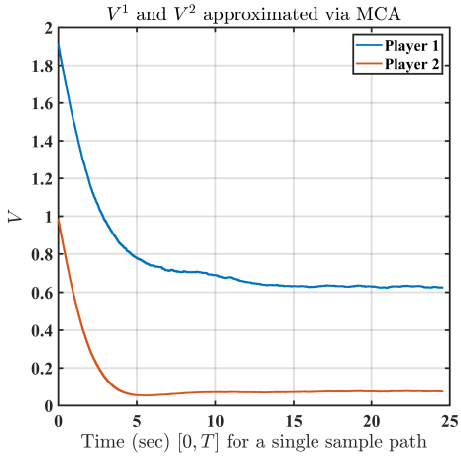


Fig. 3.5: The values obtained for both the players converge to the Minimax solution. The state space discretization was $h = 0.05$ (Source: [11]).

the consumers is tracked satisfactorily by both the controllers as shown in fig. 3.4 with spikes in pressure being due to the controllers switching off once reference is reached. As the considered water distribution network is asymmetric owing to the differences in the geographical heights and distances from the reference, it is relatively costlier for Player 1 to meet the objectives compared to Player 2 and this is visible in the approximated value functions in fig. 3.5. The following table 3.1 shows the variation in the value function approximations based on changes in the approximation parameter h.

No. of discrete states	V^1 obtained from MCA	V^2 obtained from MCA
$h = 0.2$	13.2525	1.7103
$h = 0.1$	7.8972	0.9951
$h = 0.067$	1.5339	0.1712
$h = 0.050$	0.6549	0.0732

Table 3.1: Value function approximations obtained at terminal time T for different h (Source: [11]).

3 Monte Carlo Approximation method

As an alternative to MCA, we proposed the Monte Carlo Approximation method. The motivation behind this was twofold: firstly, as the MCA method does not give any

method on how to choose a suitable approximation parameter h, we wished to verify how accurate the MCA method is in practice for the considered h. A related reason was that the transition matrix P constructed from the obtained transition probabilities p is sub-stochastic and there is a chance for the sample path to escape the considered grid space. Secondly, we wanted to obtain a procedure that constructs an approximate Markov chain from the sample paths of the simulated system instead of requiring knowledge of the drift term and the diffusion matrix (i.e. more amenable to learning-based control). This motivates the following algorithm which is the main contribution of Paper A.

Algorithm 5 Monte Carlo method for constructing P [12]

Input: State $\xi(s) = x(s)$, Sate space grid $\Delta^h x$, Maximum time T for a sample path, No. of sample paths to be evaluated M
for all sample paths $m \leq M$ **do**
  while $s < T$ **do**
    Find grid square $\Delta^h x(s)$ in $\Delta^h x$ which corresponds to $x(s)$
    Increment counter $MC(\Delta^h x(s))$ corresponding to $\Delta^h x(s)$ by 1
    Obtain realization of $x(s+1)$ using modified Euler-Maruyama method with interpolation time Δt
    $s = s + \Delta t$
  end while
  for all $i \leq n$, find $\xi_i(s+1)$ based on drift **do**
    if $b_i(x,u) \geq 0$ **then**
      $\xi_i(s+1) = \xi(s) + 1$
    else
      $\xi_i(s+1) = \xi(s) - 1$
    end if
  end for
  $p(\xi_i(s), \xi_i(s+1)) = \frac{MC_i(\Delta^h x(s+1))}{\sum_{i=1}^{n} MC_i(\Delta^h x)} \ \forall i < n$
  $p(\xi(s), \xi(s)) = 1 - \left(\sum_{i=1}^{n} \frac{MC_i(\Delta^h x(s+1))}{\sum_{i=1}^{n} MC_i(\Delta^h x)} \right)$
end for

3.1 Simulation Results for Monte Carlo Approximation method

We simulated 100 sample paths of the system using the Euler-Maruyama approximation on the model of the water distribution network in fig. 3.6 (alternatively, if we want to learn the dynamics of an unknown system, these sample paths can be obtained by sampling from it 100 times). Using these sample paths, we applied Algorithm 5 to approximate the value function. For comparison, we used the MCA scheme for

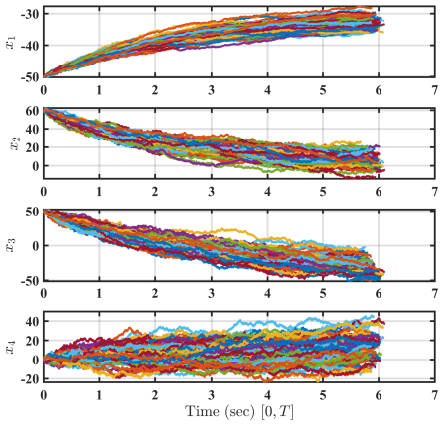


Fig. 3.6: Sample paths for initial $x = [-50\ 60\ 50\ 0]^T$ (Source: [12]).

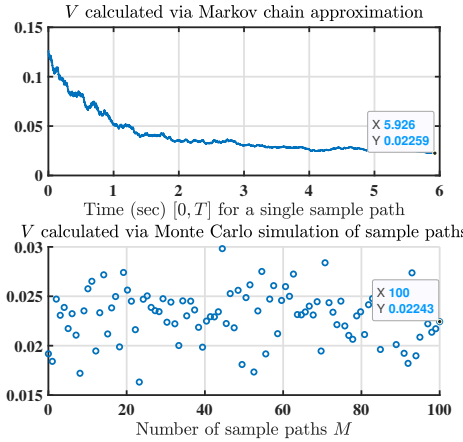


Fig. 3.7: The obtained value functions converge to almost same value (Source: [12]).

approximating the value function given the model and from fig. 3.7. In table 3.2, we compare the approximated value functions for 3 sample paths.

Initial condition	V obtained from MCA	V obtained from Algorithm 9
$x(0) = [-60\ 60\ 50\ 50]^T$	0.0170	0.0168
$x(0) = [-50\ 50\ 30\ 60]^T$	0.0157	0.0158
$x(0) = [-50\ 60\ 50\ 10]^T$	0.0194	0.0198

Table 3.2: Value functions obtained for different initial conditions (Source: [12]).

4 Solving multi-objective dynamic games using Approachability

So far, we have considered stochastic differential games having a single objective function only. The concept of Approachability (discussed in 2.10) can be used to extend the aforementioned Markov Chain Approximation method to games having multiple objectives. For simplicity, we shall consider dynamic games defined via difference equations. The reasoning behind this is that the concept of Approachability only requires the av-

erage of returns obtained via each sample path or trajectory (this average is denoted by ϕ in the sequel). Consider the dynamic game $\Gamma = (N, X, U, \mathbf{C}, f, w, T)$, where N is the number of players, $x \in X \subseteq \mathbb{R}^n$ represents the underlying system states, U is the Cartesian product space $U = U^1 \times \ldots \times U^i \ldots \times U^N$, where $U^i \subset \mathbb{R}^{m_i}$ is the control action space for a given player $i \in \{1, \ldots, N\}$. We assume U^i to be a convex and compact space for any given player i. Let $\mathbf{c}^i : X \times U \to \mathbb{R}^D$ represent the vector of costs to a given player i. The state evolves according to the following difference equation,

$$x_{t+1} = f(x_t, u_t^i, u_t^{-i}, w_t) \tag{3.14}$$

where $t = 0, 1, \ldots, T$ is the discrete time, $f : \mathbb{R}^n \times U \times \mathbb{R}^n \mapsto \mathbb{R}^n$ maps the given state x_t, control inputs u_t^i, u_t^{-i}, and state noise w_t to the next state x_{t+1}. In the sequel, we consider the stochastic control problem from player i's perspective. Let $\pi^i : X \to U^i$ denote the control policy for player i and $\pi : X \to U$ denote the joint control policy of all the players with $u = (u^i, u^{-i})$ being the corresponding joint control action as per π. We consider only stable joint policies π i.e. the solution of (3.14) under any joint policy π is stable or f is a contraction map for all inputs under π [4]. For a given initial state x, let $\mathbb{E}^x_{\pi,w}[\cdot]$ denote the conditional expectation operator dependent on the sequence of $u_1, \ldots, u_T$ and $w_1, \ldots, w_T$. Let $\mathbf{V}^i_{\pi^i}(x, \tau) := \sum_{s \leq \tau} \mathbf{c}^i(x_s, \pi^i(x_s), u_s^{-i})$ be the value of trajectory t (starting at initial state x and running till time τ) and we define $\phi^i_T(x) := (1/T) \sum_{t \leq T} \mathbf{V}^i_{\pi^i}(x, \tau)$ as the average value of starting at state x. Consider a closed convex set $\mathcal{A} \subset \mathbb{R}^D$ in the space of costs. The distance between a point a and the set $\mathcal{A}$ is defined as

$$dist_{\mathcal{A}}(a) = \inf_{b \in \mathcal{A}} \|a - b\| .$$

The goal of player i is to ensure that for a given initial state x, $\phi^i_T(x)$ can *Approach* the set $\mathcal{A}$ no matter what control actions are taken by other players. This is defined more precisely as follows.

Definition 4.1 (Approachability) A set $\mathcal{A} \subset \mathbb{R}^D$ is *Approachable* by Player i, if Player i can construct a policy π^i such that irrespective of π^{-i}, we have,

$$\lim_{T \to \infty} \mathbb{E}^x_{\pi,w}[dist_{\mathcal{A}}(\phi^i_T)] \to 0, \quad \text{almost surely.}$$

Prior to introducing Blackwell's Approachability Theorem, we shall introduce the definition of p-Enforceability.

Definition 4.2 (p-Enforceability) Consider a scalar cost $c^i(x_s, u_s^i, u_s^{-i})$, where $V^i_{\pi^i}(x, \tau) := \sum_{s \leq \tau} c^i(x_s, \pi^i(x_s), u_s^{-i})$ represents the scalar value of a player i's trajectory. This trajectory starts at state x, concludes at time τ, and player i follows

policy π^i. A scalar cost p is p-Enforceable by player i, if player i has a policy π^i such that

$$\sup_{u^{-i}\in U^{-i}} \mathbb{E}^x_{\pi,w}[V^i_{\pi^i}(x,\tau)] \leq p.$$

We define the support function $v_{\mathcal{A}} : \mathbb{R}^D \to \mathbb{R} \cup \{-\infty, \infty\}$ as,

$$\Lambda \mapsto v_{\mathcal{A}}(\Lambda) := \sup\{\Lambda \cdot y : y \in \mathcal{A}\}.$$

Theorem 4.1 (Blackwell's Theorem [14]).
The closed convex set $\mathcal{A} \subset \mathbb{R}^D$ is Approachable *by player i if and only if*

- *for every $\Lambda \in \mathbb{R}^D$ there exists a control policy π^i such that*

$$\sup_{u^{-i}\in U^{-i}} \mathbb{E}^x_{\pi,w}[\Lambda \cdot \mathbf{V}^i_{\pi^i}(x,\tau)] \leq v_{\mathcal{A}}(\Lambda), \tag{3.15}$$

  or player i has $v_{\mathcal{A}}(\Lambda)$-Enforceability for the projected scalar game.

- *At time $t+1$, player i plays the policy π^i_t (where π^i_t is the strategy corresponding to (3.15) for cost $\mathbf{V}^i_{\pi^i}(x,\tau)$) if $\phi^i_t(x) \notin \mathcal{A}$ and plays arbitrarily if $\phi^i_t(x) \in \mathcal{A}$.*

In other words, given an initial state x, if at all time t player i can find a control policy π^i_t such that the value of the projected game $\Lambda \cdot \mathbf{V}^i_{\pi^i}(x,\tau)$ falls on the other side of the halfspace containing $\mathcal{A}$, then the time-average of ϕ^i_t approaches $\mathcal{A}$ almost surely. Essentially, Blackwell's Approachability theorem allows us to reduce the problem of Approaching the set $\mathcal{A}$ to Approaching the halfspace containing $\mathcal{A}$ and we still need to construct policies that ensure $v_{\mathcal{A}}(\Lambda)$-Enforceability. Markov chain Approximation or the Monte Carlo method proposed in Algorithm 5 can be used to find such policies.

5 Conclusions

In this chapter, we have explored the Markov chain Approximation scheme as a means of solving optimal control problems with underlying dynamics described via stochastic differential equations. As shown in [13] a model-based reinforcement learning scheme can be obtained by sampling trajectories of the system and constructing an estimate of the value function based on the dynamics approximated from the sampled trajectories. We have verified the value function estimates by comparing them with a Monte Carlo method and we get similar results as shown in Paper A. Furthermore, we have

presented a sequential Minimax game solver based on Markov chain Approximation. We have relied on the convergence results presented by Kushner in [6, 7]. Empirically, the algorithm works well. However, as the consumer dynamics in the water distribution network are relatively slow compared to the fast dynamics in the piping networks, we feel that we can simplify our analysis by considering only analysis at the steady state (we do not consider tanks, for networks with tanks see [9]). This is the motivation behind the following chapter on repeated games.

As a future work, we would like to study the convergence of the algorithm presented in Paper B for arbitrary systems. The Linear programming relaxation in the algorithm for solving the corresponding stochastic game is of independent interest and its convergence should also be studied. Secondly, the Monte Carlo method presented in Paper A can be used as an alternative for systems with unknown dynamics and it would be interesting to combine the Monte Carlo Approximation method with the Markov chain Approximation based Stochastic Differential game solver in Paper B. Finally, by integrating the Monte Carlo method with Approachability, we can develop a wholly model-free algorithm capable of addressing multi-objective OCP or multi-objective DG. This is particularly intriguing because, to the best of our knowledge, there is currently a scarcity of literature on these subjects.

References

[1] P. M. Esfahani, D. Chatterjee, and J. Lygeros, "The stochastic reach-avoid problem and set characterization for diffusions," *Automatica*, vol. 70, pp. 43–56, 2016.

[2] W. H. Fleming and W. M. McEneaney, "Risk-sensitive control on an infinite time horizon," *SIAM Journal on Control and Optimization*, vol. 33, no. 6, pp. 1881–1915, 1995.

[3] W. H. Fleming and H. M. Soner, *Controlled Markov processes and viscosity solutions*. Springer Science & Business Media, 2006, vol. 25.

[4] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and J. D. Powell, *Feedback control of dynamic systems*. Prentice hall Upper Saddle River, 2002, vol. 4.

[5] P. E. Kloeden and E. Platen, "Stochastic differential equations," in *Numerical Solution of Stochastic Differential Equations*. Springer, 1992, pp. 103–160.

[6] H. J. Kushner, "Numerical approximations for stochastic differential games," *SIAM journal on control and optimization*, vol. 41, no. 2, pp. 457–486, 2002.

[7] ——, "Numerical approximations for nonzero-sum stochastic differential games," *SIAM journal on control and optimization*, vol. 46, no. 6, pp. 1942–1971, 2007.

[8] H. J. Kushner and P. G. Dupuis, *Numerical methods for stochastic control problems in continuous time*. Springer Science & Business Media, 2001, vol. 24.

[9] J. Ledesma, "Safe reinforcement learning control for water distribution networks," 2022, phD supervisor: Professor Rafał Wisniewski, Aalborg University Assistant PhD supervisor: Professor Carsten Skovmose Kallesøe, Grundfos Holding A/S / Aalborg University.

[10] R. J. LeVeque, *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.

[11] R. Misra, R. Wisniewski, and C. S. Kallesøe, "Approximating solution of stochastic differential games for distributed control of a water network," *IFAC-PapersOnLine*, vol. 55, no. 16, pp. 110–115, 2022.

[12] ——, "Approximating the model of a water distribution network as a markov decision process," *IFAC-PapersOnLine*, vol. 55, no. 20, pp. 271–276, 2022.

[13] R. Munos and P. Bourgine, "Reinforcement learning for continuous stochastic control problems," in *NIPS*, 1997, pp. 1029–1035.

[14] V. Perchet, "Approachability, regret and calibration: Implications and equivalences," *Journal of Dynamics and Games*, vol. 1, no. 2, pp. 181–254, 2014.

[15] S. Särkkä and A. Solin, *Applied stochastic differential equations*. Cambridge University Press, 2019, vol. 10.

[16] H. E. Wiltzer, D. Meger, and M. G. Bellemare, "Distributional hamilton-jacobi-bellman equations for continuous-time reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2022, pp. 23 832–23 856.

[17] D. W. Yeung and L. A. Petrosyan, *Cooperative stochastic differential games*. Springer, 2006, vol. 42.

Chapter 4

Repeated Games based Decentralized Control

Summary *This chapter summarizes the contributions of this dissertation on decentralized control based on repeated games. We consider complex systems (with the primary example being a water distribution network) and use the notion of noncooperative repeated games for modeling the interaction between the controllers (players or decision makers) and the system. The papers associated with this chapter are Paper D and Paper E. In contrast to the previous chapter, we do not explicitly consider the state of the system and instead we either assume them to be implicitly encoded in the costs incurred by the controllers per stage of the repeated game or we assume them to be composed of fast transient states and steady state with the assumption being that the system is stable and the transients decay quickly. The former assumption is useful for studying stochastic games and multi-agent reinforcement learning as the stochastic game solver typically consists of a static game solver inside some Bellman-like equation [1] while the latter assumption is especially true for water distribution networks without the inclusion of tanks. Here, the fast transient dynamics represent the pipe flows and slow dynamics represent the change in pressure in controlled pressure zones. Such networks are becoming increasingly common due to the risk of water contamination in storage tanks [2]. Practical motivation behind the design of a decentralized control algorithm for a water distribution network is twofold as follows: Firstly, most of the water networks are catering to increasing water demands and decentralization of control is required for seamless integration of new controlled units [8]. Secondly, water distribution networks are a critical infrastructure and decentralization helps in safeguarding against cyber attacks [15]. Paper D presents a model-based control scheme where the costs are obtained by solving steady-state equations of the water distribution network. The proposed decentralized control scheme was successfully implemented in the Smart water lab at Aalborg*

University.

Paper E considers repeated games perturbed by time-varying (but bounded) disturbances and considers the correlated equilibrium condition as the optimality criteria. It turns out that the correlated equilibrium is not robust to such time-varying disturbances as shown by a simple example in Paper E. We therefore propose Robust Correlated equilibrium as the optimality criteria and we propose a modified regret-matching algorithm for decentralized control with the modification being done so as to robustify the standard regret-matching scheme against the aforementioned disturbances. Convergence analysis of the modified regret-matching scheme is provided in Paper E.

1 Modeling of a water distribution network using graph theory

A water distribution network consists of three key elements pumping stations that supply water in the network, consumers that consume water fed by the pumping stations, and pipes that interconnect pumping stations to the consumers. A large-scale water distribution network is typically subdivided into smaller pressure zones or District Metered Areas (DMA). A practical water distribution network is best represented via a Process and Instrumentation diagram as shown in the following figures 4.1, 4.2. The

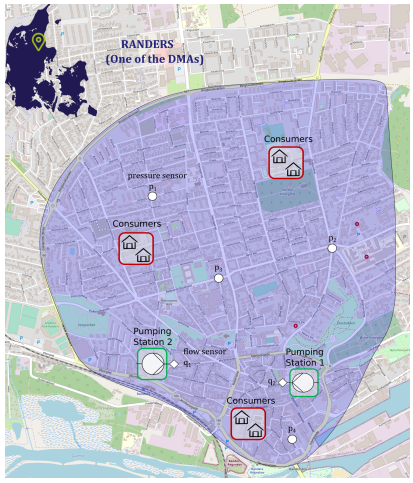


Fig. 4.1: The water distribution network of a typical medium-sized town in Denmark (Randers).

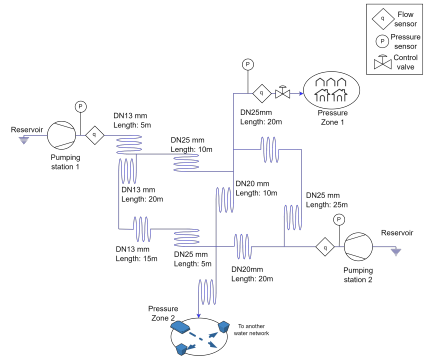


Fig. 4.2: Process and Instrumentation diagram corresponding to the Randers water distribution network in fig. 4.1 (Source: [10]).

individual components of a water distribution network such as a pumping station, con-

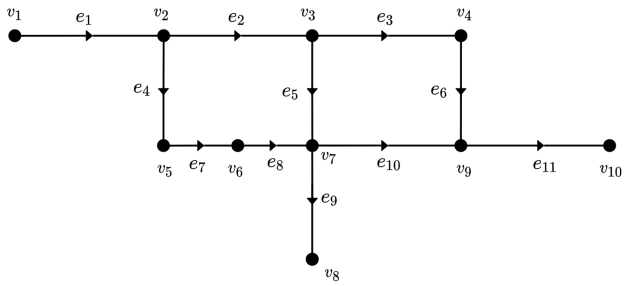


Fig. 4.3: Graph corresponding to Process and Instrumentation diagram in fig. 4.2 (Source: [10]).

sumers, and pipes are modeled via equations that represents laws of physics such as mass and energy balance [6]. The interconnecting pipe network is modeled via Graph theory which conveniently allows us to represent the network via matrices [3]. The Process and Instrumentation diagram in fig. 4.2 is equivalently represented via fig. 4.3, where the edges represent the physical components of the network (such as pumps, pipes, and consumers), and the nodes represent the points where pressure can be measured. The dynamics are due to the water flow in pipes. However, in this work, the goal is not to control the water flow in the network but instead to control the pressure at DMA's. The water flow dynamics are relatively *very fast* compared to the rate of change of pressure at the DMA's. Due to the relatively *slow* variation of the measured pressure at the DMA's, the water network can be modeled by static equations [5, 14]. The static equations represent the flow and pressure balance in the network. In the previous papers, Paper A and Paper B, we considered dynamic equations [11, 12]. However, as the flow dynamics are relatively fast, the system settles to a steady state very quickly, and thus, the dynamic equations do not significantly improve the accuracy of the estimated pressure obtained via the static equations model. For background on the derivation of the static equations based model of the water distribution network with the inclusion of measured pressure readings, please refer to Paper D.

2 Model based Control

If the model of the water distribution network is known (along with the possible control actions that each controller can take) then each controller can build a cost matrix (for a matrix game) with entries corresponding to all possible control actions. The costs obtained on applying a control action to the system serve as the feedback signal for the controller. Suppose the controllers are not allowed to communicate with each other (as they are part of a decentralized system). In that case, a simple decentralized control

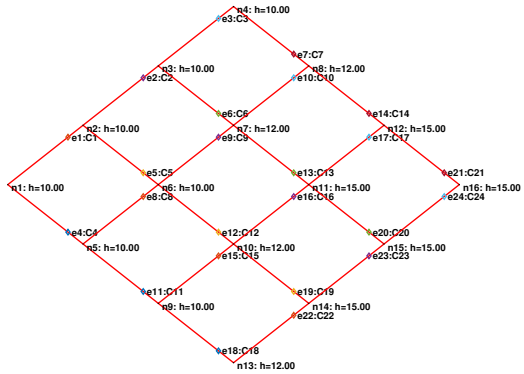


Fig. 4.4: A Complex piping Network (Source: [7]).

scheme can be obtained if each controller assumes to optimize against the worst-case strategy of the opponents. In this case, each controller updates its mixed strategy by independently solving the optimization problem (2.30). This forms the basis of the control strategy in Paper D. However, while this approach is easy to implement, it can be an overly "pessimistic" approach since each controller also has a common goal of maintaining the reference pressure alongside the "selfish" goal of minimizing their own energy costs. Thus, we have a nonzero-sum game. As nonzero-sum games have multiple Nash equilibrium points, we need to consider some sort of coordinating mechanism such that the controllers strive to reach a common equilibrium point. One way to do this is by solving the optimization problem (2.33) in the case of 2 controllers. However, the optimization problem (2.33) needs to be solved in a centralized setting as the optimizer should know the cost functions of both players and furthermore, this approach cannot be extended to more than 2 players unless we group players together in some coalitions and then search for an equilibrium solution between the coalitions. The second approach is designing a correlating signal that facilitates coordination between the two controllers. This is done in Paper E where the partial history observed by each of the controllers serves as the correlating signal and the methods discussed in Paper E can be implemented in a model-free fashion. We postpone the discussion about this method for the next subsection. Prior to implementing the decentralized control strategy based on (2.30) in the smart water lab, we performed numerical studies on a

complex water network (from [7]) to judge the degradation in overall performance on account of using minimax strategies instead of finding a Nash equilibrium. We consider the piping network shown in the fig. 4.4 and connect the pumping stations at nodes 1 and 16 and we seek to maintain pressure on nodes 9 and 14 (these represent pressure zones or DMA's). The figures 4.6 and 4.5 show the pressure at regulated nodes 9 and 14 with the corresponding control actions by the controllers (referred to as players). Each controller independently obtained the control actions by solving (2.30). The figures 4.8 and 4.7 show the pressure at regulated nodes 9 and 14 with the corresponding control actions by the controllers (referred to as players). The control actions of both players were obtained by centrally solving(2.33). From the numerical simulations it is evident that the model-based decentralized control using minimax strategies does reasonably well in comparison to the centralized Nash equilibrium controller. This motivated us to implement decentralized control based on minimax strategies in the smart water lab and the results were positive and summarized in Paper D.

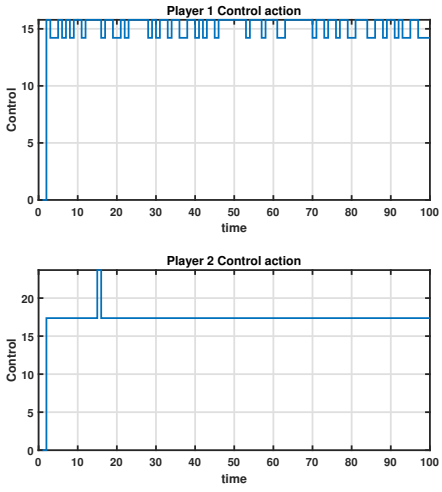


Fig. 4.5: Decentralized Control input.

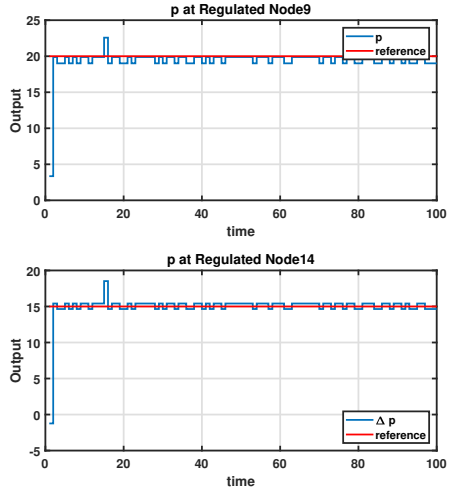


Fig. 4.6: Pressure at the regulated nodes under Decentralized Control scheme.

2.1 Model-based algorithm for decentralized control

This subsection focuses on the practical implementation of an algorithm for online decentralized control of the water distribution network with real-time measurements. The designed algorithm is a supervisory control that provides pressure set points to low-level pump controllers (standard Proportional-Integral controls) that seek to control

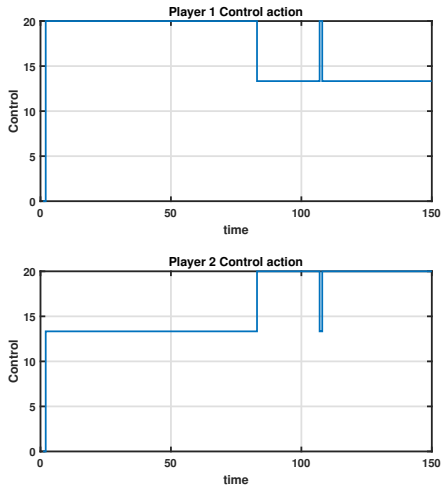


Fig. 4.7: Centralized Control input.

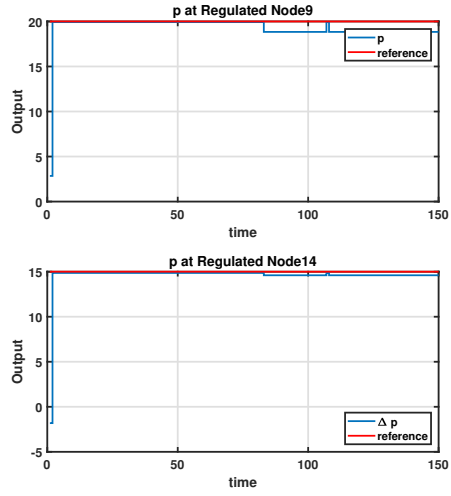


Fig. 4.8: Pressure at the regulated nodes under Centralized Control scheme (Idea setting).

smaller pumping units with the observed costs serving as the feedback signal to the controller (see fig. 4.9 for a visual representation of the control scheme). As the rate

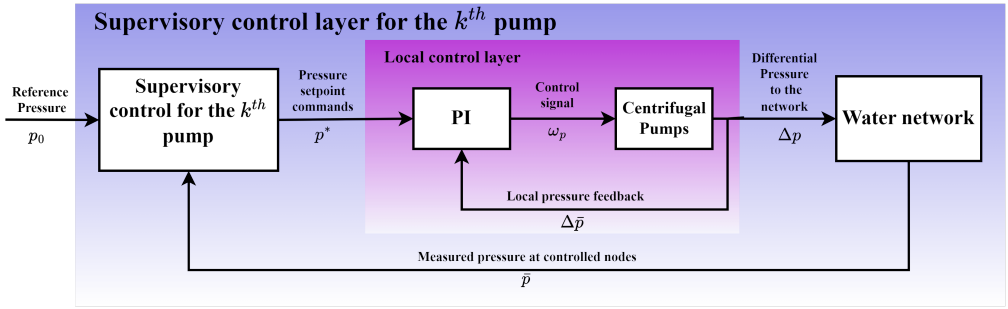


Fig. 4.9: Implemntation of the proposed control algorithms as a supervisory control layer on top of local Proportional-Integral controls.

of change of pressure is relatively slow compared to the running time of the algorithm, the algorithm is executed after certain time intervals only (so as to allow for settling of the pressure). The time at which the algorithm updates the pressure set point is referred to as a decision epoch in the sequel. The control input is constant between any two decision epochs. Since the consumption demand might change between the

epochs, we consider an average of all the consumption demands between the decision epochs in the Algorithm 6. Given the reference pressure p_0 that we would like to track, real-time pressure measurements $\bar{p}$ from the pressure sensors and the real-time consumption demands d_c and the model of water distribution network as discussed in Paper D, the following algorithm can be implemented in a decentralized setting.

Algorithm 6 Online Model-based decentralized control [10]

Input: $\bar{p}$, p_0, d_c
Calculate $\bar{d}_c$ by averaging d_c since last decision epoch
for $t = 1, \cdots, T$ **do**
    for All possible control actions of all players **do**
        Construct C_t^k using the model equations in Paper D
    end for
    Solve the game Γ' using (2.30) for π_t^k
    Sample $u_t^k \sim \pi_t^k$
    Apply u_t^k as control input
    $t \leftarrow t + 1$
end for

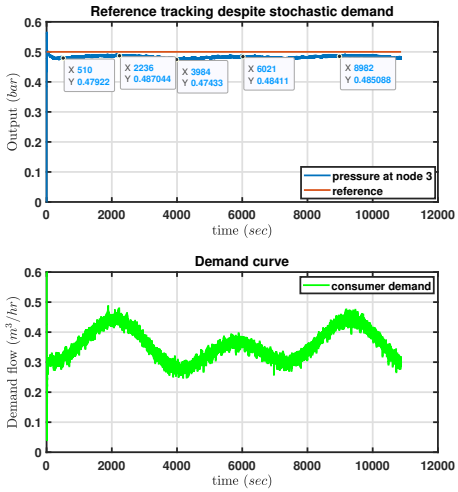


Fig. 4.10: The reference is tracked up to a certain error (maximum error being approximately 0.0256 *bar*) despite changing consumer demands (Source: [10]).

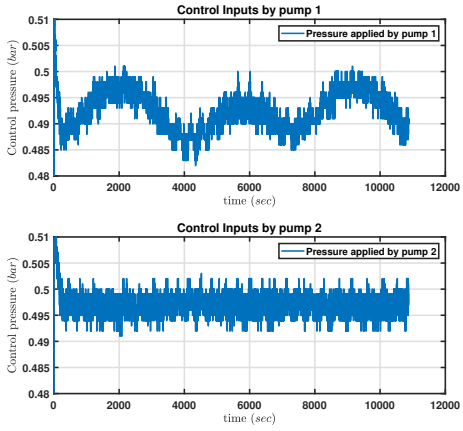


Fig. 4.11: The pressure control signal applied by both the pumping stations (Source: [10]).

Algorithm 6 was implemented in the Smart water lab with the considered network

corresponding to the real-life network of Randers, Denmark as shown in the fig. 4.1. The real-time data collected from the sensors is plotted in the following figures 4.10, 4.11. In fig. 4.11, we can observe that controller 1 applies more pressure when the consumer demand is higher, leading to a higher pressure drop while controller 2 has an almost constant control input on average.

3 Model free Control

It has been long desired in the control community to have methods that allow simple "plug and play" of new controllers in existing controlled systems [16]. To that end, we propose to solve the problem of decentralized control for water distribution networks in a repeated game framework where the controllers can only observe the costs incurred by them and keep a history of their past actions. The partial history observed by each controller acts as the correlating signal and we propose to use regret matching as the decentralized control algorithm. If each controller uses a regret matching algorithm then the joint distribution converges to the correlated equilibrium of the game [4]. However, given the perturbations in costs on account of stochastic consumption (besides due to the control actions taken by other controllers), we cannot guarantee that the correlated equilibrium obtained via the regret matching algorithm will be robust to the aforementioned disturbances. This is demonstrated by the example 2.3. We propose to robustify the standard regret matching algorithm against these disturbances by adding a "momentum" term that can basically be thought of as the exponential moving average of past gradients. This helps to smooth out fluctuations due to the perturbations in the costs on account of consumption disturbances (provided that the consumption disturbances are bounded). This modification of regrets (referred to as CR or conditional regrets) is represented by the following equation,

$$\widehat{CR}^i_{t_+}(a,b) = \left(1 - \frac{1}{t}\right)\widehat{CR}^i_{t-1_+}(a,b) + \frac{1}{t}(c^i_{d_t}(a, u_t^{-i}) - c^i_{d_t}(b, u_t^{-i})) + \frac{1}{t}(\widehat{CR}^i_{t-1_+}(a,b) - \widehat{CR}^i_{t-2_+}(a,b)), \quad (4.1)$$

where $a, b \in U^i$ are the control actions of player i that are compared against each other, $c^i_{d_t}$ is the cost (incurred by player i) perturbed due to disturbances d_t. The cost is dependent on actions of player i and the actions taken by the other players (denoted by u^{-i}). The term $\widehat{CR}^i_{t-1_+}(a,b) - \widehat{CR}^i_{t-2_+}(a,b)$ is the aforementioned "momentum" term. The algorithm for decentralized control of the water distribution network based on Robust Correlated Equilibrium is stated next as Algorithm 7. In this algorithm, the player i updates the initial policy x^i at each time step as per the steps in the Algorithm. If all the players update their control policies as per Algorithm 7, then the empirical joint distribution of play is guaranteed to converge to the Robust Correlated Equilibrium of

the considered game (see Paper E of all the assertions here). In Paper E, we provide convergence analysis of Algorithm 7 and the main result is the following proposition.

Algorithm 7 Perturbed Conditional Regret Matching [13]

Input: Control space U^i, initial policy x_1^i is uniform distribution over U^i.
Obtain $u_t^i \sim x_t^i$ and get corresponding $c_{d_t}^i(u_t^i, u_t^{-i})$
Let $u_t^i = a$
Set $\alpha_t = \frac{1}{t}$
for $u^i = 1, \cdots, b, \cdots, \mid U^i \mid$ **do**
    Define the vector $q_\pi \in \Delta(U^i)$ as

$$q_\pi(b) = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{otherwise .} \end{cases}$$

    Update estimated perturbed regrets $\widehat{CR}_{t_+}^i$

$$\widehat{CR}_{t_+}^i(a,b) = (1-\alpha_t)\widehat{CR}_{t-1_+}^i(a,b) + \alpha_t(c_{d_t}^i(a, u_t^{-i}) - c_{d_t}^i(b, u_t^{-i})) + \alpha_t(\widehat{CR}_{t-1_+}^i(a,b) - \widehat{CR}_{t-2_+}^i(a,b))$$

end for
Define normalizing constant $\mu = \sum_{b \neq a} \widehat{CR}_{t_+}^i(a,b)$
for $u^i = 1, \cdots, b, \cdots, \mid U^i \mid$ **do**
    Update entries of the transition matrix $\pi_t^i(a,b)$

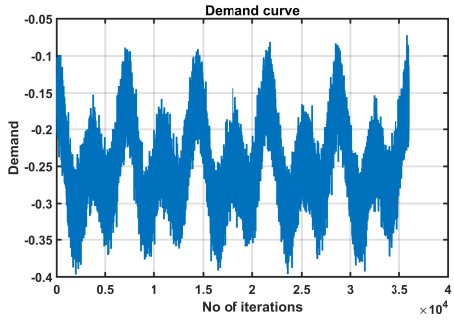
$$\pi_t^i(a,b) = \begin{cases} \frac{1}{\mu}\widehat{CR}_{t_+}^i(a,b), & \text{if } a \neq b, \\ 1 - \sum_{b' \neq a} \frac{1}{\mu}\widehat{CR}_{t_+}^i(a,b'), & \text{if } a = b, \\ \frac{1}{|U^i|}, & \text{if } \mu = 0 \end{cases}$$

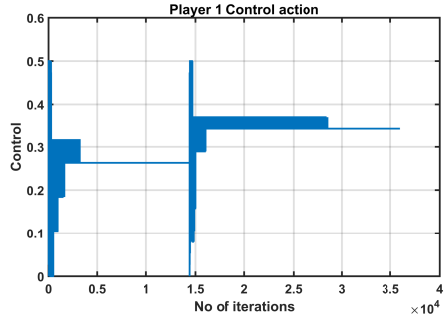
    Update the control policy $x_{t+1}^i(b) = q_\pi(a)\pi_t^i(a,b)$
end for

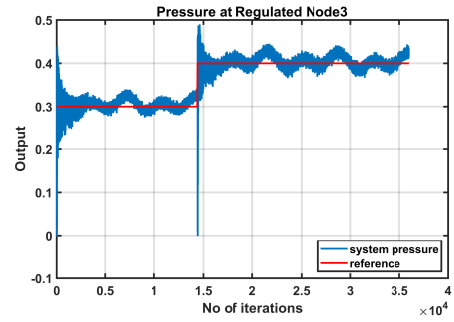
Proposition 3.1 ([13]).
The estimated perturbed conditional regret $\widehat{CR}^i$ and the corresponding perturbed conditional regret CR^i in Algorithm 16 converges to 0 for all actions $a, b \in U^i$ and for all players $i \in N$ if all the players use Algorithm 7 in a decentralized manner.

Algorithm 7 was applied on the same network as shown in fig. 4.1 and the following

results were obtained.







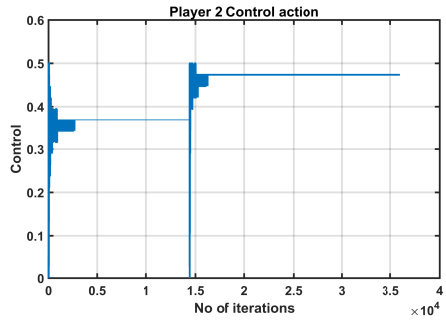


Fig. 4.12: The reference pressure is tracked satisfactorily by both players despite consumption disturbances (Source: [13]).

Fig. 4.13: Control actions of both players corresponding to fig. 4.12 (Source: [13]).

4 Conclusions

In this chapter, we explored the application of repeated games as a framework for the decentralized control of large-scale water distribution networks. Given, the slow dynamics of the water distribution network repeated games framework is useful for achieving decentralized control. This was verified both in simulation and empirically in the smart water lab as well. Furthermore, the theoretical results from Paper E prove that the empirical joint distribution of play converges to the set of robust correlated equilibrium if all the controllers apply regret matching on estimated perturbed conditional regrets.

Future work involves extending the results presented here to Markov games. Although the disturbances can also be thought of as different states of the system, the primary goal behind the design of the control policy is to be indifferent to changing disturbances. Instead of this approach, future research should focus on developing control policies that depend on the state. The ideas from [9] can be used as inspiration towards this research direction. Another interesting research direction is to define the concept of Correlated equilibrium for Differential Games or DG's. As discussed in the previous chapters, finding feedback Nash equilibrium for DG's is challenging but Nash equilibrium is challenging even for static games. Analogous to feedback Nash equilibrium and feedback Stackelberg Equilibrium, one should be able to define the concept of feedback Correlated Equilibrium and we conjecture that a variant of Hamilton-Jacobi optimality equations should exist for feedback Correlated equilibrium and they might be easier to solve relative to coupled Hamilton-Jacobi equations arising for feedback Nash equilibrium.

References

[1] M. Bowling and M. Veloso, "Multiagent learning using a variable learning rate," *Artificial Intelligence*, vol. 136, no. 2, pp. 215–250, 2002.

[2] D. Chalchisa, M. Megersa, and A. Beyene, "Assessment of the quality of drinking water in storage tanks and its implication on the safety of urban water supply in developing countries," *Environmental Systems Research*, vol. 6, no. 1, pp. 1–6, 2018.

[3] N. Deo, *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2017.

[4] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[5] T. N. Jensen, "Plug & play control of hydraulic networks," Ph.D. dissertation, Aalborg University, 2012.

[6] T. N. Jensen, C. S. Kallesøe, J. D. Bendtse, and R. Wisniewsk, "Plug-and-play commissionable models for water networks with multiple inlets," in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 1–6.

[7] T. N. Jensen, C. S. Kallesøe, J. D. Bendtsen, and R. Wisniewski, "Iterative learning pressure control in water distribution networks," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018, pp. 583–588.

[8] P. P. Kalbar and S. Lokhande, "Need to adopt scaled decentralized systems in the water infrastructure to achieve sustainability and build resilience," *Water Policy*, vol. 25, no. 4, pp. 359–378, 2023.

[9] S. Mannor and N. Shimkin, "The empirical bayes envelope and regret minimization in competitive markov decision processes," *Mathematics of Operations Research*, vol. 28, no. 2, pp. 327–345, 2003.

[10] R. Misra, C. S. Kallesøe, and R. Wisniewski, "Decentralized control of a water distribution network using repeated games," in *2023 27th International Conference on Methods and Models in Automation and Robotics (MMAR)*. IEEE, 2023, pp. 181–186.

[11] R. Misra, R. Wisniewski, and C. S. Kallesøe, "Approximating solution of stochastic differential games for distributed control of a water network," *IFAC-PapersOnLine*, vol. 55, no. 16, pp. 110–115, 2022.

[12] ——, "Approximating the model of a water distribution network as a markov decision process," *IFAC-PapersOnLine*, vol. 55, no. 20, pp. 271–276, 2022.

[13] R. Misra, R. Wisniewski, C. S. Kallesøe, and M. L. Bujorianu, "Robust correlated equilibrium: Definition and computation," *arXiv preprint arXiv:2311.17592*, 2023.

[14] L. A. Rossman *et al.*, *EPANET 2: users manual*. US Environmental Protection Agency. Office of Research and Development . . . , 2000.

[15] S. Shin, S. Lee, S. J. Burian, D. R. Judi, and T. McPherson, "Evaluating resilience of water distribution networks to operational failures from cyber-physical attacks," *Journal of Environmental Engineering*, vol. 146, no. 3, p. 04020003, 2020.

[16] J. Stoustrup, "Plug & play control: Control technology towards new challenges," *European Journal of Control*, vol. 15, no. 3-4, pp. 311–330, 2009.

Chapter 5

Reinforcement Learning with Probabilistic Safety Guarantees

Summary *This chapter summarizes the contributions of this dissertation toward balancing the dual objectives of ensuring safety while maintaining optimality. We consider a Constrained Markov Decision Process (CMDP) with probabilistic safety constraints. The objective function of CMDP is to find a policy that minimizes the costs while the constraint ensures that all feasible policies restrict the safety function (encapsulates probability of visiting unsafe states) to be less than a predefined threshold p. This is referred to as p-Safety in papers [4, 5]. This chapter is primarily based on the Paper C [4].*

1 Formulation of p-Safe Reinforcement Learning problem

A Safety Constrained Markov decision process is defined by a tuple $(\mathcal{X}, U, \mathbb{P}, c, k, \mu, p)$, where

- $\mathcal{X} = \{x_1, x_2, \cdots, x_n\}$ is a finite set of states partitioned into transient states E, Goal states $\mathcal{G}$ and Unsafe states $\mathcal{O}$.
- $U = \{u_1, u_2, \cdots, u_m\}$, is a finite set of control actions.
- $\mathbb{P} : \mathcal{X} \times U \times \mathcal{X} \to \Delta(\mathcal{X})$ is the probabilistic state transition function.
- $c : \mathcal{X} \times U \to \mathbb{R}$, is the one-step cost function.

- $k : \mathcal{X} \times U \to [0, 1]$, is the one-step safety probability.
- $\mu : \mathcal{X} \to [0, 1]$ is the initial distribution of states.

Define $t = 1, 2, \cdots$ as the discrete time, and consider a discrete-time stochastic process $\{x(t)\}_{t=0}^{\tau}$ with randomized stopping time τ defined on the set of finite states $\mathcal{X}$. The control action taken at time t is $u(t) \in U$. Let the space of sample paths generated by $\{x(t)\}$, $\{u(t)\}$ denote the canonical sample space Ω of the considered stochastic process. We partition the state space $\mathcal{X}$ into transient states E, Goal States $\mathcal{G}$, and Unsafe states $\mathcal{O}$ (an example is shown in fig. 5.1). For a state sequence $\{x(1), x(2), \cdots, x(t)\}$ evolving on $\mathcal{X}$ according to $\mathbb{P}$, let $p \in [0, 1]$ be the probability of the state sequence hitting $\mathcal{O}$ i.e. the final state $x(t) \in \mathcal{O}$. The goal of the decision maker is to choose a sequence of actions that ensures the state sequence to *Reach* Goal states $\mathcal{G}$ and simultaneously *Avoid* Unsafe states $\mathcal{O}$. Since the considered stochastic process evolves probabilistically

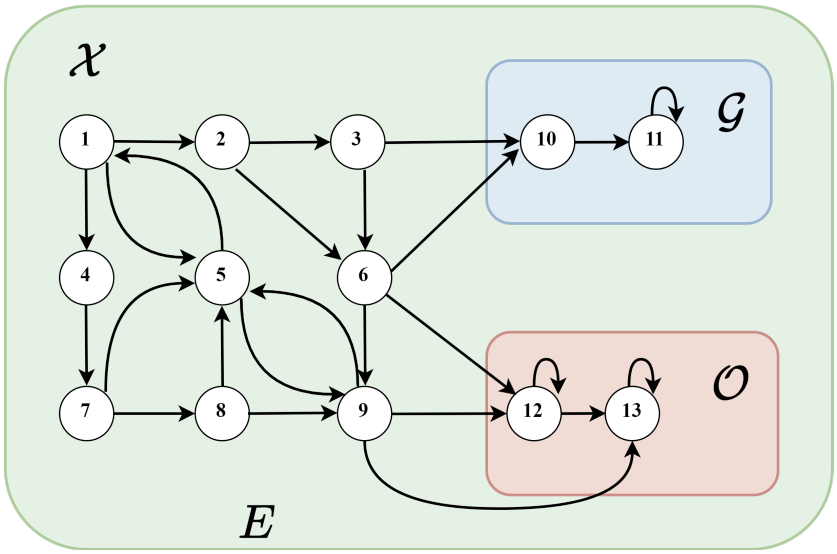


Fig. 5.1: A generic Reach-Avoid problem for MDP with states numbered $1, \cdots, 13$ (Source: [4]).

and our interest in the process is limited while it evolves on the set E, we need to consider randomized stopping time that quantifies the minimum time taken for a given state sequence to hit $\mathcal{G} \cup \mathcal{O}$. Let $\tau_{\mathcal{O}}$ represent the time taken for the state sequence to hit $\mathcal{O}$ as follows,

Definition 1.1 (First Hitting time of $\mathcal{O}$ [4]) The first hitting time of $\mathcal{O}$, $\tau_{\mathcal{O}}$ is the time taken for $x(t) \in \mathcal{O}$ given $x(1) \in E$,

$$\tau_{\mathcal{O}} := \left\{ \inf_t x(t) \in \mathcal{O} \mid x(1) \in E \right\}. \tag{5.1}$$

Similarly, we define the exit time as follows,

Definition 1.2 (Exit time [4]) The Exit time τ represents the time taken for $x(t) \notin E$ given $x(1) \in E$ i.e.,

$$\tau := \left\{ \inf_t x(t) \in \mathcal{G} \cup \mathcal{O} \mid x(1) \in E \right\}. \tag{5.2}$$

We consider Markov policies i.e. the policy only depends on the current state as per definition 2.12. Each policy π induces a unique transition probability matrix $\mathbb{P}(\pi)$ as follows,

$$P(\pi) := \sum_{u \in U} P_{ij}(u)\pi(u \mid i), \quad i, j \in E. \tag{5.3}$$

We can now formulate the p-Safety problem as the following CMDP. Consider the expectation operator $\mathbb{E}_\pi$ with respect to the transition probability matrix $P(\pi)$, than we can define p-Safe MDP as,

Definition 1.3 (p-Safe MDP [4]) For a given p, compute the policy $\pi : E \to \Delta(U)$ that solves

$$\begin{aligned} \min_\pi \ & V_\pi(i) := \mathbb{E}_\pi \left[\sum_{t=0}^{\tau_{\mathcal{O}}} c(x_t, u_t) \mid x_0 \right], \ x_0 \in E, \\ \text{s.t.} \ & S_\pi(i) := \mathbb{E}_\pi \left[\sum_{t=0}^{\tau_{\mathcal{O}}} k(x_t, u_t) \mid x_0 \right] \leq p, \ x_0 \in E. \end{aligned}$$

In the next section, we illustrate why the considered policy π needs to be randomized for CMDP.

2 The need for Randomized policies for CMDP

The following example demonstrates the need for having randomized policies.

2.1 Example (Optimization with Expected visits Constraint) Consider an MDP with a single state j and two control actions a and b as shown in fig. 5.2. If action a is chosen then with probability 1, the state goes to chain 1 with the cost of 20 units and if action b is chosen then with probability 1, the state goes to chain 2 at a cost of 10 units. Firstly, consider the unconstrained optimization problem, where we simply choose to minimize the costs i.e. $\min_{\{a,b\}} \mathbb{E}_P[\text{Cost}]$. Then, action b is the optimal action. Now, suppose that if the state is in chain 1 then the expected probability of visiting state S_1 is 0.05, and if the state is in chain 2, then the expected probability of visiting state S_2 is 0.1. Let us now consider the constrained optimization problem where besides minimizing the expected costs, we would also like to restrict the expected visits to the states S_1 and S_2 to be less than 0.075 i.e. $\min_{\{a,b\}} \mathbb{E}_P[\text{Cost}]$ s.t. $\mathbb{E}_P[\text{visits to } S_1 \cup S_2] \leq 0.075$. Now, action b is infeasible and we are forced to pay a higher cost of 20 units or can we do better? If we extend the decision space of policies to randomized policies, we will get a better solution, as follows. Consider a randomized policy $\pi : j \mapsto \Delta\{a, b\}$ where we assign 0.5 probability to choosing action a and the remaining probability for choosing action b. This policy is feasible as $\mathbb{E}_{P(\pi)}[\text{visits to } S_1 \cup S_2] = 0.5 \times 0.05 + 0.5 \times 0.1 = 0.075$ and is better than choosing action a, as the expected cost incurred by π is $\mathbb{E}_{P(\pi)}[\text{Cost}] = 0.5 \times 20 + 0.5 \times 10 = 15$. Thus, we obtain a better solution (in expectation) by extending the decision space from discrete policies to randomized policies.

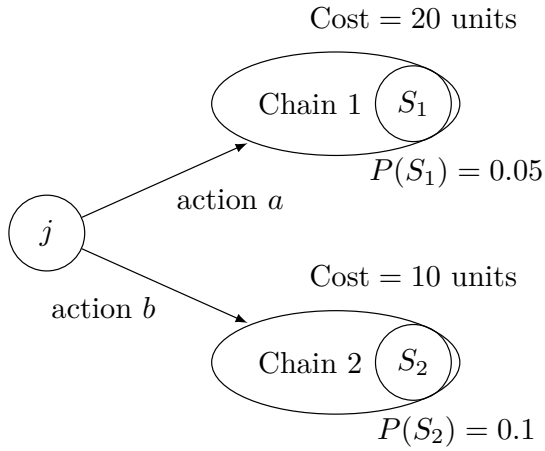


Fig. 5.2: The above example shows why randomized policies might give a better solution compared to deterministic policies for constrained Markov decision processes.

Besides the requirement of having randomized policies, the celebrated Bellman's Principle of Optimality also does not hold for CMDPs in general. This is demonstrated by the counterexample due to Haviv in [3] and is presented in Paper E. Using ideas from game theory such as subgame perfection [1] and Hoffman and Karp's Algorithm 2 for 2-player zero-sum games [2], we resolve the counterexample in Paper E by imposing sequential decision making at each state or decision node. We have designed the following Reinforcement Learning Algorithm based on the concept of subgame perfect equilibrium for Markov games and applied it to the CMDP setting in Paper C.

Algorithm 8 p-Safe Q-learning [4]

Input: ϵ, p, Max. number of episodes N_e
Initialize $Q(i, u, \lambda(i)) = 0 \ \forall i \in E, u \in U$
Initialize state-action counters f, g and learning rate $\alpha_i = 1 \ \forall i \in E$
for Each episode $e = 1 : N_e$ **do**
  Initialize state x_0 and policy $\pi_0 = \frac{1}{|U|}, \quad \forall u \in U$, and $\forall i \in E$
  while $t < \tau_{\mathcal{O}}$ **do**
    Given state x, Choose $u_t \leftarrow \begin{cases} u \sim \frac{1}{|U|} \text{ with probability } \epsilon \\ u \sim \pi_t(x) \text{ with probability } 1-\epsilon \end{cases}$
    Observe new state $x_{t+1} = i$, cost c_t after applying u_t
    Update $f_i^{t+1} \leftarrow f_i^t + 1$
    if $i \in \mathcal{O}$ **then**
      Update $g_i^{e+1} = g_i^{e+1} + 1$
      Update estimated safety cost $k_{e+1}(i, u_t) = \frac{g_i^{e+1}}{g_i^{e+1} + f_i^{t+1}}$
    end if
    Update learning rate $\alpha_i \leftarrow \frac{1}{f_i^{t+1}}$
    $\lambda_{t+1}(i) \leftarrow \arg\max_{\lambda \geq 0} Q_t(i, u_t, \lambda)$
    $\pi_{t+1}(i) \leftarrow \arg\min_{\pi \in \Delta(U), V} V \quad \text{s.t.} \sum_{u \in U} \pi(u) Q_t(i, u, \lambda_{t+1}(i)) \leq V$
    $val[L_t(i)] \leftarrow V$
    Update Q value

$$Q_{t+1}(i, u, \lambda(i)) \leftarrow (1 - \alpha_i) Q_t(i, u, \lambda(i)) + \alpha_i (d_t + val[L_t(i)])$$

    $t \leftarrow t + 1$
  end while
end for

The proposed algorithm also ensures that the decision space is extended to the space of randomized policies. We propose a lower bound on how long the algorithm should run for sufficiently learning a p-Safe optimal policy. Any episode that ensures runtime up to T in Proposition 6.1 is enough for learning an ε-optimal p-Safe policy.

Proposition 2.1 ([4]).
Consider the p-Safe MDP (C.4) *with unknown costs and transition probability matrix. Define*

$$c_M := \max_{i\in E, u\in U} c(i,u), \text{ and } \phi_M := \max_{i\in E, u\in U} \lambda(i)(k(i,u)-p),$$

where $\lambda(i) = \max_{t<\tau_{\mathcal{O}}} \lambda^t(i)$ *with* t *being the iteration time in Algorithm 12. Let* T *be the final time. For an* $\varepsilon > 0$,

$$\mathbf{L}^*_\pi - \mathbf{L}^T_{\pi^T} \leq \varepsilon, \text{ where } \mathbf{L}^T_{\pi^T} = val[\mathbf{L}^T] \text{ from Algorithm 12,}$$

if Assumption 5.1 is satisfied and Algorithm 12 runs upto time T *given by,*

$$T \geq \frac{1}{p}\log\left(\frac{c_M + \phi_M}{\varepsilon p}\right).$$

The proposition 2.1 gives bounds on the minimum running time of an episode for learning the ε-approximate Lagrangian but unfortunately does not give any bounds on the number of episodes that would be sufficient for learning the ε-approximate Lagrangian. However, it indirectly implies that if any episode runs for time period T, then it is sufficient. In practice, we can check after running E episodes whether any single episode ran for at least time T. Note that the lower bound on T is entirely dependent on the problem parameters c_M, ϕ_M, and the desired level of p-safety and accuracy ε.

3 Conclusions

We have constructed a Reinforcement Learning that obtains p-Safe optimal policies. We highlight the fact that standard Bellman's Principle of Optimality does not hold for the p-Safe CMDP in Paper C, and furthermore the policies need to be randomized. Applications of p-Safe optimality can be managing a portfolio of assets where p represents the probabilistic risk of asset depreciation that we are willing to take in order to ensure greater profits over the entire portfolio. An engineering application can be a water distribution network with an overhead tank that stores water and supplies water to the consumers when the operating costs of the pumping stations are higher or during drought conditions. The safety criteria can be ensuring that the water stored in the tank does not exceed the tank's physical capacity. Another criteria can be to ensure that the water is not stored for too long as this might lead to contamination. Another engineering application can be wind turbines where the blades of the wind turbine can tolerate visits to unsafe states up to a certain threshold (defined by p).

In future work, we would like to calculate the lower bound on the number of episodes. A promising idea towards that direction is to use the concept of Approachability where we consider the p-Safe RL problem as a bi-objective optimization problem with the two objectives being to minimize costs (optimality) and to minimize safety cost (safety). An episodic algorithm can then be constructed where we learn a policy each episode using a standard RL-algorithm such as $TD(\lambda)$. At the end of each episode, we can optimize the decision variable that encapsulates the balance between safety and optimality (by using projected gradient descent) such that we ensure Approachability to the set defined by $S_\pi(x(1)) \leq p$. The lower bound on episodes can be obtained by exploiting the finite sample guarantees that ensure ε-Approachability (see definition 2.10).

References

[1] T. Başar and G. Zaccour, *Handbook of dynamic game theory*. Springer, 2018.

[2] J. A. Filar and B. Tolwinski, "On the algorithm of pollatschek and avi-ltzhak," in *Stochastic Games and Related Topics: In Honor of Professor LS Shapley*. Springer, 1991, pp. 59–70.

[3] M. Haviv, "On constrained markov decision processes," *Operations research letters*, vol. 19, no. 1, pp. 25–28, 1996.

[4] R. Misra, R. Wisniewski, and C. S. Kallesøe, "On bellman's principle of optimality and reinforcement learning for safety-constrained markov decision process," *arXiv preprint arXiv:2302.13152*, 2023.

[5] R. Wisniewski and M. L. Bujorianu, "Probabilistic safety guarantees for markov decision processes," *IEEE Transactions on Automatic Control*, 2023.

Part II

Papers

Paper A

Approximating the model of a water distribution network as a Markov decision process

Rahul Misra, Rafał Wisniewski, Carsten S. Kallesøe

The paper has been published as a part of the conference proceedings for
10th Vienna International Conference on Mathematical Modelling MATHMOD 2022,
Volume 55, Issue 20, 2022, Pages 271-276.

Abstract

In this paper, our objective is to convert the model of a water distribution network described via Stochastic differential equations (SDE) into a Markov decision process (MDP). The motivation behind this work is that while MDP's represents the underlying dynamics for dynamic programming and reinforcement learning, the actual underlying model is best described via differential equations, and therefore, we would like to convert the SDE to MDP. We have applied Kushner's Markov chain approximation (MCA) method and verified it using a novel modified Monte Carlo method which can be considered as an alternative to the well-known Kushner's MCA. Both the methods approximate the value function and simulation studies show that the obtained value functions from both the methods converge to almost the same value.

Keywords: Modeling for control optimization, Markov decision process, Stochastic differential equations, Dynamic programming, Markov chain approximation, Monte Carlo methods.

1 Introduction

Efficient pressure management in a Water Distribution Network (WDN) is a complex control problem since it entails an inherently multi-input, multi-output system with control objectives of ensuring supply with minimal variance in pressure at demand side while ensuring energy efficiency of supply pumps. If the WDN is used for drinking water distribution than we would also need to control water quality (see [1] and the references therein). The WDN considered in this work consists of suppliers and consumers which are connected together by a piping network. Such a WDN can be modeled using graph theory which represents the topology of the network connecting individual components as stated in [2]. The uncertainty in consumption pattern is the reason for the stochastic nature of a water distribution network and therefore, we have used a stochastic differential equation (SDE) where the diffusion matrix takes into account the uncertainty due to demand side consumption. For solving stochastic control problems, Dynamic Programming (DP) is one of the fundamental mathematical tools and forms the basis of modern reinforcement learning (RL) algorithms such as Q-learning, SARSA, policy gradient, actor-critic etc. (see [3] and [4] for an introduction to dynamic programming and reinforcement learning algorithms). A common feature of all these algorithms is that they assume that the underlying system is a Markov decision process (MDP) or its generalization such as a Stochastic game in [5]. All the aforementioned settings have a finite number of states which makes it possible to represent the value of each state in a tabular representation. Bellman's optimality equation can than be used to find the optimal states and corresponding sequence of control actions constitute the optimal control policy. However in a practical setting such as a robot, satellite, HVAC system

or a water distribution network, the dynamics are modeled using differential equations derived from physical laws such as mass or energy conservation. The state space of such equations is infinite, thereby making traditional DP and RL methods not applicable in such settings. To overcome this limitation, function approximation based techniques such as least squares or a neural networks are used for approximating the value function or the control strategy or both simultaneously. Function approximation based RL algorithms come with some inherent challenges such as non-convergence to the target in the case of off-policy learning with bootstrapping, overfitting and instability in the case of neural networks (see chapters 9, 11 and references therein from [3]). The method of Markov chain approximation was developed in [6] as a numerical method for solving stochastic control problems and can be considered as an alternative to standard functional approximation techniques. This idea has previously been considered in [7] as *finite-differences RL*. However, there is no existing literature on practical applications of *finite-differences RL* and this forms a motivation for this work. The key contributions of this work are summarized as follows:

- Markov chain approximation is applied on a novel SDE model of a WDN so as to obtain an MDP.
- A novel Monte Carlo based method is developed as an alternative to Markov chain approximation.
- To the best of authors knowledge, this is the first attempt at modeling a WDN as an SDE.

The rest of the paper is organized as follows. In the next Section, we present a model of WDN described via SDE's. In Section 3, we approximate the value function for a fixed policy and derive the elements of probability transition matrix using the drift and diffusion terms of the model presented in Section 2. In Section 4, we derive the elements of probability transition matrix using a modified Monte Carlo method. In Section 5 we compare the results obtained from Section 3 and 4. Lastly, we present our conclusions and future work.

2 Modeling of WDN using SDE's

Consider the WDN network shown in fig. A.1. There are two pumping stations which supply water to pressure zones 1 and 2 via the network shown in fig. A.1. The aim of the pumping stations is to minimize variation in pressure at pressure zones 1 and 2. The model of the WDN is defined by state equations of the following type.

$$dx = \left(\mathrm{f}(x) + \sum_{k=1}^{\mathcal{N}} \mathrm{g}^k(x)u^k\right)dt + \sigma(x)d\mathrm{w}, \tag{A.1}$$

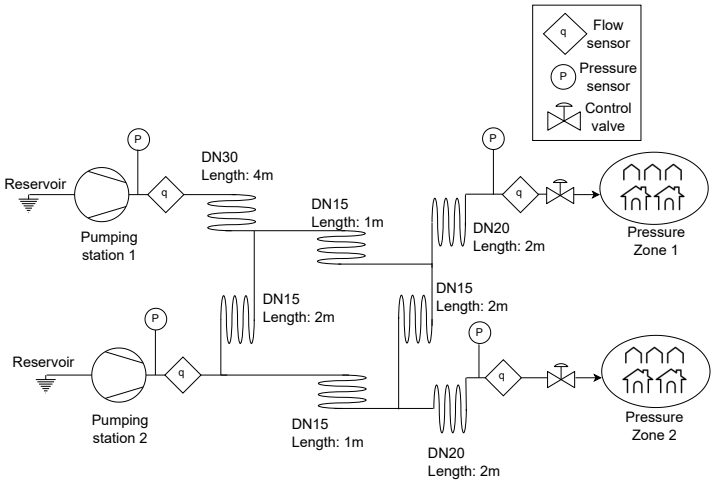


Fig. A.1: Process and Instrumentation diagram

where $x \in X \subseteq \mathbb{R}^n$ is a stochastic process which represents the free flows in the WDN, $u \in U \subseteq \mathbb{R}^p$ represents the pressure control action due to pumps, $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^p$ represent the drift, $\sigma : \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^l$ represents the diffusion with $a(\cdot) = \sigma(\cdot)\sigma(\cdot)^T$ being the corresponding diffusion matrix, w represents the standard Wiener process on $\mathbb{R}^l$. The WDN is modeled by a directed graph $\Gamma = \{N, E\}$ in fig. A.2 where $N = \{n_1, \cdots, n_{11}\}$ represents the nodes or vertices and $E = \{e_1, \cdots, e_{11}\}$ represents the edges where components of WDN such as pipes, pumps and valves are connected. The pumps in fig. A.1 are represented as edges e_{10} and e_{11} in fig. A.2 and the demands are represented by edges e_8 and e_9. We begin by defining the incidence matrix H which encodes the interconnections between different components of the WDN as shown in the fig. A.1.

$$H_{i,j} = \begin{cases} -1, & \text{if the } j^{th} \text{ edge is entering } i^{th} \text{ node,} \\ 0, & \text{if the } j^{th} \text{ edge is not connected to the } i^{th} \text{ node,} \\ 1, & \text{if the } j^{th} \text{ edge is leaving } i^{th} \text{ node,} \end{cases} \tag{A.2}$$

and the cycle matrix B which encodes the information about the edges which belong to cycles (or loops) and their orientation.

$$B_{i,j} = \begin{cases} -1, & \text{if the } j^{th} \text{ edge belongs to the } i^{th} \text{ cycle and their directions disagree,} \\ 0, & \text{if the } j^{th} \text{ edge does not belong to the , } i^{th} \text{ cycle,} \\ 1, & \text{if the } j^{th} \text{ belongs to the } i^{th} \text{ cycle and their directions agree.} \end{cases} \tag{A.3}$$

The model is defined with following assumptions.

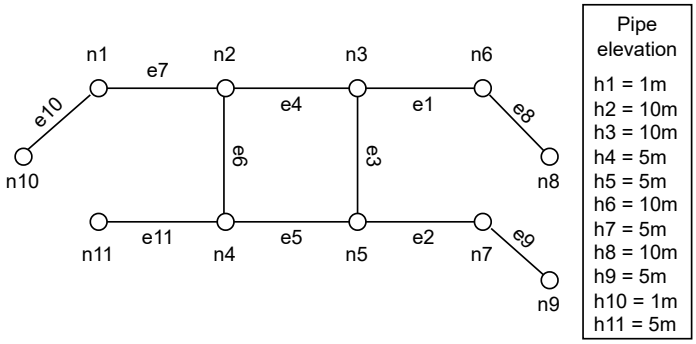


Fig. A.2: Graph of the WDN in fig. A.1 with pipe elevation $\bar{z}$.

Assumption 2.1 The graph G is a connected graph.

Assumption 2.2 We require existence and uniqueness of solutions of (A.1). Strong existence holds if for a given probability space $(\Omega, \mathcal{F}, P)$, a filtration $\mathcal{F}_t$, an $\mathcal{F}_t$-Wiener process w and an $\mathcal{F}_0$- measurable initial condition $x(0)$, there exists an $\mathcal{F}_t$-adapted process $x(t)$ satisfying (A.1) for all $t \geq 0$. Furthermore, uniqueness holds if for any two sample paths $x_1(t), x_2(t)$, $P\{x_1(0) = x_2(0)\} = 1 \implies P\{x_1(t) = x_2(t) \forall t \geq 0\} = 1$.

We shall now derive the model of the WDN in a deterministic sense for simplicity. Thereafter, the model is extended to include stochastic noise on the states as consumption of water is uncertain. Let $\mathcal{T}$ be a arbitrary spanning tree of the graph G and let $\mathcal{C}$ be the chords of the same. $H_{\mathcal{T}}$ and $H_{\mathcal{C}}$ represent the incidence matrix of the spanning

tree $\mathcal{T}$ and corresponding chords $\mathcal{C}$ respectively. For a WDN with graph G, H can be partitioned as $H = \begin{bmatrix} H_{\mathcal{C}} & H_{\mathcal{T}} \end{bmatrix}$. We now need to choose a reference node. The reference node is chosen to be one of the supply nodes (n_1 or n_4) since it is only at the nodes with non-zero demands that water can flow in or out of the network. In this setup, we have chosen the reference node as n_1 and defined $\bar{H}_{\mathcal{T}}$ as the reduced incidence matrix of $\mathcal{T}$ (obtained by removing the row corresponding to the reference node) and $\bar{H}_{\mathcal{C}}$ as the corresponding chord matrix. We then define the free flow vectors $q_{\mathcal{C}}$ (flow in one of the chords for example e_6) and $d \in \mathbb{R}^{n-1}$ (external flows e_8, e_9 and e_{11} excluding the flow e_{10} in the reference node),

$$q = B^T q_{\mathcal{C}} + B_d^T F d, \tag{A.4}$$

where $B = \begin{bmatrix} I & -\bar{H}_{\mathcal{C}}^T \bar{H}_{\mathcal{T}}^{-T} \end{bmatrix}$ is the standard cycle matrix ($\bar{H}_{\mathcal{T}}$ is always invertible as given in [8]), $B_d = \begin{bmatrix} 0 & \bar{H}_{\mathcal{T}}^{-T} \end{bmatrix}$ represents the loop matrix which includes free flows and F is a matrix of 1's and 0's which selects the consumption nodes (n_6 and n_7) in G (fig. A.2). The individual components of a water distribution network are characterized by the pressure drop across them (analogous to the voltage drop across resistors, inductors and capacitors in an electrical circuit). For a pipe element k in the network having length L, cross-section area A with water density ρ, the pressure drop across its ends i, j is given by the following equation.

$$p_i - p_j = J\dot{q_k} + Rf(q_k) - \rho g \Delta z_k, \tag{A.5}$$

where $J = \frac{L\rho}{A}$, q_k is the flow in pipe k, g is the gravitational constant, R is a diagonal matrix which represents the friction factor of the pipes, $f_i(q) = |q_i|\, q_i$ and $\Delta z_k = z_i - z_j$ is the difference in geodesic level of the pipe. The term $f_i(q)$ represents the nonlinear pipe resistance to flow q and it preserves the direction of flow (see [9]). The parameters of friction factor R can be calculated as per Chapter 2 in [10]. Equation (A.5) is a consequence of Newton's second law of motion and is derived in equation 2.4 in [11]. The pump is modeled as a centrifugal pump and can be represented as a positive pressure difference u_k across the k^{th} pump as shown in the following equation,

$$p_i - p_j = u_k. \tag{A.6}$$

This is a simplified model which considers pressure drop across pump element. We can now define the state vector for (A.1) consisting of free flows as $x = \begin{bmatrix} q_{\mathcal{C}} & d \end{bmatrix}^T$. The pump and pipe dynamics can be combined together as drift terms f and g in (A.1) as follows.

$$\mathrm{f}(x(t)) + \mathrm{g}(x(t)) = -{J_{ex}}^{-1}(A_{ex}(Rf(q) - \rho g G u) + \rho g F_{ex}\,(\bar{z} - z_0)), \tag{A.7}$$

where

$$J_{ex} = \begin{bmatrix} BJB^T & BJB_d^T F \\ F^T B_d J B^T & F^T B_d J B_d^T F \end{bmatrix},$$

$$A_{ex} = \begin{bmatrix} I & -\bar{H}_{\mathcal{C}}^{T}\bar{H}_{\mathcal{T}}^{-T} \\ 0 & F^{T}\bar{H}_{\mathcal{T}}^{-T} \end{bmatrix},$$

G is a matrix of 1's and 0's which selects the edges in G (fig. A.2) on which controllers are connected (in this case the suppliers), F_{ex} is a matrix of 1's and 0's which selects the edges in G (fig. A.2) on which the demands are connected (in this case the pressure zones), $\bar{z}$ is the physical height of each node as shown in fig. A.2, z_0 is the height of reference node. Note that, the matrix J_{ex} and A_{ex} combine the incidence matrix H and cycle matrices B and B_d with component models. A detailed derivation of (A.7) is given in Appendix A. The matrices J and R can also be partitioned as $J = \begin{bmatrix} J_{\mathcal{C}} & J_{\mathcal{T}} \end{bmatrix}$ and $R = \begin{bmatrix} R_{\mathcal{C}} & R_{\mathcal{T}} \end{bmatrix}$ respectively. The stochastic nature of consumption at nodes n_8 and n_9 in fig. A.2 is captured as a Wiener process.

Assumption 2.3 The diffusion matrix is a diagonal matrix implying that the state noise are uncorrelated with each other.

For small piping networks, the assumption 2.3 is unrealistic since the flows will be strongly correlated but for large networks spanning over large geographical distances, the assumption 2.3 is a suitable relaxation since aggregated demand of end-users can be treated as a periodic disturbance and rejected as done in [1]. The output equation (A.10) represents the pressure measurement at a selected node in the WDN and is derived from equations (A.5) and (A.6) by replacing p_i, p_j with $\bar{p}, p_0$. The output equation is derived as follows.

$$y = C\left(\bar{p} - p_0\right) = C\bar{H}_{\mathcal{T}}^{-T}J_{\mathcal{T}}\dot{q}_{\mathcal{T}} + C\bar{H}_{\mathcal{T}}^{-T}R_{\mathcal{T}}\left(q_{\mathcal{T}}\right) - \rho g C\left(\bar{z} - z_0\right) - \rho g C\bar{H}_{\mathcal{T}}^{-T}Gu, \quad \text{(A.8)}$$

where C is a matrix of 1's and 0's which selects the measured quantities in G (fig. A.2), $\dot{(\cdot)}$ notation represents the usual time derivative $\frac{d(\cdot)}{dt}$, $q_{\mathcal{T}}$ is the flow in the spanning tree $\mathcal{T}$ of G in fig. A.2 (it includes both chord flow q_c and consumer flow d), $R_{\mathcal{T}}$ is the associated friction factor matrix and p_0 is the reference pressure. Now, let $\Pi_{\mathcal{T}}$ project the component flows q onto the tree flow $q_{\mathcal{T}}$, such that $\dot{q}_{\mathcal{T}} = \Pi_{\mathcal{T}}\dot{q}$. This means that

$$\bar{p} - p_0 = \bar{H}_{\mathcal{T}}^{-T}J_{\mathcal{T}}\Pi_{\mathcal{T}}\left(B^{T}\dot{q}_{\mathcal{C}} + B_d^{T}F\dot{d}\right) + \bar{H}_{\mathcal{T}}^{-T}R_{\mathcal{T}}\left(\Pi_{\mathcal{T}}\left(B^{T}q_{\mathcal{C}} + B_d^{T}Fd\right)\right) - \rho g\left(\bar{z} - z_0\right) - \rho g\bar{H}_{\mathcal{T}}^{-T}Gu. \quad \text{(A.9)}$$

Finally define $T_r = \begin{bmatrix} B^{T} & B_d^{T}F \end{bmatrix}$ then

$$\bar{p} - p_0 = \bar{H}_{\mathcal{T}}^{-T}J_{\mathcal{T}}\Pi_{\mathcal{T}}Tdx + \bar{H}_{\mathcal{T}}^{-T}R_{\mathcal{T}}\Pi_{\mathcal{T}}T_r x - \rho g\left(\bar{z} - z_0\right) - \rho g\bar{H}_{\mathcal{T}}^{-T}Gu. \quad \text{(A.10)}$$

This concludes the section on modeling of WDN. In the remaining paper, we will approximate solution of (A.1).

3 Markov chain approximation

In [6], the authors have presented a numerical method referred to as Markov chain approximation (MCA) for solving stochastic control problem. MCA finds the reachable states under a given control policy and calculates a probability transition matrix via application of finite differences method on the Hamilton-Jacobi-Bellman equation. At any time t, the number of reachable states depends on the time step ahead (i.e. interpolation time) and MCA method automatically calculates the interpolation time. If the dynamics and/or diffusion are large in magnitude at a given time, the interpolation time becomes smaller. For this reason, MCA can be considered as a "successive approximation" technique as the probability transition matrix will not be stationary unless the steady state is reached (see fig. A.3 for a visual representation of the same). The calculated probabilities are a function of the dynamics alone and therefore this method can also be applied for representing a dynamical system described via differential equations into a Markov decision process.

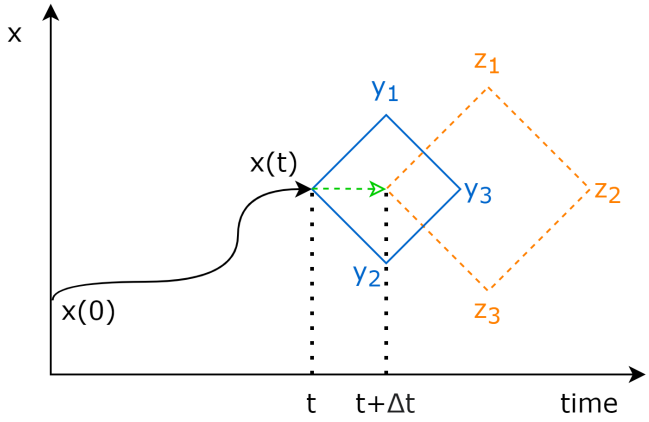


Fig. A.3: Illustration of MCA as a successive approximation. Here we see a sample path of (A.1) starting at $x(0)$ and reaching $x(t)$ at time t, y_1, y_2 and y_3 represent the reachable states for $x(t)$ in time interval Δt. Any realized state $x(t + \Delta t)$ (shown by the green arrow) can be found as a convex combination of extremities of blue polygon. Once the next state $x(t+\Delta t)$ is reached the process repeats as shown by orange polygon with reachable states z_1, z_2 and z_3.

We will now formally approximate (A.1) as an Markov decision process. We begin by defining a state grid for the WDN based on normalizing the amount of water flowing in the network. Consider the k^{th} edge of the WDN denoted by e_k. Suppose that the total amount of water which can flow through e_k is Q. Than by considering how much percentage of Q is flowing through e_k, we can assign a number between 1 to 100 indicating the flow through e_k. This allows us to construct a grid where all free flows

are normalized. We will now state the stochastic control problem. We begin by defining the cost function. The aim of the pumping station 1 and 2 in fig. A.1 is to minimize variation in pressure at pressure zones 1 and 2 with minimal control input and this is represented in their cost functions and can be stated as follows.

$$c(t) = (\bar{p} - p_0)^T(\bar{p} - p_0) + u^T u. \tag{A.11}$$

Note that the cost function is dependent on the states, despite there being no explicit state variable due to the output equation (A.10). Since, MCA is a successive approximation technique, we need to define a local reachable state space. Let $\mathcal{B}$ be the boundary of such a state space. The discounted stochastic control problem can now be defined as follows.

$$\min_{u} \quad \mathop{\mathbb{E}}_{x,u(\cdot)} \left(\int_s^\tau \gamma^t c\left(x(t), u(t)\right) dt + \gamma^\tau c_\tau\left(x_\tau\right) \right) \mid x(s) \tag{A.12a}$$

$$\text{s.t. } dx(t) = (\mathrm{f}(x(t)) + \mathrm{g}(x(t))u(t))dt + \sigma d\mathrm{w}(t), \tag{A.12b}$$

$$y(t) = \mathrm{h}(x(t)) + \mathrm{d}(x(t))u(t), \tag{A.12c}$$

$$u(t) \in U, \quad x(t) \in \mathcal{B}, \tag{A.12d}$$

where $x(s)$ is the given state at some starting time s, $\tau = \min\{t : x(t) \notin \mathcal{B}\}$ represents the minimum time taken to exit the local state space, γ is the discount factor and U is the admissible controls. Let the stochastic integral in the objective function of (A.12) be denoted by

$$Q(x, u(\cdot)) = \mathop{\mathbb{E}}_{x,u(\cdot)} \left(\int_s^\tau \gamma^t c\left(x(t), u(t)\right) dt + \gamma^\tau c_\tau\left(x_\tau\right) \right). \tag{A.13}$$

The optimal value function can than be obtained by minimizing $Q(x, u(\cdot))$ with respect to $u(\cdot)$ for each state as follows

$$V(x) = \min_{u(\cdot)} Q(x, u(\cdot)). \tag{A.14}$$

Let $\xi(s) \in \mathbb{R}^n$ denote the state $x(s)$ at some time s, than the next reachable discrete state in the set $\mathcal{B}$ is denoted by $\xi(s+1) = x(s) + e_i h$, where h is an approximation parameter which represents how coarse or fine our state-grid is and $e_i \in \mathbb{R}^n$ is the basis vector in i^{th} direction of the state space. The grid is shown in fig. A.4. Note that we are considering only state transitions of the type $x \pm e_i h$ and not of the type $x \pm e_i h \pm e_j h$ since the interpolation time is small (depending on the drift and diffusion as we shall see later) and the diffusion matrix is diagonal. We shall now apply the well-known Ito's lemma (which can be thought of as chain rule for SDE's) in order to obtain evolution

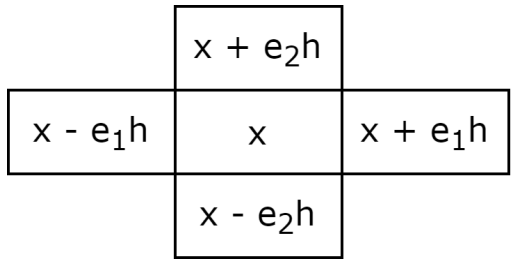


Fig. A.4: Illustration of state space as a grid. Any state whose numerical value is within the boundary of a particular grid box is considered as a part of the same.

of $Q(x, u(\cdot))$ with time. We begin by defining the differential operator $\mathcal{L}$ as follows. Let $a_{ij}(x)$, $i, j = 1, \cdots, n$ be an element of the diffusion matrix $a(x)$, than we can define $\mathcal{L}$ for (A.1) as follows.

$$\mathcal{L} = \frac{1}{2}\sum_{i,j=1}^{n} a_{ij}(x)\frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^{n}(\mathrm{f}_i(x) + \mathrm{g}_i(x)u)\frac{\partial}{\partial x_i} \tag{A.15}$$

Applying Ito's differential operator on (A.12) gives us the stochastic analogue of the well-known Hamilton-Jacobi-Bellman equation as follows.

$$\begin{aligned} &\mathcal{L}V(x(t)) + \gamma^t c(x(t), u(t)) = 0, \\ &V(x(\tau)) = c(x(\tau)), V(x(s)) = c(x(s)), \end{aligned} \tag{A.16}$$

where $V(x(\tau)) = c(x(\tau))$ and $V(x(s)) = c(x(s))$ represents the boundary condition. Expanding the differential operator in (A.16) gives us,

$$\frac{1}{2}\sum_{i,j=1}^{n} a_{ij}(x)\frac{\partial^2 V(x)}{\partial x_i \partial x_j} + \sum_{i=1}^{n}(\mathrm{f}_i(x) + \mathrm{g}_i(x)u)\frac{\partial V(x)}{\partial x_i} + \gamma^t c(x, u) = 0, \tag{A.17}$$

where we have written $x(t), u(t)$ as x, u for notational simplicity. In the sequel, we shall refer to the drift term $\mathrm{f}_i(x) + \mathrm{g}_i(x)u$ as $b_i(x, u)$ for the sake of notational simplicity. We now apply finite-differences with space approximation parameter h on (A.17) as follows.

$$\begin{aligned} \frac{1}{2}\sum_{i,j=1}^{n} a_{ij}(x)\frac{V(x + e_i h) + V(x - e_i h) - 2V(x)}{h^2} + \sum_{i=1}^{n} |b_i(x, u)|^+ \frac{V(x + e_i h) - V(x)}{h} \\ - \sum_{i=1}^{n} |b_i(x, u)|^- \frac{V(x) - V(x - e_i h)}{h} + \gamma^t c(x, u) = 0, \end{aligned} \tag{A.18}$$

where $|b_i(x,u)|^+ = \max(b_i(x,u),0)$ and $|b_i(x,u)|^- = \max(-b_i(x,u),0)$. This is the standard "upwind" scheme in numerical analysis of hyperbolic partial differential equations (see [12]). The intuition behind this scheme is that the approximation of V should be in the same direction as the drift of (A.1). It can also be verified that $|b_i(x,u)|^+ + |b_i(x,u)|^- = |b_i(x,u)|$. Since the diffusion matrix is diagonal as per assumption 2.3, we can rearrange the terms in (A.18) so as to obtain elements of probability transition matrix $p(x, x \pm e_i h \mid u)$ and interpolation time Δt as follows.

$$V(x) = \underbrace{\frac{a_{ii}(x)/2 + h\,|b_i(x,u)|^+}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x,u)|)}}_{p(x,x+e_ih|u)} V(x+e_ih) + \underbrace{\frac{a_{ii}(x)/2 + h\,|b_i(x,u)|^-}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x,u)|)}}_{p(x,x-e_ih|u)} V(x-e_ih) + \underbrace{\frac{h^2}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x,u)|)}}_{\Delta t} \gamma^t c(x,u). \quad \text{(A.19)}$$

The probability of state remaining unchanged is given by

$$p(x, x \mid u) = 1 - (p(x, x+e_ih \mid u) + p(x, x-e_ih \mid u)). \quad \text{(A.20)}$$

Thus, we obtain an MDP which is defined by the tuple $(\mathcal{X}, \mathcal{U}, P, \gamma, \mathcal{C})$ where $\mathcal{X} = \{x, x \pm e_i h \mid i = 1, \cdots, n\}$, $\mathcal{U}$ are discreatized control actions $(1, \cdots, 100$ in (A.6) which is the normalized pump pressure), P is the probability transition matrix whose elements have been calculated in (A.19), $\gamma \in [0,1]$ is the discount factor and $\mathcal{C} = \gamma^t c(x,u)\Delta t$ is the stage cost.

4 Verification using modified Monte Carlo method

The convergence proof for MCA is given in [6] and is based on probabilistic arguments whose intuition is explained in fig. A.3. However, in practice the matrix P will be sub-stochastic i.e. the sum of row entries will be slightly < 1. This is due to the small probability of some sample path of (A.1) escaping our considered reachable state grid. Thus, we would like to verify in practice how good the approximation (A.19) is and we do that using a modified Monte Carlo method stated in Algorithm 9. Algorithm 9 is based

on the sample paths of (A.1) and it calculates transition probabilities based on number of times a grid square $\Delta^h x(s)$ (with the entire state space being represented by the grid $\Delta^h x$) corresponding to state $x(s)$ is visited. This is in contrast to MCA which calculates transition probabilities based solely on drift and diffusion terms of (A.1) as done in (A.19). The sample paths of (A.1) are realized using Euler-Maruyama method (see [13])

Algorithm 9 Monte Carlo method for constructing P

1: **Input**: State $\xi(s) = x(s)$, Sate space grid $\Delta^h x$, Maximum time T for a sample path, No. of sample paths to be evaluated M
2: **for** all sample paths $m \leq M$ **do**
3: **while** $s < T$ **do**
4: Find grid square $\Delta^h x(s)$ in $\Delta^h x$ which corresponds to $x(s)$
5: Increment counter $MC(\Delta^h x(s))$ corresponding to $\Delta^h x(s)$ by 1
6: Obtain realization of $x(s+1)$ using modified Euler-Maruyama method with interpolation time Δt
7: $s = s + \Delta t$
8: **end while**
9: **for** all $i \leq n$, find $\xi_i(s+1)$ based on drift **do**
10: **if** $b_i(x,u) \geq 0$ **then**
11: $\xi_i(s+1) = \xi(s) + 1$
12: **else**
13: $\xi_i(s+1) = \xi(s) - 1$
14: **end if**
15: **end for**
16: $p(\xi_i(s), \xi_i(s+1)) = \frac{MC_i(\Delta^h x(s+1))}{\sum_{i=1}^{n} MC_i(\Delta^h x)} \ \forall i < n$
17: $p(\xi(s), \xi(s)) = 1 - \left(\sum_{i=1}^{n} \frac{MC_i(\Delta^h x(s+1))}{\sum_{i=1}^{n} MC_i(\Delta^h x)} \right)$
18: **end for**

modified with interpolation time Δt obtained via (A.19). The interpolation time Δt is used instead of standard equally spaced time intervals so as to retain similar time intervals as MCA for comparison. Furthermore Algorithm 9, finds next state component $\xi_i(s+1)$ on the grid $\Delta^h x$ based on the sign of the corresponding i^{th} component of the drift vector $b_i(x,u)$. This is done for ensuring that both Algorithm 9 and MCA are looking at the same future state. Since Algorithm 9 is a Monte Carlo based method, it requires evolution of a complete sample path (from $x(0)$ to $x(T)$, where $[0,T]$ is the time interval for numerical evaluation of (A.1)) prior to constructing P. This implies that we cannot compare P obtained from both the methods directly since they are evaluated at different

times. This motivates us to consider the approximate value function constructed by both the methods as a means of comparison. The approximate value function from modified Monte Carlo method is obtained by substituting the probabilities obtained from Algorithm 9 into (A.19) as coefficients of $V(x + e_i h)$ and $V(x - e_i h)$.

5 Simulation Results

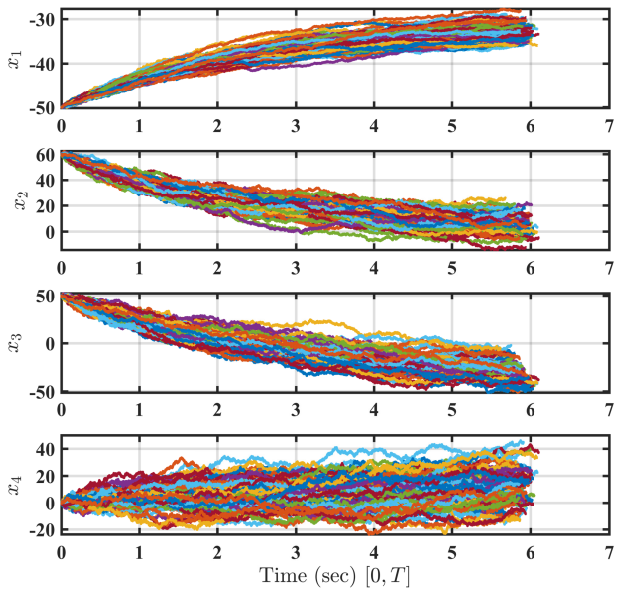


Fig. A.5: Sample paths for initial $x = [-50\ 60\ 50\ 0]^T$

In fig. A.5, we see 100 sample paths of (A.1) obtained using modified Euler-Maruyama method. These sample paths form the basis for calculating value function using Algorithm 9. In fig. A.6, we see convergence of value function obtained from MCA in upper subplot and convergence of value function obtained from Algorithm 9 in the lower subplot. Furthermore, we also simulated both the methods for different initial conditions and the results are summarized in the following table A.1.

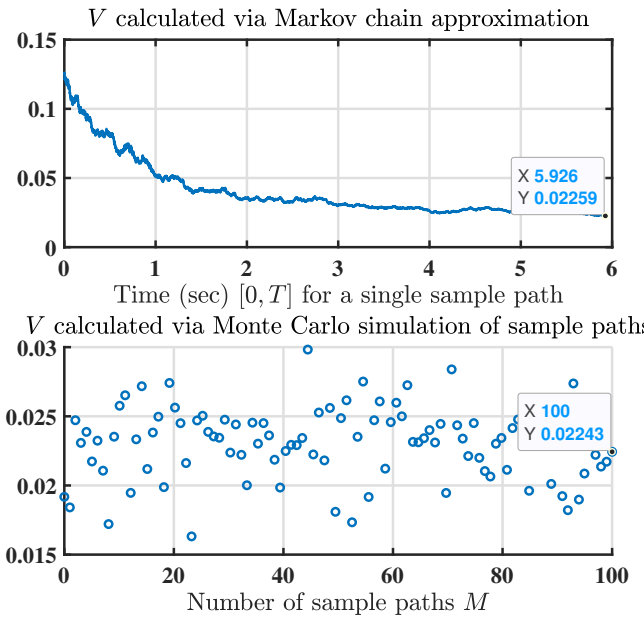


Fig. A.6: The obtained value functions converge to almost same value.

Initial condition	V obtained from MCA	V obtained from Algorithm 9
$x(0) = [-60\ 60\ 50\ 50]^T$	0.0170	0.0168
$x(0) = [-50\ 50\ 30\ 60]^T$	0.0157	0.0158
$x(0) = [-50\ 60\ 50\ 10]^T$	0.0194	0.0198

Table A.1: Value functions obtained for different initial conditions

6 Conclusions and Future Work

In this paper, we have presented a model of WDN and converted the same to an MDP using MCA. We have also proposed Algorithm 9 as an alternative for MCA. As discussed earlier, the assumption of diffusion matrix being diagonal is not realistic for small WDN and therefore we would like to estimate the coefficients of diffusion matrix from real-time data. MCA can than be applied to non-diagonal diffusion matrix by modifying

(A.19) to account for non-diagonal terms. Furthermore, we can now apply reinforcement learning methods directly to the obtained MDP. In future, we would like to apply MCA for WDN's with dynamic games as discussed in [14].

Acknowledgements

Financial support from the Poul Due Jensen Foundation (Grundfos Foundation) for this research is gratefully acknowledged.

References

[1] M. M. Polycarpou, J. G. Uber, Z. Wang, F. Shang, and M. Brdys, "Feedback control of water quality," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 68–87, 2002.

[2] M. Tahavori, C. S. Kallesøe, J. Leth, and R. Wisniewski, "Modeling of water supply systems: Circuit theoretic approach," in *2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2012, pp. 1561–1566.

[3] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[4] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, 2005, vol. 1,2, no. 2.

[5] J. Filar and K. Vrieze, *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

[6] H. J. Kushner and P. G. Dupuis, *Numerical methods for stochastic control problems in continuous time*. Springer Science & Business Media, 2001, vol. 24.

[7] R. Munos and P. Bourgine, "Reinforcement learning for continuous stochastic control problems," in *NIPS*, 1997, pp. 1029–1035.

[8] N. Deo, *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2017.

[9] T. N. Jensen and R. Wisniewski, "Global practical stabilisation of large-scale hydraulic networks," *IET control theory & applications*, vol. 5, no. 11, pp. 1335–1342, 2011.

[10] P. K. Swamee and A. K. Sharma, *Design of water supply pipe networks*. John Wiley & Sons, 2008.

[11] M. M. Pétursson Geir Bjarni, *Energy Optimisation of Water Distribution Networks: Using a distributed pumping solution for optimal pressure control.* Department of Control Engineering, Aalborg University, 2015.

[12] R. J. LeVeque, *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems.* SIAM, 2007.

[13] P. E. Kloeden and E. Platen, "Stochastic differential equations," in *Numerical Solution of Stochastic Differential Equations.* Springer, 1992, pp. 103–160.

[14] H. J. Kushner, "Numerical approximations for stochastic differential games," *SIAM journal on control and optimization*, vol. 41, no. 2, pp. 457–486, 2002.

A Derivation of drift equation

The total pressure drop Δp across all the components of WDN is obtained by combining (A.5) and (A.6) as follows.

$$\Delta p = J\dot{q} + Rf(q) - \rho g \Delta z + u. \tag{A.21}$$

Recall Kirchhoff's mesh law which is expressed as $B\Delta p = 0$ which is combined with (A.21) as follows.

$$0 = BJ\dot{q} + BRf(q) - B\Delta z + B\rho g G u, \tag{A.22}$$

where q is a linear function of q_c and d as per (A.4). The rest of the derivation will consider pressure at the consumer side. Partition $G = \begin{bmatrix} G_{\mathcal{C}} & G_{\mathcal{T}} \end{bmatrix}$. Consider the pressure equation (A.9). The pressure at all nodes open to atmosphere $F^T\bar{p}$ can be set to 0, where F selects the nodes open to the atmospheric pressure. This means that

$$\begin{aligned} 0 = F^T\bar{p} = F^T\bar{H}_{\mathcal{T}}^{-T} J_T \dot{q}_T + F^T\bar{H}_{\mathcal{T}}^{-T} R_{\mathcal{T}} f_{\mathcal{T}}(q_{\mathcal{T}}) \\ - \rho g F^T\bar{H}_{\mathcal{T}}^{-T}\Delta z_{\mathcal{T}} + F^T\bar{H}_{\mathcal{T}}^{-T} G_{\mathcal{T}} u, \end{aligned} \tag{A.23}$$

which together with definition of $B = \begin{bmatrix} I & -\bar{H}_{\mathcal{C}}^T\bar{H}_{\mathcal{T}}^{-T} \end{bmatrix}$ and $B_d = \begin{bmatrix} 0 & \bar{H}_{\mathcal{T}}^{-T} \end{bmatrix}$ gives us

$$\begin{aligned} \underbrace{\begin{bmatrix} BJB^T & BJB_d^T F \\ F^T B_d J B^T & F^T B_d J B_d^T F \end{bmatrix}}_{J_{ex}} \begin{bmatrix} \dot{q}_C \\ \dot{d} \end{bmatrix} = -\underbrace{\begin{bmatrix} I & -\bar{H}_C^T\bar{H}_T^{-T} \\ 0 & F^T\bar{H}_T^{-T} \end{bmatrix}}_{A_{ex}} \\ \left(\begin{bmatrix} R_C \\ R_T \end{bmatrix} f(q) - \rho g \begin{bmatrix} G_C \\ G_T \end{bmatrix} u \right) + \rho g \underbrace{\begin{bmatrix} 0 \\ F^T \end{bmatrix}}_{F_{ex}} (\bar{z} - z_0) \end{aligned} \tag{A.24}$$

Paper B

Approximating solution of stochastic differential games for distributed control of a water network

Rahul Misra, Rafał Wisniewski, Carsten S. Kallesøe

The paper has been published as a part of the conference proceedings for *18th IFAC Workshop on Control Applications of Optimization CAO 2022 Gif sur Yvette, France, 18–22 July,* 2022, Volume 55, Issue 16, 2022, Pages 110-115.

Abstract

In this paper, our objective is to design a distributed optimal control for pumping stations operating in a Water Distribution Network (WDN), where we would like to satisfy consumer demands with minimum energy consumption. The WDN has been modeled using graph theory and stochastic differential equations. This leads to a non-zero sum stochastic differential game. We have approximated the solution of the aforementioned game using Markov chain approximation and combined it with Shapley's algorithm so as to obtain Minimax mixed strategies. Minimax solution can be obtained as a distributed computation at the pumping stations without any knowledge of the costs incurred by the other pumping stations. Simulation results on the water network show convergence to an approximate Minimax solution.

Keywords: Stochastic Games, Stochastic differential equations, Dynamic programming, Markov chain approximation, Water distribution networks.

1 Introduction

Efficient pressure management in a Water Distribution Network (WDN) is a complex control problem since it entails an inherently multi-input, multi-output system with control objectives of ensuring supply with minimal variance in pressure at demand side while ensuring energy efficiency of supply pumps. The WDN considered in this work consists of suppliers and consumers which are connected together by a piping network. Such a WDN can be modeled using graph theory which represents the topology of the network connecting individual components as stated in [1]. The uncertainty in consumption pattern is the reason for the stochastic nature of a water distribution network and therefore, we have used Stochastic Differential Equations (SDE) where the diffusion matrix takes into account the uncertainty due to demand side consumption.

For solving stochastic control problems, Dynamic Programming (DP) is one of the fundamental mathematical tools and forms basis of Reinforcement learning (RL) algorithms (see [2] for an introduction to dynamic programming). A generalization of DP to include multiple controllers called *Stochastic Games* was introduced in [3]. The survey paper [4] and the book [5] provide an overview of development in the field of Stochastic Games. All the aforementioned settings have a finite number of states which makes it possible to represent the value of each state in a tabular representation. Bellman's optimality equation (or Shapley's equation in the case of Stochastic Games) can than be used to find the optimal states and the corresponding sequence of control actions which constitute the optimal control policy. However, in a practical setting such as a robot, satellite, HVAC system or a water distribution network, the dynamics are modeled using differential equations derived from physical laws such as mass or energy conservation. The state space of such equations is infinite, thereby making the aforementioned

algorithms not applicable in such settings. To overcome this limitation, function approximation based techniques such as least squares or a neural networks are used for approximating the value function or the control strategy or both simultaneously. Function approximation based RL algorithms come with some inherent challenges such as non-convergence to the target (see chapters 9, 11 and references therein from [6]) and non-convergence to the saddle point in the case of linear quadratic zero sum dynamic games (see [7]).

The method of Markov chain approximation (MCA) was developed in [8] as a numerical method for solving stochastic control problems and can be considered as an alternative to standard functional approximation techniques. This method has been extended to stochastic differential games and convergence of the value to underlying stochastic process has been proved in [9] for zero sum games and in [10] for non-zero sum games. MCA has been applied for study of cooperative Nash equilibria in fishery games in [11]. In this work, we have considered the model based setting and define cost functions which result in a non-zero sum differential game. We approximate the solution of the aforementioned game using MCA and Minimax strategies. The reason behind using Minimax strategies for a non zero sum game is to ensure distributed computation of control strategies with private cost matrices, as the pumping stations in fig. B.1 are located geographically far away. These methods can also be extended to the model free RL setting as described in [12]. The key contributions of this work are summarized as follows:

- MCA is applied on a SDE model of a WDN in the setting of a non-zero sum differential game.
- A distributed control algorithm based on Shapley's algorithm and MCA is simulated on the WDN.

The rest of the paper is organized as follows. In the next Section, we describe the WDN as an SDE. In Section 3, we introduce MCA method for stochastic control. In Section 4, we introduce stochastic games and present an algorithm combined with MCA introduced in Section 3. In Section 5 we present the simulation results. Lastly, we present our conclusions and future work.

2 WDN represented as an SDE

A general model of the WDN with $\mathcal{N}$ controllers is defined by state equations of the following type

$$dx = \left(\mathrm{f}(x) + \sum_{k=1}^{\mathcal{N}} \mathrm{g}^k(x) u^k \right) dt + \sigma(x) d\mathrm{w}, \tag{B.1}$$

where $x \in X \subseteq \mathbb{R}^n$ is a stochastic process which represents the free flows in the WDN, $u \in U \subseteq \mathbb{R}^p$ represents the pressure control action due to pumps, $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^p$ represent the drift, $\sigma : \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^l$ represents the diffusion with $a(\cdot) = \sigma(\cdot)\sigma(\cdot)^T$ being the corresponding diffusion matrix, w represents the standard Wiener process on $\mathbb{R}^l$. For simplicity, we shall consider the WDN network shown in fig. B.1 with 2 controllers (2 pumping stations) although, the methods developed in this paper are extendable to complex controlled systems with more control inputs. The

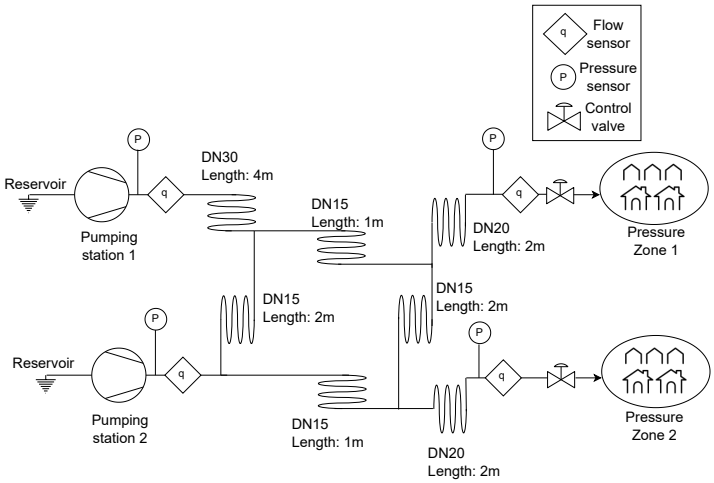


Fig. B.1: Process and Instrumentation diagram

aim of the pumping stations 1 and 2 in fig. B.1 is to minimize the pressure variation at pressure zones 1 and 2. The WDN is modeled by a directed graph $\Gamma = \{N, E\}$ in fig. B.2 where $N = \{n_1, \cdots, n_{11}\}$ represents the nodes or vertices and $E = \{e_1, \cdots, e_{11}\}$ represents the edges where components of WDN such as pipes, pumps and valves are connected. The pumps in fig. B.1 are represented as edges e_{10} and e_{11} in fig. B.2 and the demands are represented by edges e_8 and e_9. The model is defined with following assumptions.

Assumption 2.1 The graph Γ is a connected graph.

Assumption 2.2 We require existence and uniqueness of solutions of (B.1). Let Ω represent the sample space, $\mathcal{F}$ represent the event space and P represent the probability function. Strong existence holds if for a given probability space $(\Omega, \mathcal{F}, P)$, a filtration $\mathcal{F}_t$, an $\mathcal{F}_t$-Wiener process w and an $\mathcal{F}_0$- measurable initial condition $x(0)$,

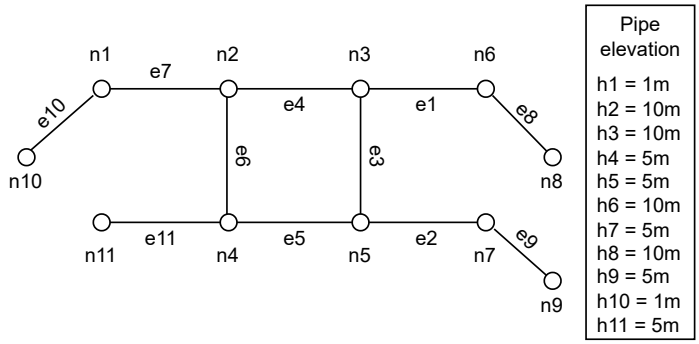


Fig. B.2: Graph of the WDN in fig. B.1 with pipe elevation $\bar{z}$.

there exists an $\mathcal{F}_t$-adapted process $x(t)$ satisfying (B.1) for all $t \geq 0$. Furthermore, uniqueness holds if for any two sample paths $x_1(t), x_2(t)$, $P\{x_1(0) = x_2(0)\} = 1 \implies P\{x_1(t) = x_2(t) \forall t \geq 0\} = 1$.

The model of WDN is derived in [13] and due to space constraints we refer the reader to the same. The stochastic nature of water consumption at nodes n_8 and n_9 in fig. B.2 is captured as a Wiener process.

Assumption 2.3 The diagonal terms of diffusion matrix are dominant over off-diagonal terms as follows

$$a_{ii}(x) - \sum_{j:j\neq i} |a_{ij}(x)| \geq 0. \tag{B.2}$$

This assumption is valid for WDN as in practice only the free flows at end-user edges (e_8 and e_9 in fig. B.2) are correlated with each other. The correlation between free-flows in the other edges are accounted for in the drift term (B.1). Therefore, the off-diagonal terms are negligible relative to diagonal terms and can be ignored. Due to uncertain water demands and correspondingly stochastic nature of free flows in (B.1), the pressure measurement is represented via an Ito integral (reviewed in [8]) with h representing the stochastic integrand. We assume that h is a right-continuous, adapted and a locally bounded process. We approximate the Ito integral using a mesh m with

grid size $\mathcal{Y}_m \to 0$ as,

$$y = \int_s^t \mathrm{h} dx = \lim_{m\to\infty} \sum_{[t_{i-1},t]\in\mathcal{Y}_m} \mathrm{h}_{t_{i-1}}(x_{t_i} - x_{t_{i-1}}). \tag{B.3}$$

We shall now define the control objectives of both the players as the following stage cost functions for a player k,

$$c^k(y, x, u) = y^T W_1 y + W_2 \left| x u^k \right|, \tag{B.4}$$

where $|\cdot|$ represents the standard 1-norm, W_1 and W_2 are normalized weights. The first term in (B.4) represents the pressure variance at consumer nodes and the second term represents the energy consumption (kW) for a pumping station k.

3 Markov chain approximation

In [8], the authors have presented a numerical method referred to as Markov chain approximation (MCA) for solving stochastic control problems. MCA finds the reachable states under a given control policy and calculates transition probabilities via application of finite differences method on the Hamilton-Jacobi equation. At any time s, the number of reachable states depends on the time step ahead (i.e. interpolation time) and MCA method automatically calculates the interpolation time (see fig. B.3 for an illustration of MCA). If the dynamics and/or diffusion are large in magnitude at a given time, the interpolation time correspondingly becomes smaller. The calculated probabilities are a function of the dynamics alone and therefore this method can be used to represent (B.1) as transition probabilities. We define $u^{-k}(t) := [u^1(t), \cdots, u^{k-1}(t), u^{k+1}(t), \cdots u^m(t)]$ as the control vector $u(t)$ without the k^{th} players component or the control vector can be partitioned as $u(t) = [u^k(t), u^{-k}(t)]$. We consider $\mathcal{B} \subset \mathbb{R}^n$ as a compact state space with absorption on boundary. The discounted stochastic control problem for a player k with $\tau = \inf\{t : x(t) \notin \mathcal{B}\}$ is the minimum time taken to exit $\mathcal{B}$ can now be defined as follows.

$$\inf_{u^k(t)x(s),u(\cdot)} \mathbb{E}\left(\int_s^\tau \gamma^t c^k\left(x(t), u^k(t), u^{-k}(t)\right) dt + \gamma^\tau c_\tau^k\left(x(\tau)\right) \right) \tag{B.5a}$$

$$\text{s.t. } dx = \left(\mathrm{f}(x(t)) + \sum_{k=1}^{\mathcal{N}} \mathrm{g}^k(x(t)) u^k(t) \right) dt + \sigma d\mathrm{w}, \tag{B.5b}$$

$$u^k(t) \in U, \quad x(t) \in \mathcal{B}, \tag{B.5c}$$

where $x(s)$ is the state at some starting time s, u is a vector with all $\mathcal{N}$ players control actions, c^k is running cost for player k, c_τ^k is the terminal cost for player k and γ is

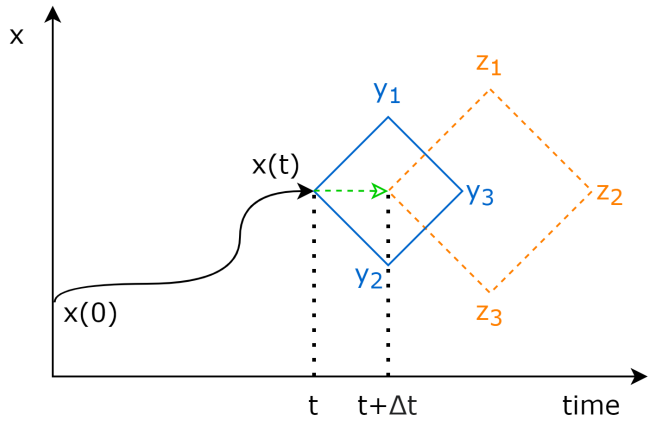


Fig. B.3: Here we illustrate MCA by a sample path of (B.1) starting at $x(0)$ and reaching $x(t)$ at time t, y_1, y_2 and y_3 represent the reachable states for $x(t)$ in time interval Δt. Any realized state $x(t+\Delta t)$ (shown by the green arrow) can be found as a convex combination of extremities of blue polygon. Once the next state $x(t+\Delta t)$ is reached the process repeats as shown by orange polygon with reachable states z_1, z_2 and z_3.

the discount factor. We have assumed that the control space U is the same for all the players for simplicity although the methods discussed in this work are easily applicable to problems where control spaces are different. Let the stochastic integral in the objective function of (B.5) be denoted by

$$Q_s^k(x, u^k, u^{-k}) = \mathop{\mathbb{E}}_{x(s),u(\cdot)}\left(\int_s^\tau \gamma^t c^k \left(x(t), u^k(t), u^{-k}(t)\right) dt + \gamma^\tau c_\tau^k \left(x(\tau)\right) \right), \tag{B.6}$$

where subscript s indicates dependence on s. In the sequel, the notation $-k$ shall denote all the players except player k. The optimal value can than be obtained by minimizing Q_s^k with respect to u^k for each state as follows

$$V_s^k(x) = \inf_{u^k(s)} Q_s^k(x(s), u^k(s), u^{-k}(s)). \tag{B.7}$$

Note that the above formulation gives rise to a non-zero sum differential game. Since the value of a state for any player (B.7) depends on the joint action u of the all the players, we need to define the solution in the sense of strategies π of all the players. Let each player choose control action u^k from probability distribution π^k. The approximate Minimax solution for (B.5) can be defined as the mixed strategy $\pi^{k\star}$ such that,

$$Q_s^k(x, \pi^{k\star}, \pi^{-k\star}) < Q_s^k(x, \pi^k, \pi^{-k\star}) + \varepsilon, \tag{B.8}$$

where ϵ indicates the amount of tolerable sub-optimality to the exact Minimax solution. For discretizing (B.5), we divide the compact state space $\mathcal{B}$ into v number of discrete states. Consequently, we can define the state space approximation parameter h (which represents the coarseness of the state space grid) as follows

$$h = \frac{1}{v-1}. \tag{B.9}$$

Let $\xi(s) \in \mathbb{R}^v$ denote the discrete state centered at $x(s)$ with a grid box of dimension h at some time s, than the next reachable discrete state in the set $\mathcal{B}$ is denoted by $\xi(s+1) = x(s) + e_i h$, where $e_i \in \mathbb{R}^n$ is the basis vector in i^{th} direction of the state space (see fig. B.4). Note that we are considering only state transitions of the type $x \pm e_i h$

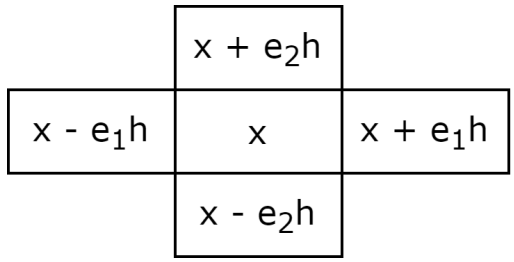


Fig. B.4: Illustration of state space as a grid. Any state whose numerical value is within the boundary of a particular grid box is considered as a part of the same.

and not of the type $x \pm e_i h \pm e_j h$ since the interpolation time is small (depending on the drift and diffusion as we shall see later) and due to Assumption 2.3. These additional terms can be considered at the cost of a greater computation time. We shall now apply the well-known Ito's lemma in order to obtain evolution of $V_s^k(x)$ with time. In [10], convergence of Markov chain approximation for non-zero sum differential games is proved based on certain key assumptions.

Assumption 3.1 ([10]) The drift of (B.1) is additively separable into two components which represents the contributions to the drift due to individual players. Furthermore, the cost c^k for each player is also additively separable into two components which represents the contributions to the cost due to the individual players.

Assumption 3.1 is satisfied for both the considered system (B.1) and the cost function (B.4) and can be verified by dividing the drift in (B.1) as

$$b^k(x,u) = \frac{\mathrm{f}(x)}{\mathcal{N}} + \mathrm{g}^k(x)u^k. \tag{B.10}$$

Similarly (B.4) can be split among the players. Let $a_{ij}(x)$, $i,j = 1,\cdots,n$ be an element of the diffusion matrix $a(x)$ and let $b_i(x,u)$ represent the drift term $\mathrm{f}_i(x) + \sum_{k=1}^{\mathcal{N}} \mathrm{g}^k(x)u^k$, than we can define $\mathcal{L}$ for (B.1) as follows.

$$\mathcal{L} = \frac{1}{2}\sum_{i,j=1}^{n} a_{ij}(x(t))\frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^{n} b_i(x(t),u(t))\frac{\partial}{\partial x_i}. \tag{B.11}$$

Applying Ito's differential operator on (B.5) gives us the stochastic analogue of the well-known Hamilton-Jacobi equations (see [8]) for a player k as follows.

$$\begin{aligned} &\mathcal{L}V^k(x(t)) + \gamma^t c^k(x(t),u(t)) = 0, \\ &V_s^k(x(\tau)) = c^k(x(\tau)), V_s^k(x(s)) = c^k(x(s)), \end{aligned} \tag{B.12}$$

where the last two equations represent the boundary conditions. Assume that each player acts simultaneously at a given time instant to ensure the well-posedness of solutions of (B.12). Expanding the differential operator in (B.12) gives,

$$\frac{1}{2}\sum_{i,j=1}^{n} a_{ij}(x)\frac{\partial^2 V_s^k(x)}{\partial x_i \partial x_j} + \sum_{i=1}^{n} b_i(x,u)\frac{\partial V_s^k(x)}{\partial x_i} + \gamma^t c^k(x,u) = 0, \tag{B.13}$$

where we have written $x(t), u(t)$ as x, u for notational convenience. We now apply finite-differences with space approximation parameter h on (B.13) as follows.

$$\begin{aligned} \sum_{i,j=1}^{n} a_{ij}(x)\frac{V_s^k(x+e_ih) + V_s^k(x-e_ih) - 2V_s^k(x)}{2h^2} + \sum_{i=1}^{n} |b_i(x,u)|^+ \frac{V_s^k(x+e_ih) - V_s^k(x)}{h} \\ - \sum_{i=1}^{n} |b_i(x,u)|^- \frac{V_s^k(x) - V_s^k(x-e_ih)}{h} + \gamma^t c^k(x,u) = 0, \end{aligned} \tag{B.14}$$

where $|b_i(x,u)|^+ = \max(b_i(x,u),0)$ and $|b_i(x,u)|^- = \max(-b_i(x,u),0)$. This is the standard "upwind" scheme in numerical analysis of hyperbolic partial differential equations (see [14]). The intuition behind this scheme is that the approximation of V_s^k should be in the same direction as the drift of (B.1). It can also be verified that $|b_i(x,u)|^+ + |b_i(x,u)|^- = |b_i(x,u)|$. In order to make the probability transitions independent of joint control action u, we calculate the joint control action u_{max} which gives maximal drift and replace $|b_i(x,u)|$ with $|b_i(x,u_{max})|$. We can rearrange the terms in (B.14) so as to obtain elements of probability transition matrix $p(x, x \pm e_ih \mid u)$ and

interpolation time Δt as follows.

$$V_s^k(x) = \underbrace{\frac{a_{ii}(x)/2 + h\,|b_i(x,u)|^+}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x,u_{max})|)}}_{p(x,x+e_ih|u)} V_s^k(x+e_ih) + \underbrace{\frac{a_{ii}(x)/2 + h\,|b_i(x,u)|^-}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x,u_{max})|)}}_{p(x,x-e_ih|u)} V_s^k(x-e_ih) + \underbrace{\frac{h^2}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x,u_{max})|)}}_{\Delta t} \gamma^t c^k(x,u). \quad \text{(B.15)}$$

The probability of state remaining unchanged is given by

$$p(x, x \mid u) = 1 - (p(x, x + e_ih \mid u) + p(x, x - e_ih \mid u)). \quad \text{(B.16)}$$

It should be noted that we only get the transition probabilities required for the next state transition and therefore, the probability transition matrix constructed using this method will be sub-stochastic (i.e. sum of probabilities will be less than 1). Thus, the differential game (B.5) can be approximated by a Stochastic Game in the sense of [3].

4 Solving Stochastic Games

We are proposing a distributed Stochastic Game solver based on Shapley's algorithm which can be solved by each of the players without any knowledge of the cost incurred by other player. Stochastic games are a generalization of static games where the decisions taken by a player influences both the immediate costs and also the states reached in the future. Formally a stochastic game $\mathcal{G}$ can be defined as a tuple $\mathcal{G} = \{\mathcal{N}, \mathcal{X}, (U^1 \times \cdots U^{\mathcal{N}}), \mathcal{P}, (C^1, \cdots, C^{\mathcal{N}})\}$, where $\mathcal{N}$ is the no. of players, $\mathcal{X}$ is the finite state space of the game with cardinality $\mid \mathcal{X} \mid = \nu$, U is the finite control space of all the players, $\mathcal{P}$ is the probability transition matrix, $C^1, \cdots, C^{\mathcal{N}}$ are the cost matrices for each player corresponding to U. We begin by discretizing the control space U into finite control actions u. For the considered WDN, these represent the operating power of pumping stations (for ex. $u^1 = 1$ implies that the pumping station 1 is operating at 10% of its maximum capacity).

Assumption 4.1 A player only knows the possible finite control actions that can be taken by the other players.

Let C^k denote the cost matrix of dimensions $\nu\times \mid u^k \mid \times \mid u^{-k} \mid$ for a player k at a discrete state $\xi(s)$, where $\mid(\cdot)\mid$ is the cardinality of discrete set $(\cdot)$. Every non-zero sum game $\mathcal{G} = \{\mathcal{N}, \mathcal{X}, U, \mathcal{P}, (C^1, \cdots, C^{\mathcal{N}}), \gamma\}$ can be solved using Minimax strategies if each player k solves the corresponding zero-sum game $\mathcal{G}' = \{\mathcal{N}, \mathcal{X}, U, \mathcal{P}, (C^k, -C^k), \gamma\}$ to obtain their worst-case costs. Such solutions are referred to as mixed security strategies in [15]. In the sequel $\xi = \xi(s) \in \mathcal{X}$ shall denote a finite state of the game $\mathcal{G}$ at time s and the operator $val[\cdot]$ denotes the value V of a matrix game as per Von Neumann's Minimax theorem which states that for any real $n_1 \times n_2$ matrix A with elements a_{ij}, there exists a pair of probability vectors $\pi^{\mathbf{1}\star} = (\pi_1^{1\star}, \cdots, \pi_{n_1}^{1\star})$ and $\pi^{\mathbf{2}\star} = (\pi_1^{2\star}, \cdots, \pi_{n_2}^{2\star})$ such that

$$\sum_i a_{ij}\pi_i^{1\star} \leq V \leq \sum_j a_{ij}\pi_i^{2\star} \quad \forall i, j. \tag{B.17}$$

Let $u = (u^1, \cdots, u^{\mathcal{N}})$ be the joint control actions and v be the number of discrete states. Since we will be approximating the problem (B.5) via (B.15), the instantaneous costs for the associated Stochastic game for the k^{th} player will be defined as follows,

$$c_s^k(\xi, u) := \underbrace{\frac{h^2}{\sum_{i=1}^{n}(a_{ii}(x) + h\,|b_i(x, u_{max})|)}}_{\Delta t}\gamma^t c^k(x, u). \tag{B.18}$$

A Stochastic game encodes the transition probabilities in the cost matrix by defining Shapley game matrix as follows.

$$M_s^k(\xi, u) = c_s^k(\xi, u) + \sum_{\chi=1}^{v} p_s(\chi|\xi, u)V_s^k(\chi), \tag{B.19}$$

where $c^k(\xi, u) \in C^k$ is the cost at state ξ, γ is the discount factor, χ represents the reachable states from ξ, $p_s(\chi|\xi, u)$ is the transition probability of reaching χ given ξ and $V_s^k(\chi)$ is the value of state χ. For a given state ξ, the mixed security strategy $\pi^k(\xi)$ for

the game $\mathcal{G}$ can be computed using the following linear program.

$$\min_{\pi^k, V_s^k(\xi)} \quad V_s^k(\xi) \tag{B.20a}$$

$$\text{s.t.} \quad \sum_{u^k \in U} M_s^k(\xi, u)\pi^k(u^k) \leq V_s^k(\xi), \quad \forall u^{-k} \in U, \tag{B.20b}$$

$$\sum_{u^k \in U} \pi^k(u^k) = 1, \tag{B.20c}$$

$$\pi^k \geq 0, \quad \forall u^k \in U, \tag{B.20d}$$

where constraint (B.20b) ensures that player k chooses mixed strategy π^k such that for every joint pure strategy profile $u^{-k} \in U$ of other players, player k's expected value for state ξ is at most $V_s^k(\xi)$ (which is being minimized in (B.20a)) and remaining constraints ensure that mixed strategy π^k obeys axioms of probability. Note that the Linear program is only valid for the given starting state ξ. Thus, as the state changes, we need to solve a sequence of Linear programs corresponding to each ξ. We now state the MCA based algorithm for solving stochastic games in Algorithm 10.

Algorithm 10 MCA Stochastic differential game solver

Input: Initial state $x(s)$, State space Grid $\Delta^h x$, drift b, diffusion matrix a, control space U, terminal time T.
Initialize $V_s^k = 0$ for all states in the grid
while $s < T$ **do**
    $\xi \leftarrow x(s)$
    Find grid square $\Delta^h x(s)$ corresponding to ξ
    Construct cost matrix C^k, $\forall u^1, \cdots, u^{\mathcal{N}}$
    Let χ denote possible reachable states from ξ
    Obtain transition probabilities using (B.15) and (B.16)
    for All possible control actions of players $-k$ **do**

$$V_{s+\Delta s}^k(\xi) \leftarrow val\Bigg[\underbrace{c_s^k(\xi, u) + \sum_{\chi=1}^{v} p_s(\chi|\xi, u)V_s^k(\chi)}_{M_s^k}\Bigg] \tag{B.21}$$

    end for
    Solve the Shapley game (B.21) using (B.20) for $\pi^k(\xi)$
    Sample u^k from $\pi^k(\xi)$ and apply on (B.1) at time s
    $s \leftarrow s + \Delta t$
end while

5 Simulation Results and Discussions

We ran a variety of simulations with different initial conditions on (B.1). The simulation results presented in this section are with initial conditions $x = [-50\ 60\ 50\ 0]^T$ and with $v = 20$ discrete states. In fig. B.5, we can observe that the states representing the

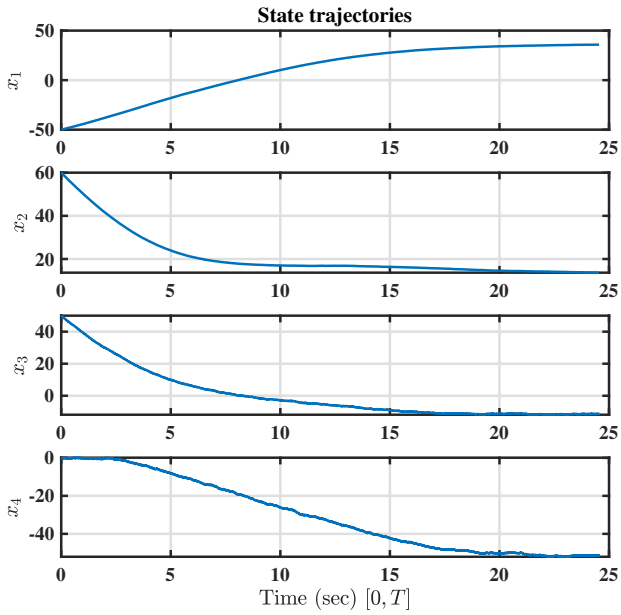


Fig. B.5: State trajectory for initial $x = [-50\ 60\ 50\ 0]^T$

free flows in the network reach a steady state and the system dynamics are stable. In fig. B.6, we can observe that the optimal values for both the players reach a fixed point of the Shapley equation (B.21). The value functions of both the players decrease to their minimum as both the players apply control actions. Player 1 applies higher control input relative to Player 2 and consequently Player 1's value is higher than Player 2. This situation can be changed by modifying the weights W_1 and W_2 in (B.4). The accuracy of Algorithm 10 is dependent on the coarseness of state grid which is determined by the considered number of discrete states v. In table B.1, we show how the numerical accuracy improves with higher v.

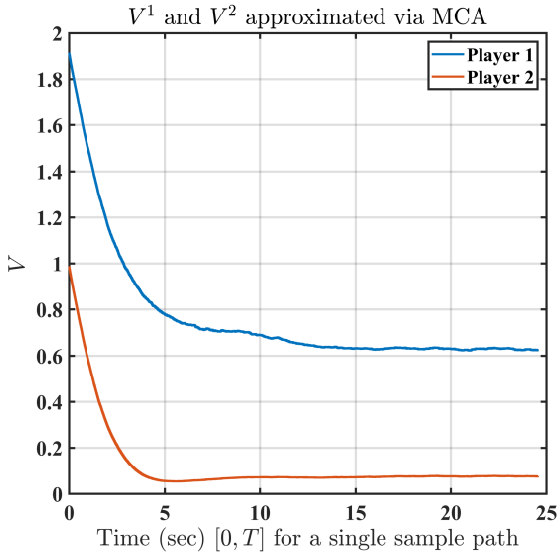


Fig. B.6: The values obtained for both the players converge to the Minimax solution. The state space discretization was $v = 20$.

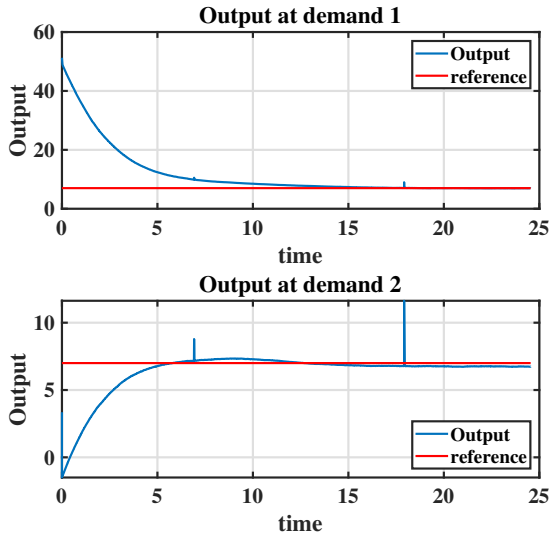


Fig. B.7: Output pressure is tracked successfully to the reference.

No. of discrete states	V^1 obtained from MCA	V^2 obtained from MCA
$v = 5$	13.2525	1.7103
$v = 10$	7.8972	0.9951
$v = 15$	1.5339	0.1712
$v = 20$	0.6549	0.0732

Table B.1: Value function approximations obtained at terminal time T for different v.

It is proved in [9] and [10] that the Minimax solution of original continuous time problem (B.5) can be approximated upto an arbitrary $\epsilon > 0$ by making the grid parameter h small (which increases simulation time in practice) and this is shown in the table B.1. In fig. B.7, we can observe that the output y successfully tracks the reference pressure. The spikes in fig. B.7 are due to the controllers periodically switching off as there is a cost associated with operation in (B.4). A Minimax solution may or may not always converge to an Nash equilibrium solution unless the non-zero sum differential game is strategically equivalent to a zero-sum differential game (see [16]). However, in this case, simulation studies show that the Minimax solution does correspond to the Nash equilibrium solution.

6 Conclusion

We have designed a control strategy for (B.5) using MCA and Stochastic games. We have obtained approximate Minimax strategies and simulation results show that steady state is reached. However, better solutions can be obtained if the players have access to the cost matrices or control actions of players at previous iteration. Given the challenging nature of non-zero sum differential games and stochastic games in general (see [17] and [5]), we do obtain a reasonable solution by using Minimax strategies and MCA. The Minimax solution can also be extended to $\mathcal{N} > 2$ players in a straight forward as each player only needs to know their own cost matrix. However, the algorithm scales exponentially as the number of players is increased. Besides scalability in future, we would also like to consider model-free learning of Nash equilibrium in this setting.

Acknowledgements

Financial support from the Poul Due Jensen Foundation (Grundfos Foundation) for this research is gratefully acknowledged.

References

[1] M. Tahavori, C. S. Kallesøe, J. Leth, and R. Wisniewski, "Modeling of water supply systems: Circuit theoretic approach," in *2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2012, pp. 1561–1566.

[2] D. P. Bertsekas, *Abstract dynamic programming*. Athena Scientific, 2022.

[3] L. S. Shapley, "Stochastic games," *Proceedings of the national academy of sciences*, vol. 39, no. 10, pp. 1095–1100, 1953.

[4] T. Raghavan and J. A. Filar, "Algorithms for stochastic games—a survey," *Zeitschrift für Operations Research*, vol. 35, no. 6, pp. 437–472, 1991.

[5] J. Filar and K. Vrieze, *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

[6] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[7] E. Mazumdar, L. J. Ratliff, and S. S. Sastry, "On gradient-based learning in continuous games," *SIAM Journal on Mathematics of Data Science*, vol. 2, no. 1, pp. 103–131, 2020.

[8] H. J. Kushner and P. G. Dupuis, *Numerical methods for stochastic control problems in continuous time*. Springer Science & Business Media, 2001, vol. 24.

[9] H. J. Kushner, "Numerical approximations for stochastic differential games," *SIAM journal on control and optimization*, vol. 41, no. 2, pp. 457–486, 2002.

[10] ——, "Numerical approximations for nonzero-sum stochastic differential games," *SIAM journal on control and optimization*, vol. 46, no. 6, pp. 1942–1971, 2007.

[11] A. Haurie, J. B. Krawczyk, and M. Roche, "Monitoring cooperative equilibria in a stochastic differential game," *Journal of optimization Theory and Applications*, vol. 81, no. 1, pp. 73–95, 1994.

[12] R. Munos and P. Bourgine, "Reinforcement learning for continuous stochastic control problems," in *NIPS*, 1997, pp. 1029–1035.

[13] R. Misra, R. Wisniewski, and C. S. Kallesøe, "Approximating the model of a water distribution network as a markov decision process," *IFAC-PapersOnLine*, vol. 55, no. 20, pp. 271–276, 2022.

[14] R. J. LeVeque, *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.

[15] T. Alpcan and T. Başar, *Network security: A decision and game-theoretic approach.* Cambridge University Press, 2010.

[16] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory.* SIAM, 1998.

[17] A. Bressan, "Noncooperative differential games. a tutorial," *Milan Journal of Mathematics*, vol. 79, no. 2, pp. 357–427, 2011.

Paper C

On principle of optimality for safety-constrained Markov Decision Process and p-Safe Reinforcement Learning

Rahul Misra, Rafał Wisniewski

The paper has been accepted for publication as a part of the conference proceedings for *26th International Symposium on Mathematical Theory of Networks and Systems (MTNS), 19-23 August 2024*

The layout has been revised.

Abstract

We study optimality for the safety-constrained Markov decision process which is the underlying framework for safe reinforcement learning. Specifically, we consider an undiscounted safety-constrained Markov decision process subject to random stopping times. The decision maker's goal is to reach a goal state while avoiding unsafe states with certain probabilistic guarantees. Therefore the underlying Markov chain for any control policy will be Multichain or non-ergodic since by definition there exists a goal set and an unsafe set. Bellman's principle of optimality does not hold for such a safety-constrained Markov decision process in a Multichain setting as highlighted by a counterexample. We resolve the aforementioned counterexample by considering a zero-sum game setting between the policy and the Lagrange multiplier vector. Under suitable assumptions regarding the existence of admissible policy, we propose an off-policy RL algorithm for learning an optimal policy that satisfies the probabilistic safety guarantees. After that, we present the finite time error bound of the proposed RL algorithm. Lastly, we present simulation results of the aforementioned RL algorithm on a robot in a grid world setting.

Keywords: Reinforcement Learning, Markov Decision Process, Zero-Sum Game, Reach-Avoid problem, p-Safety, Hitting time.

1 Introduction

In recent years, due to the success of Reinforcement learning (RL) techniques, there has been a renewed interest in the study of Markov decision processes (MDP), the underlying mathematical framework for RL. Since several applications of RL are safety-critical (for example. robotics [1], autonomous cars [2] and healthcare [3]), recent research on RL is focused on safety-constrained RL. In this paper, we consider RL problems with probabilistic safety guarantees where the objective of the decision maker (or the controller) is to steer the system trajectory to reach a certain goal state while avoiding unsafe states with a certain probabilistic guarantee. Such problems are referred to as *Reach-Avoid* problems in literature such as [4]. The Reach-Avoid problem is formulated as a p-Safe MDP (p being the probabilistic safety guarantee) with a known transition probability matrix and known rewards) in the paper [5] with random stopping time. This paper extends the work presented in [5] to the RL setting with an unknown transition probability matrix and unknown rewards.

Constrained MDP problems have been studied to model multi-objective optimization problems in [6] and safety-constrained MDPs in [7]. Typically such problems are solved by a Linear program that considers the state-action occupation measure as the decision variable with the optimal randomized policy being a normalized ratio of the optimal state-action frequencies [8]. An important distinction made in the study of MDPs is

whether the underlying Markov chain for all policies is unichain (also referred to as ergodic Markov chain) or multichain (also referred to as non-ergodic Markov chain) [8], [9]. The celebrated Bellman's principle of optimality does not hold in general for constrained MDPs with underlying multichain structure as shown in the counterexample due to [10]. Attempts have been made to resolve this counterexample for average-reward MDPs in [11] and for finite time-horizon MDPs in [12]. Reference [11] proposes changing constraints as new states are visited (i.e. constraint changes depending on previous outcomes). This is not entirely satisfactory in our case as safety guarantees should not be changed based on previous outcomes. Reference [12] proposes dynamic programming formulation with time consistency for risk-constrained stochastic optimal control problems wherein they introduce an additional dynamic programming operator that ensures the selection of only feasible policies. This is a similar approach as in [13] and [14] (considers non-randomized policies). The aforementioned approaches of [12], [13], and [14] can be computationally prohibitive (depending on the considered constrained MDP) since they require the computation of a feasible set at each time step in addition to the standard dynamic programming operator. Therefore, this approach is computationally more expensive with an increasing number of control actions, states, and constraints compared to the standard dynamic programming.

Despite the aforementioned counterexample, there are many works on RL for constrained MDPs such as the model-based approach of [15] and [16], or the actor-critic algorithm proposed in [17], the natural policy gradient primal-dual proposed in [18] and the Lagrange multiplier based approach for constrained MDPs also considered in [16]. This is because it is a common assumption that the underlying Markov chain has a unichain structure and the counterexample is for an MDP with an underlying multichain structure. It is important to note that despite the recent results on safe RL, almost all of the works mention an assumption regarding the ergodicity of the underlying Markov chain under any policy (which is the same as the unichain assumption). In this work, we consider the safe RL problem as a constrained multichain MDP problem and present algorithms for the same. The primary contributions are summarized as follows.

- The counterexample due to [10] is studied in the context of p-Safe MDP and resolved.
- An off-policy modified Q-learning algorithm is proposed for learning optimal policies for p-Safe MDP.
- The aforementioned algorithm is simulated on a robot in a grid world setting.

The rest of the paper is organized as follows: Section 2 summarizes some common notation used throughout this paper. Section 3 formally introduces the problem formulation. Bellman's principle of Optimality is formally stated and the counterexample due to [10] is introduced in Section 4. The p-Safe Lagrange iteration Algorithm is proposed

in Section 5. In Section 6, we consider the problem of RL for constrained MDPs and propose an off-policy algorithm for learning the Lagrangian up to some error bounds, and simulation results on a robot are presented in Section 7. Finally, the conclusion and future work is presented in Section 8.

2 Notation

- For $n \in \mathbb{N}$, $\mathbb{1}_n$ denotes a vector with all elements 1.
- $\mathbb{E}_\mu$ represents the expectation operator corresponding to a probability measure μ.
- $\Delta(A)$ represents a probability simplex in $\mathbb{R}^{|A|}$ where A is a finite set.
- Boldface denotes a vector (ex. $\mathbf{V}$ is the value vector with components $V(x)$ corresponding to each state).
- Superscript $*$ indicates optimal value.

3 Problem Formulation

We consider the reach-avoid problem for Markov decision processes where the goal of the decision maker is to pick control actions such that the state reaches a goal set $\mathcal{G}$ while simultaneously avoiding some unsafe set $\mathcal{O}$ (with a certain probabilistic guarantee). The problem is formulated as a constrained Markov decision process with random hitting time representing the time taken to hit $\mathcal{O}$ and an exit time. Consider a stochastic process (X_t) defined on the state space $\mathcal{X}$. The state space is partitioned into three sets: a set of transient states E, the goal set $\mathcal{G}$, and an unsafe set $\mathcal{O}$ (see fig. C.1). The process X_t

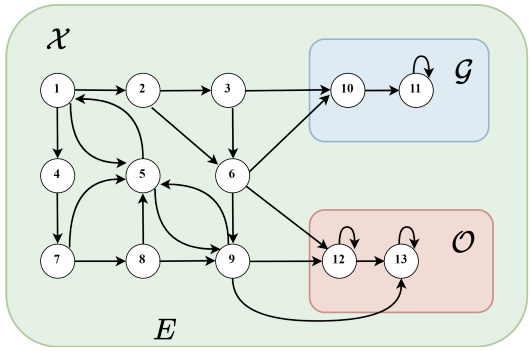


Fig. C.1: A generic Reach-Avoid problem for MDP with states numbered $1, \cdots, 13$. $\mathcal{X}$ partitioned into the goal set $\mathcal{G}$, unsafe set $\mathcal{O}$ and transient states E.

starts in the set of non-absorbing states denoted by E (referred to as transient states as X_t leaves E in finite time). Since $\mathcal{G}$ and $\mathcal{O}$ consist of absorbing states, we have a Multichain MDP which is defined next.

Definition 3.1 (Multichain MDP) An MDP is said to be multichain if there exists a policy π such that the corresponding Markov chain consists of at least two ergodic chains and a (possibly empty) set of transient states.

The constrained MDP considered in this work is a multichain MDP as it consists of at least two recurrent chains: goal set $\mathcal{G}$ and unsafe set $\mathcal{O}$. Let U be a set of a finite number of actions. Define $c : \mathcal{X} \times U \to \mathbb{R}$ as the immediate cost and $k : \mathcal{X} \times U \to [0, 1]$ as the immediate probability of hitting $\mathcal{O}$. Let $c_t := c(X_t, U_t)$, and $k_t := k(X_t, U_t)$, where U_t is a stochastic process defined on the set of actions U which represents the control action taken at time t. We consider state-feedback memoryless policies (i.e. Markov policies) that are defined next.

Definition 3.2 (Markov policies π) A Markov policy $\pi : E \to \Delta(U)$ is a stationary (i.e. independent of time) control policy that is dependent only on the given state as

$$\pi(x) = [\pi(u_1 \mid x), \cdots, \pi(u_{|U|} \mid x)] \in \Delta(U). \tag{C.1}$$

In this work, we restrict our attention to X_t as it evolves on transient state space E. This is a natural way to study the Reach-Avoid problem as the study of X_t is no longer interesting once it reaches either the goal set or the unsafe set. Therefore, the sets $\mathcal{G}$ and $\mathcal{O}$ consist only of absorbing states where $c_t = 0$ and $k_t = 0$. Since X_t stops once it exits E, we introduce the hitting time $\tau_{\mathcal{O}}$ and exit time τ.

Definition 3.3 (First Hitting time of $\mathcal{O}$) The first hitting time of $\mathcal{O}$, $\tau_{\mathcal{O}}$ is the time taken for $X_{\tau_{\mathcal{O}}} \in \mathcal{O}$ given $X_0 \in E$,

$$\tau_{\mathcal{O}} := \left\{ \inf_t X_t \in \mathcal{O} \mid X_0 \in E \right\}. \tag{C.2}$$

Definition 3.4 (Exit time) The Exit time τ represents the time taken for $X_t \notin E$ given $X_0 \in E$ i.e.,

$$\tau := \left\{ \inf_t X_t \in \mathcal{G} \cup \mathcal{O} \mid X_0 \in E \right\}. \tag{C.3}$$

The value of state i is denoted by $V_\pi(i)$ that represents the expected costs accumulated before the exit of X_t from the set E given that $X_0 = i$ and policy is π. We define a Safety function $S_\pi(i)$ which represents the expected probability of hitting the unsafe set starting from state i and following policy π. The problem of reaching $\mathcal{G}$ before hitting $\mathcal{O}$ is defined as the following p-Safe MDP problem.

Definition 3.5 (p-Safe MDP) For $0 \leq p \leq 1$, compute

$$\min_\pi V_\pi(i) := \mathbb{E}_\pi \left[\sum_{t=0}^{\tau} c_t | x_0 = i \right], \ i \in E, \tag{C.4a}$$

$$\text{subject to } S_\pi(i) := \mathbb{E}_\pi \left[\sum_{t=0}^{\tau_\mathcal{O}} k_t | x_0 = i \right] \leq p, \ i \in E, \tag{C.4b}$$

A more detailed explanation of the p-Safe MDP problem can be found in [5]. Note that the p-Safe MDP problem (C.4) differs from the standard formulation of constrained MDP as the stopping time for the objective function and the constraint function are different (τ and $\tau_\mathcal{O}$ respectively).

Remark 3.1. *The exit time is always less than or equal to the first hitting time of the unsafe set i.e.* $\tau \leq \tau_\mathcal{O}$.

The p-Safe MDP is well-posed since the exit time is always less than or equal to the first hitting time of the unsafe set and since the goal states $\mathcal{G}$ are absorbing with zero cost. More specifically, consider X_t and suppose $\tau \leq t \leq \tau_\mathcal{O}$, than by definitions 3.3 and 3.4, $X_t \in \mathcal{G}$. Therefore,

$$\begin{aligned} V_\pi(i) &= \mathbb{E}_\pi \left[\sum_{t=0}^{\tau} c_t + \sum_{t=\tau}^{\tau_\mathcal{O}} c_t | x_0 = i \right], \\ &= \mathbb{E}_\pi \left[\sum_{t=0}^{\tau} c_t | x_0 = i \right], \text{ since } \mathbb{E}_\pi \left[\sum_{t=\tau}^{\tau_\mathcal{O}} c_t | x_0 = i \right] = 0, \end{aligned}$$

as $\mathcal{G}$ consists of only absorbing states with $c_t(x_t, u_t) = 0$ for all $x_t \in \mathcal{G}$. We denote the vector value function and vector safety function as $\mathbf{V}_\pi = (V_\pi(i))_{i \in E} \in \mathbb{R}^N$ and $\mathbf{S}_\pi = (S_\pi(i))_{i \in E} \in \mathbb{R}^N$ respectively for a policy π. For each stationary randomized policy $\pi : E \to \Delta(U)$, let $\tilde{P}(\pi)$ denote the probability transition matrix for the resulting Markov chain. In the sequel, the subscript for P shall denote the start state and superscript shall denote the next state. For example, $P_E^\mathcal{O}$ denotes the probability transitions starting

from states in E to states in $\mathcal{O}$. The transition matrix $\tilde{P}(\pi)$ can be partitioned as

$$\tilde{P}(\pi) = \begin{bmatrix} P(\pi) & P_E^{\mathcal{O}} & P_E^{\mathcal{G}} \\ P_{\mathcal{O}}^E & P_{\mathcal{O}}^{\mathcal{O}} & P_{\mathcal{O}}^{\mathcal{G}} \\ P_{\mathcal{G}}^E & P_{\mathcal{G}}^{\mathcal{O}} & P_{\mathcal{G}}^{\mathcal{G}} \end{bmatrix} = \begin{bmatrix} P(\pi) & P_E^{\mathcal{O}} & P_E^{\mathcal{G}} \\ 0 & \mathbf{1}_{|\mathcal{O}|} & 0 \\ 0 & 0 & \mathbf{1}_{|\mathcal{G}|} \end{bmatrix},$$

where the last equality represents the multichain MDP. Clearly, the matrix $P(\pi)$ will be sub-stochastic (i.e. its entries sum to < 1) and its entries are calculated as

$$P_{ij}(\pi) = \sum_{u \in U} P_{ij}(u)\pi(u \mid i), \quad i, j \in E. \tag{C.5}$$

Let $K_\pi := P_E^{\mathcal{O}}(\pi)\mathbb{1}_{|\mathcal{O}|}$ represent the matrix of probabilities of reaching the set $\mathcal{O}$

$$K_\pi = \sum_{u \in U} \sum_{j \in \mathcal{O}} P_{ij}(u)\pi(u \mid i). \tag{C.6}$$

The immediate safety probability $k_t = k(x_t, u_t)$ in (C.4b) can be obtained from K_π. If it is unknown (i.e. we are using an RL algorithm), it can be calculated by the empirical frequency of visits to the unsafe sets during the training of the RL algorithm.

4 Bellman's Principle of Optimality

In this section, we will state Bellman's principle of optimality for MDPs. This principle does not necessarily hold for constrained MDPs as will be shown later in this section. The principle of optimality for unconstrained MDPs as stated in [19] is *An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.* Mathematically, it can be formulated as follows. Consider the unconstrained optimization problem defined by (C.4a) and (C.1). We introduce the optimal Bellman operator for this problem as follows

$$\mathcal{B}_{\pi^\star}[V_{\pi^\star}](i) = \min_{u \in U} \left[c_t + \sum_{j=1}^{N} P_{ij}(u) V_{\pi^\star}(j) \right], \quad \forall i \in E, \tag{C.7}$$

where $\pi^\star$ is an optimal Markovian policy. Let $\mathcal{B}^i_{\pi^\star}$ denote the optimal Bellman operator if $x_0 = i$ (i.e. starting state is i) and let $\mathcal{B}^j_{\pi^\star}$ denote the optimal Bellman operator if $x_0 = j$ (i.e. starting state is j) than $\pi^\star$ should satisfy the following property

$$\mathcal{B}^i_{\pi^\star}[\mathbf{V}_{\pi^\star}] = \mathcal{B}^j_{\pi^\star}[\mathbf{V}_{\pi^\star}] = \mathbf{V}_{\pi^\star} \quad \text{for any } i, j \in E. \tag{C.8}$$

Furthermore as noted in [19], (C.8) indicates that the problem defined by (C.4a) and (C.1) has an optimal substructure and therefore, we can find the optimal policy $\pi^\star$ by

finding the fixed point of (C.7) for each state $i \in E$. However, the following counterexample due to [10] shows that (C.8) is not true in general for constrained multichain MDPs (and therefore (C.7) does not hold for constrained multichain MDPs).

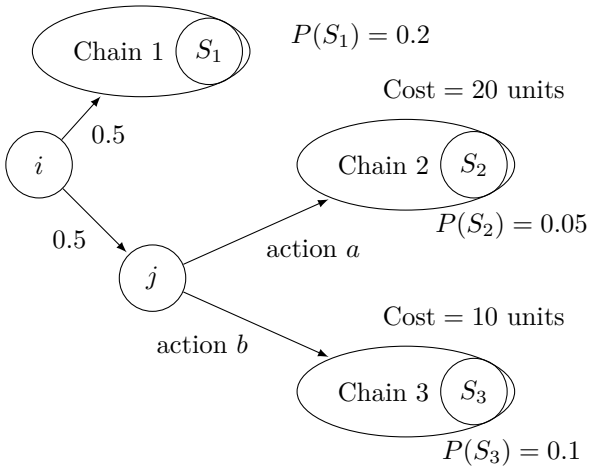


Fig. C.2: The counterexample due to [10]. i and j are transient states. The decision maker can choose either action a or action b resulting in the next state being chain 2 or chain 3 respectively.

4.1 Example (Haviv's Counterexample [10]) Consider a constrained multichain MDP with three recurrent chains $1, 2, 3$, three unsafe sets S_1, S_2, and S_3, and two transient states $i, j \in E$. If the process reaches Chain 1, then the probability of hitting the unsafe set S_1 is $P(S_1) = 0.2$. If the process does not hit S_1 despite being in Chain 1 then we say it has reached the goal set. Similar is the case for Chains 2 and 3 with unsafe sets S_2 and S_3. Note, that the process terminates once it reaches either of the recurrent chains and hits the unsafe sets or it reaches either of the recurrent chains and avoids hitting the unsafe sets. If the process starts at state i, then with 0.5 probability, it will go either to Chain 1 or to state j. The decision maker can only decide whether to choose action a or action b. The goal of the decision-maker is to have the expected probability of visiting $S_1 \cup S_2 \cup S_3 \leq 0.125$. Formally, the problem can be stated in terms of (C.4a) and (C.4b) as follows.

$$\min_{\pi} \; V_\pi(i) := \mathbb{E}_\pi \left[\sum_{t=0}^{\tau} c_t | x_0 = i \right], \tag{C.9a}$$

$$\text{s.t.} \quad S_\pi(i) := \mathbb{E}_\pi \left[\sum_{t=0}^{\tau_\odot} k_t | x_0 = i \right] \leq p, \tag{C.9b}$$

where $p = 0.125$. The safety function calculated from state i is $S(i) = 0.5 \times 0.2 + 0.5 \times 0.05 = 0.125$ for action a and $S(i) = 0.5 \times 0.2 + 0.5 \times 0.1 = 0.15$ for action b. The safety function calculated from state j is $S(j) = 0.05$ for action a and $S(j) = 0.1$ for action b. Therefore, action b is feasible and optimal if we start from state j. However, action b is infeasible and therefore not optimal if we start from state i. Thus (C.8) does not hold in general for constrained multichain MDPs.

This implies that a naive implementation of dynamic programming algorithms (and RL algorithms based on Bellman equations) on constrained multichain MDPs may lead to solutions that are feasible but not optimal (i.e. conservative solutions). In the sequel, we seek to address this issue by using the Zero-sum Stochastic (Markov) Game theory.

5 Proposed Solution

In this section, we propose to find a solution to the problem (C.4a) and (C.4b) that satisfy Bellman's principle of optimality i.e. optimal policy $\pi^\star$ which satisfies (C.8). The constrained MDP defined by (C.4a) and (C.4b) is considered as a zero-sum Markov game between the state-dependent Lagrange multipliers λ and the state-dependent policy π with the Lagrangian in (C.10) representing the cost for the decision maker. We define the Lagrangian $\mathbf{L}_{\pi,\Lambda}$ as the following min max value,

$$\mathbf{L}_{\pi,\Lambda} := val[\mathbf{V}_\pi + \Lambda \circ (\mathbf{S}_\pi - p\mathbb{1}_{|E|})], \tag{C.10}$$

where $\Lambda = (\lambda)_{i \in E} \in \mathbb{R}^N$ represents the vector of Lagrange multipliers λ with a multiplier associated to each statewise constraint, and $val[\cdot]$ operator is the solution of the $\min_{\pi \in \Delta(U)} \max_{\lambda \geq 0}[\cdot]$ (alternatively $\inf_{\pi \in \Delta(U)} \sup_{\lambda \geq 0}[\cdot]$ in case min max is unattainable) problem taken component-wise for (C.10). Let the immediate Lagrangian cost at time $t \leq \tau$ be $d_t = c_t + \lambda_t(k_t - p)$, where λ_t is the Lagrange multiplier associated with the state visited at time t. Suppose at time t, the Lagrangian values of states $1, \cdots, i$ have been updated asynchronously using Value iteration, then we can use the updated Lagrangian values of the same and use the old values for the remaining $i+1, \cdots, N$ states. Such a procedure is known as the Gauss-Seidel procedure and it generally has fast convergence as shown in [20]. In the sequel, the superscript on $\mathbf{L}$ will denote the iteration time. The Gauss-Seidel procedure for the Markov game is the iteration in value space for successive substitutions.

$$L^t(i) = d_t + \sum_{j=1}^{i-1} P_{ij}(\pi) L^t(j) + \sum_{j=i}^{N} P_{ij}(\pi) L^{t-1}(j). \tag{C.11}$$

Our proposed solution relies on the following two assumptions that imply the existence

Algorithm 11 p-Safe Lagrangian iteration

1: **Input**: i, p
2: Initialize $\mathbf{L} = 0$
3: **while** $t < \tau_{\mathcal{O}}$ **do**
4: Update $L^t(i)$

$$L^t(i) \leftarrow val\left[d_t + \sum_{j=1}^{i-1} P_{ij}(\pi)L^t(j) + \sum_{j=i}^{N} P_{ij}(\pi)L^{t-1}(j)\right]$$

5: $i \leftarrow i+1$
6: $t \leftarrow t+1$
7: **end while**

of p-safe Optimal policy π with corresponding λ.

Assumption 5.1 There exists at least one admissible p-safe policy π satisfying (C.4b) and for every admissible policy π, $\mathbf{L}_{\pi,\Lambda}$ is a continuous functions of π, Λ for every $i, j \in E$.

Assumption 5.2 There exists a pair of p-safe policy $\pi^\star$ and $\lambda^\star$ such that the optimum of Lagrangian $\mathbf{L}_{\pi,\Lambda}$ is attained.

Assumptions 5.1 and 5.2 imply that there exists a value of the zero-sum game (C.10) and as per [21] these assumptions together imply,

$$\min_{\pi\in\Delta(U)} \max_{\lambda\geq 0} [\mathbf{L}_{\pi,\Lambda}] = \max_{\lambda\geq 0} \min_{\pi\in\Delta(U)} [\mathbf{L}_{\pi,\Lambda}].$$

Given a Lagrangian $\mathbf{L}_{\pi,\Lambda}$ with dual variable Λ, the corresponding greedy policy $\pi^\star$ can be obtained as follows

$$\pi^\star = \arg \min_{\pi\in\Delta(U)} \mathbf{L}_{\pi,\Lambda}.$$

5.1 Resolution of counterexample due to [10]

Consider the example described by fig. C.2 and (C.9). Using (C.10), we rewrite (C.9) as the following unconstrained optimization problem

$$L(i) = \min_{\pi\in\Delta(U)} \max_{\lambda\geq 0} V_\pi(i) + \lambda(i)(S_\pi(i) - p), \tag{C.12}$$

with i being the initial state in fig. C.2 and $p = 0.125$. Applying algorithm 11 on (C.12) for state i and $t = 1$,

$$L(i) = val\Big[0.5(\lambda(i)(0.5 \times 0.2 - 0.125)) + 0.5(val\,[L(j)])\Big], \tag{C.13}$$

and for state j results in,

$$\begin{aligned} L(j) &= val\begin{bmatrix} 20 + \lambda(j)(0.05 - 0.125) \\ 10 + \lambda(j)(0.10 - 0.125) \end{bmatrix}, \\ L(j) &= 10. \end{aligned} \tag{C.14}$$

Therefore, the optimal action at state j is action b. Substituting (C.14) in (C.13) for iteration $t = 2$ gives us the optimal action at state i to be b as well. Thus $\mathcal{B}^i_{\pi^\star}[\mathbf{V}_{\pi^\star}] = \mathcal{B}^j_{\pi^\star}[\mathbf{V}_{\pi^\star}]$ and Bellman's principle of optimality is not violated.

6 RL Algorithm for p-Safe MDP

In this section, we propose an off-policy RL algorithm based on the Lagrangian iteration algorithm presented above. Define $Q(i, u, \lambda)$ as the state-action Lagrangian corresponding to $L(i)$. Assuming an initial admissible policy, the algorithm estimates the Lagrangian for the state visited (for ex. i) using stochastic approximation. It thereafter solves the corresponding saddle point problem by first evaluating $\max_{\lambda \geq 0} Q(i, u, \lambda)$ for the previous estimate and thereafter calculating the optimal action by evaluating $\min_{\pi \in \Delta(U)} Q(i, u, \lambda^\star)$, where $\lambda^\star \in \arg\max_{\lambda \geq 0} Q(i, u, \lambda)$. Specifically, the Q values are parameterized by λ and our algorithm is a model-free variant of [22]. Due to the presence of constraints, the optimal policy $\pi^\star(u \mid i)$ will be a randomized policy composed of optimal state-action occupation frequencies as shown in Chapter 8 of [8]. The goal of the following RL algorithm is to learn $\mathbf{L}_\pi$ by learning the corresponding state-action Q-values. Exploration-exploitation trade-off is handled by standard ϵ-greedy action selection. $\mathbf{L}^T_\pi$ obtained from Algorithm 12 achieves ε-optimality as per the following proposition.

Algorithm 13 p-Safe Q-learning

Input: ϵ, p, Max. number of episodes N_e
Initialize $Q(i, u, \lambda(i)) = 0 \ \forall i \in E, u \in U$
Initialize state-action counters f, g and learning rate $\alpha_i = 1 \ \forall i \in E$
for Each episode $e = 1 : N_e$ **do**
  Initialize state x_0 and policy $\pi_0 = \frac{1}{|U|}, \quad \forall u \in U$, and $\forall i \in E$
  while $t < \tau_{\mathcal{O}}$ **do**
    Given state x, Choose $u_t \leftarrow \begin{cases} u \sim \frac{1}{|U|} \text{ with probability } \epsilon \\ u \sim \pi_t(x) \text{ with probability } 1 - \epsilon \end{cases}$
    Observe new state $x_{t+1} = i$, cost c_t after applying u_t
    Update $f_i^{t+1} \leftarrow f_i^t + 1$
    if $i \in \mathcal{O}$ **then**
      Update $g_i^{e+1} = g_i^{e+1} + 1$
      Update estimated safety cost $k_{e+1}(i, u_t) = \frac{g_i^{e+1}}{g_i^{e+1} + f_i^{t+1}}$
    end if
    Update learning rate $\alpha_i \leftarrow \frac{1}{f_i^{t+1}}$
    $\lambda_{t+1}(i) \leftarrow \arg\max_{\lambda \geq 0} Q_t(i, u_t, \lambda)$
    $\pi_{t+1}(i) \leftarrow \arg \min_{\pi \in \Delta(U), V} V \quad \text{s.t.} \sum_{u \in U} \pi(u) Q_t(i, u, \lambda_{t+1}(i)) \leq V$
    $val[L_t(i)] \leftarrow V$
    Update Q value

$$Q_{t+1}(i, u, \lambda(i)) \leftarrow (1 - \alpha_i) Q_t(i, u, \lambda(i)) + \alpha_i (d_t + val[L_t(i)])$$

    $t \leftarrow t + 1$
  end while
end for

Proposition 6.1.
Consider the p-Safe MDP (C.4) with unknown costs and transition probability matrix. Define

$$c_M := \max_{i \in E, u \in U} c(i, u), \ \text{and} \ \phi_M := \max_{i \in E, u \in U} \lambda(i)(k(i, u) - p),$$

where $\lambda(i) = \max_{t < \tau_{\mathcal{O}}} \lambda^t(i)$ *with* t *being the iteration time in Algorithm 12. Let* T *be the final time. For an* $\varepsilon > 0$,

$$\mathbf{L}_{\pi}^{\star} - \mathbf{L}_{\pi^T}^{T} \leq \varepsilon, \ \text{where} \ \mathbf{L}_{\pi^T}^{T} = val[\mathbf{L}^T] \ \text{from Algorithm 12,}$$

if Assumption 5.1 is satisfied and Algorithm 12 runs upto time T *given by,*

$$T \geq \frac{1}{p} \log \left(\frac{c_M + \phi_M}{\varepsilon p} \right).$$

Proof. — Consider an admissible policy π. Note that π is a p-Safe policy since any admissible policy must satisfy the constraint (C.4b). Therefore, p is the maximal probability of stopping due to $X_t \in \mathcal{O}$. Then the optimal Lagrangian along this path will be,

$$\mathbf{L}_\pi^\star = c_1^\star + \phi_1^\star + (1-p)(c_2^\star + \phi_2^\star) + (1-p)^2(c_3^\star + \phi_3^\star) + \cdots,$$

where c_t and $\phi_t := \lambda_t(k_t - p)$ represent the costs incurred at iteration time t of Algorithm 12. Define $\gamma := 1-p$ as the maximal probability of continuation (or not stopping) among all the states. Due to Assumption 5.1, the expected optimal Lagrangian will be bounded as follows

$$\begin{aligned}\mathbf{L}_\pi^\star &\leq c_1 + \phi_1 + \gamma(c_2 + \phi_2) + \gamma^2(c_3 + \phi_3) + \cdots \\ &\leq \sum_{t=1}^{T} \gamma^{t-1}(c_t + \phi_t) + (c_M + \phi_M)\gamma^T \sum_{t=0}^{\infty} \gamma^t \\ &\leq \mathbf{L}_{\pi^T}^T(\omega) + (c_M + \phi_M)\gamma^T \frac{1}{1-\gamma}.\end{aligned} \tag{C.15}$$

We now bound the last term in equation (C.15) as

$$(c_M + \phi_M)\gamma^T \frac{1}{1-\gamma} \leq \varepsilon. \tag{C.16}$$

Note that since $0 < \gamma < 1$, $\varepsilon \to 0$ as $T \to \infty$. Rearranging the terms in (C.16),

$$\begin{aligned}\gamma^T &\leq \frac{\varepsilon(1-\gamma)}{c_M + \phi_M} \\ T\log\gamma &\leq \log\left(\frac{\varepsilon(1-\gamma)}{c_M + \phi_M}\right) \\ T(-\log\gamma) &\geq \log\left(\frac{c_M + \phi_M}{\varepsilon(1-\gamma)}\right).\end{aligned} \tag{C.17}$$

Substituting

$$\log\gamma = (\gamma - 1) - \frac{1}{2}(\gamma-1)^2 + \frac{1}{3}(\gamma-1)^3 - \frac{1}{4}(\gamma-1)^4 + \cdots, \tag{C.18}$$

in (C.17) (and ignoring higher order terms since $0 < \gamma < 1$) completes the proof. ∎

7 Simulation Results

In this section, we simulate Algorithm 12 on a robot in a windy grid world state space in an episodic setting. We consider a 5×5 grid where the first coordinate represents

the x-coordinate of the robot and the second coordinate represents the y-coordinate of the robot. Suppose the robot chooses an action u_t, then the next state is given by

$$x_{t+1} = x_t + \begin{cases} w_t, & \text{with probability } \sigma \\ u_t, & \text{otherwise,} \end{cases} \tag{C.19}$$

where w_t is a random action from a uniform distribution and σ is the probability of encountering wind in a given state. For example, a valid state for the robot can be $[1, 1]$, and choosing action $[0, 1]$ results in a new state $[1, 2]$. The goal for the robot is to reach the goal state while avoiding an unsafe state without any knowledge of dynamics (C.19). A reward (or a negative cost) of $+1$ is incurred for transitioning to the goal state and -1 for transitioning to an unsafe state. Visits to the unsafe state or goal state result in the termination of the episode as the stopping condition given by Definitions 3.4 and 3.3 is satisfied. Within 800 episodes, Algorithm 12 was able to obtain optimal

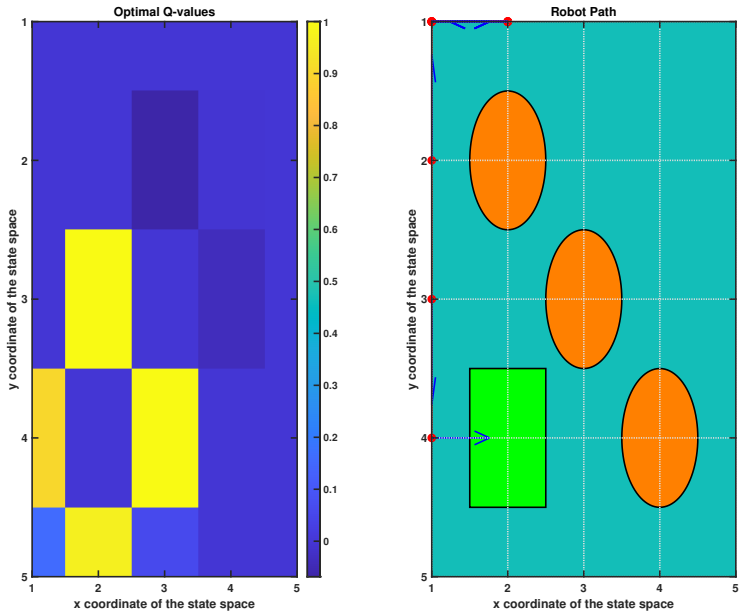


Fig. C.3: The robot moves in a grid world state space as shown in the right image. The green rectangle represents the goal state $\mathcal{G}$ and the orange circles represent the unsafe states $\mathcal{O}$. Red dots represent the states visited by the robot during the episode and blue arrows are the corresponding actions. The left image represents the optimal Q-values corresponding to the grid on the right image. The values are visualized by the bar chart.

Q-values as shown in the fig. C.3.

8 Conclusion

It can be concluded from this work, that RL or any dynamic programming-based approach cannot be trivially applied to Reach-Avoid problems in the case of MDPs as Bellman's principle of optimality may not hold. We resolve this issue by using the $val[\cdot]$ operator defined in Section V. The $val[\cdot]$ operator forces the decision maker to select policies that are independent of the initial state and only depend on the current state. This also makes sense intuitively for counterexample due to [10] as once the state trajectory reaches state j in fig. C.2, it does not make sense to consider $P(S_1)$ as there is no chance of going to Chain 1 from j. From the proof of Proposition 1, it is evident that Algorithm 11 does not guarantee p-Safe policy during the training phase of Algorithm 11, and the future work will involve calculating bounds on the number of times the constraint is violated during the learning phase and proposing a safe on-policy RL algorithm which guarantees safety during exploration.

References

[1] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013.

[2] B. Lütjens, M. Everett, and J. P. How, "Safe reinforcement learning with model uncertainty estimates," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8662–8668.

[3] M. Tejedor, A. Z. Woldaregay, and F. Godtliebsen, "Reinforcement learning application in diabetes blood glucose control: A systematic review," *Artificial intelligence in medicine*, vol. 104, p. 101836, 2020.

[4] S. Summers, M. Kamgarpour, J. Lygeros, and C. Tomlin, "A stochastic reach-avoid problem with random obstacles," in *Proceedings of the 14th international conference on Hybrid systems: computation and control*, 2011, pp. 251–260.

[5] R. Wisniewski and M. L. Bujorianu, "Probabilistic safety guarantees for markov decision processes," *IEEE Transactions on Automatic Control*, 2023.

[6] E. Altman, *Constrained Markov decision processes: stochastic modeling*. Routledge, 1999.

[7] M. El Chamie, Y. Yu, B. Açıkmeşe, and M. Ono, "Controlled markov processes with safety state constraints," *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1003–1018, 2018.

[8] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 1994.

[9] M. Kearns and S. Singh, "Near-optimal reinforcement learning in polynomial time," *Machine learning*, vol. 49, pp. 209–232, 2002.

[10] M. Haviv, "On constrained markov decision processes," *Operations research letters*, vol. 19, no. 1, pp. 25–28, 1996.

[11] E. K. Chong, S. A. Miller, and J. Adaska, "On bellman's principle with inequality constraints," *Operations research letters*, vol. 40, no. 2, pp. 108–113, 2012.

[12] Y. Chow and M. Pavone, "A time consistent formulation of risk constrained stochastic optimal control," *arXiv preprint arXiv:1503.07461*, 2015.

[13] R. C. Chen and G. L. Blankenship, "Dynamic programming equations for discounted constrained stochastic control," *IEEE transactions on automatic control*, vol. 49, no. 5, pp. 699–709, 2004.

[14] R. C. Chen and E. A. Feinberg, "Non-randomized policies for constrained markov decision processes," *Mathematical Methods of Operations Research*, vol. 66, pp. 165–179, 2007.

[15] E. Altman and A. Shwartz, "Adaptive control of constrained markov chains: Criteria and policies," *Annals of Operations Research*, vol. 28, no. 1, pp. 101–134, 1991.

[16] Y. Efroni, S. Mannor, and M. Pirotta, "Exploration-exploitation in constrained mdps," *arXiv preprint arXiv:2003.02189*, 2020.

[17] V. S. Borkar, "An actor-critic algorithm for constrained markov decision processes," *Systems & control letters*, vol. 54, no. 3, pp. 207–213, 2005.

[18] D. Ding, K. Zhang, T. Basar, and M. Jovanovic, "Natural policy gradient primal-dual method for constrained markov decision processes," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8378–8390, 2020.

[19] R. Bellman, "Dynamic programming," *Princeton University Press, New Jersey*, 1957.

[20] H. J. Kushner, "The gauss-seidel numerical procedure for markov stochastic games," *IEEE transactions on automatic control*, vol. 49, no. 10, pp. 1779–1784, 2004.

[21] S. Boyd and L. Vandenberghe, "Convex optimization, cambridge univ," *Press, UK*, 2004.

[22] A. J. Hoffman and R. M. Karp, "On nonterminating stochastic games," *Management Science*, vol. 12, no. 5, pp. 359–370, 1966.

Paper D

Decentralized control of a water distribution network using Repeated Games

Rahul Misra, Carsten S. Kallesøe, Rafał Wisniewski

The paper has been published in the
27th International Conference on Methods and Models in Automation and Robotics (MMAR), 2023, Aug 22, pp. 181-186

The layout has been revised.

Abstract

In this paper, our aim is to design a decentralized control scheme for pumping stations in a water distribution network that supplies drinking water. The considered water distribution network consists only of pumping stations, piping networks, and consumers (since the inclusion of storage tanks poses the risk of contamination of drinking water). The pumping stations supply water to consumers and their objective is to ensure the supply of water to the consumers in an optimal way such that the consumer demand is satisfied with minimum energy consumption by the pumping station itself. This gives rise to a nonzero-sum game as the pumping stations have a common objective of satisfying consumer demand and a selfish objective of minimizing their own energy consumption. A real-life water distribution network with two pumping stations was emulated in a lab and the proposed control scheme was tested on this setup. The consumer demand follows a periodic trend that mimics real-life consumption and has some stochastic noise added to it so as to emulate uncertainty in consumer demand. The proposed control scheme was able to track the reference signal while each pumping station was minimizing its own energy consumption.

Keywords: Decentralized and distributed control, Game theory, Minimax strategies, Linear Programming, Optimization and control of large-scale network systems.

1 Introduction

A Water Distribution Network (WDN) consists of pumping stations whose aim is to supply water to the consumers which are connected via a piping network. If the pressure on the consumer side is too high then pipe bursts and subsequent leakages may occur in the WDN and if it is too low then the consumption demand will not be fulfilled. Consumer demands are stochastic but they follow a periodic trend (for example, the average consumption of water is relatively low at night i.e. between 22h and 06h). Therefore, the aim of the pumping stations is to ensure the maintenance of pressure on the consumer side despite fluctuating consumer demands. Furthermore, each of the pumping stations would also like to independently minimize its own energy consumption while satisfying consumption demand (see [1] and [2]). Optimal control of a real-life WDN is a challenging problem from a control perspective as it is a complex multi-input and multi-output system spread over large geographic distances. Nevertheless, a lot of research has been done in this field. The book [3] provides a good overview of the existing state-of-art in the field of optimal control, fault identification, and fault-tolerant control for WDN. The paper [4] focuses on the control of water quality in a WDN. Application of model predictive control on WDNs is discussed in [3] and [5]. In this work, we consider WDN which provides drinking water and therefore, we do not include storage tanks as they are being phased out due to the risk of contamination [6].

Game theory has emerged as a suitable paradigm for distributed control of multi-agent systems as discussed in [7], [8] and [9]. We will specifically focus on the theory of Repeated games and an introduction to it can be found in the book [10]. Game theory has also been applied on WDN in the papers [11] and [12]. Our approach differs from the former in the sense that the pumping stations have a selfish objective of minimizing their own energy consumption and are not concerned about the energy consumption of other pumping stations which results in a nonzero-sum game formulation and therefore potential function approach of [11] cannot be applied in this case. In the latter paper, the authors have considered a stochastic differential equations-based model of WDN and discretized it in space and time which can be computationally prohibitive for a large-scale WDN.
In [13], a simplified model of WDN based on solving static equations at each time instant is presented. It is essentially a mapping between control and pressure reading from the consumer side under some mild assumptions on pressure drop and consumer demands. The costs incurred by the controller serve as the feedback signal. An implicit equation needs to be solved in order to use the aforementioned mapping and we have used Newton's method [14] for solving the same. Thereafter, we have used this model to derive local supervisory control schemes for each of the pumping stations which gives set points to their individual local controllers. This approach is tested in the Smart Water Lab at Aalborg University (see [15] for more information on the laboratory). To the best of the author's knowledge, this is the first work, where static equations are used to provide decentralized control of WDN using game theory and that forms the primary contribution of this paper.
The rest of the paper is organized as follows. We introduce some standard notation used throughout this paper in the next subsection. Thereafter, we introduce the model of WDN in section 2. The control design and algorithm are presented in section 3. The control algorithm designed in section 3 was applied on the Smart Water Lab and the results are presented in section 4. Finally, we conclude this paper and highlight future research topics.

Notation: Let superscript $k \in \{1, \cdots, N\}$ denote a generic player (or controller) in an N player game and $-k = \{1, \cdots, k-1, k+1, \cdots, N\}$ denote all other players except player k. $\Delta(n)$ denotes the probability simplex over $\mathbb{R}^n$. The notation $M(n, m; \mathbb{R})$ denotes an $n \times m$ matrix (hence M) with entries belonging to the set of $\mathbb{R}$. The superscript T denotes the transpose operator and the superscript $-T$ indicates the inverse for a transposed matrix. $\mathbb{1}_n$ denotes an n dimensional vector of 1's. The subscript $\mathcal{T}$ denotes the spanning tree and subscript $\mathcal{C}$ denotes the chords of the associated graph. Gaussian distribution is denoted by $\mathcal{N}\left(\mu, \sigma^2\right)$ with mean μ and variance σ^2.

2 Describing WDN using static equations

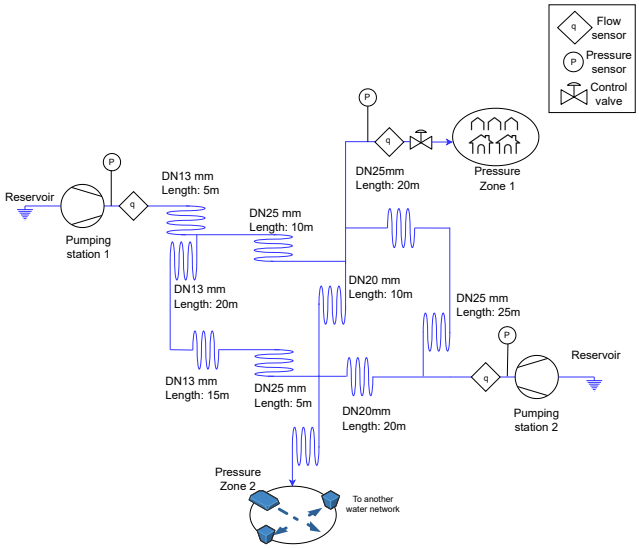


Fig. D.1: Process and Instrumentation Diagram of the considered WDN.

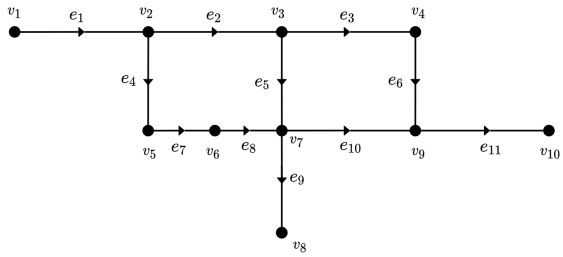


Fig. D.2: Graph of the WDN in fig. D.1.

This section introduces the model for WDN and is based on [13]. We have extended the work of [13] by presenting Newton's algorithm for solving implicit equations in subsection B. Consider a WDN as shown in fig. D.1. Such a network can be modeled as a directed graph as shown in fig. D.2. The edges represent piping networks and vertices represent pressure nodes where water can either flow in the system (if a pumping station is connected to it) or can flow out of the system (if a consumer is connected to it).

Consider a network with graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ represents the set of n vertices and $\mathcal{E}$ represents the set of m edges. We begin by defining how to construct the incidence matrix H which encodes the interconnections between different components of the WDN in fig. D.1

$$H_{i,j} = \begin{cases} -1, & \text{if the } j^{th} \text{ edge is entering } i^{th} \text{ vertex,} \\ 0, & \text{if the } j^{th} \text{ edge is not connected to the } i^{th} \text{ vertex,} \\ 1, & \text{if the } j^{th} \text{ edge is leaving } i^{th} \text{ vertex,} \end{cases} \tag{D.1}$$

and how to construct the cycle matrix B which encodes the information about the edges which belong to cycles (or loops) and their orientation for a generic WDN.

$$B_{i,j} = \begin{cases} -1, & \text{if the } j^{th} \text{ edge belongs to the } i^{th} \text{ cycle and their directions disagree,} \\ 0, & \text{if the } j^{th} \text{ edge does not belong to the , } i^{th} \text{ cycle,} \\ 1, & \text{if the } j^{th} \text{ belongs to the } i^{th} \text{ cycle and their directions agree.} \end{cases} \tag{D.2}$$

Let p represent the vector of pressure values at all the vertices, Δp represent the differential pressure across all the edges, and q represent the vector of flows in the edges. Then by Ohm's law, there exists a resistance equation that describes the relationship between the pressures across the edges and the flows through the edges

$$\Delta p = H^T(p + z) = f(q), \tag{D.3}$$

where $f \in \mathbb{R}^m$ is a vector of the flow-dependent resistance-related pressure drops and $z \in \mathbb{R}^n$ is the geodetic elevation of the vertices. f has the following structure $f(q) = \left(f_1\left(q_1\right), \cdots, f_m\left(q_m\right)\right)^T$ as per [16]. All networks follow Kirchoff's vertex law (or node law) and this can be expressed by the following equation

$$Hq = d, \tag{D.4}$$

where $d \in \mathbb{R}^n$ is the demand vector holding the demand for each of the n vertices. Since the WDN is a closed network, there can be only $n-1$ independent nodal demands which imply $\sum_{i=1}^{n} d_i = 0$ due to mass conservation law (equivalent to Kirchoff's vertex law in this case). Note that from [17], the kernel of H^T is spanned by $\mathbb{1}$ meaning that $\mathbb{1}^T H = 0$. Therefore, $\mathbb{1}^T Hq = 0 = \mathbb{1}^T d$, which indirectly impose the constraint $\sum_{i=1}^{n} d_i = 0$.

Equations (D.3) and (D.4) are sufficient to represent a WDN as they together represent a mapping between consumer demand and pressure at vertices. We can control the pressure at some of the vertices directly (as a pumping station is connected to them) and indirectly at other vertices (as no pumping station is connected to them). This

makes it necessary for us to partition the WDN into vertices, where pressure can be directly controlled (henceforth referred to as controlled vertices) and vertices, where pressure can be measured but only be controlled indirectly (henceforth referred to as non-controlled vertices or measured vertices). In the sequel, we shall present a partitioning of the WDN model into controlled vertices and non-controlled vertices. To that end, we put the following assumptions on the model considered so far.

Assumption 2.1 The resistance related pressure drop of the i^{th} edge is given by $f_i(q_i) = r_i |q_i| q_i$, where $r_i > 0$.

Assumption 2.2 The demands related to non-pressure controlled vertices $\bar{d}$ are given by $\bar{d} = \bar{v}D + e$, where $D = -\sum_{i=1}^{n-c} \bar{d}_i$ is the total water demand from the WDN (-ve sign represents water being taken out of the system by consumers), $\bar{v}$ is a constant vector with $\sum_{i=1}^{n-c} \bar{v}_i = 1$ representing the distribution of water demand among vertices, and $e \sim \mathcal{N}(0, \sigma^2)$.

2.1 Partitioning of Model

We will now partition the model of WDN by collecting vertices into two sets. One set denoted by $\bar{p}, \bar{z}, \bar{d} \in \mathbb{R}^{n-c}$ where a subset of the pressures $\bar{p}$ are measured, and another set of vertices denoted $\hat{p}, \hat{z}, \hat{d} \in \mathbb{R}^{c}$, where the pressures $\hat{p}$ are controlled, thus pumping stations are controlling the c vertex pressures $\hat{p}$ and deliver the flows $\hat{d}$. Hence $n - c$ represents the uncontrolled nodes. This partitioning allows us to model the relationship between measured pressures on output water flow and controlled input pressures.

Without loss of generality we sort the vertices such that $p = \begin{pmatrix} \bar{p}^T & \hat{p}^T \end{pmatrix}^T$, $z = \begin{pmatrix} \bar{z}^T & \hat{z}^T \end{pmatrix}^T$, and $d = \begin{pmatrix} d^T & \hat{d}^T \end{pmatrix}^T$. Also, we sort the edge flows q into two sets, such that $q = \begin{pmatrix} q_{\mathcal{T}}^T & q_{\mathcal{C}}^T \end{pmatrix}^T$. From [13] a partitioning always exists where $q_{\mathcal{T}} \in \mathbb{R}^{n-c}$ is chosen such that $\bar{H}_{\mathcal{T}} \in \mathbb{R}^{n-c \times n-c}$ is invertible. With this definition of the flow and pressure vectors, the incidence matrix is partitioned into

$$H = \begin{pmatrix} \bar{H}_{\mathcal{T}} & \bar{H}_{\mathcal{C}} \\ \hat{H}_{\mathcal{T}} & \hat{H}_{\mathcal{C}} \end{pmatrix}. \tag{D.5}$$

The following Lemma makes it possible to partition the incidence matrix H while ensuring the existence of its inverse. This is required for solving for non-controlled node pressures in (D.8).

Lemma 2.1 ([13]).
Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a connected and directed graph with incidence matrix $H \in M(n, m; \{-1, 0, 1\})$. Furthermore, let $\mathcal{V} = \{\mathcal{V}, \hat{\mathcal{V}}\}$ be a partitioning such that $\hat{\mathcal{V}} = \{\hat{v}_1, \cdots, \hat{v}_c\}$ is non-empty and $\mathcal{E} = \{\mathcal{E}_\mathcal{T}, \mathcal{E}_\mathcal{C}\}$ be a partitioning such that the corresponding sub-matrix $\bar{H}_\mathcal{T}$ of H is square and invertible. Then the following is true

$$-\bar{H}_\mathcal{T}^{-T} \hat{H}_\mathcal{T}^T \mathbb{1}_c = \mathbb{1}_{n-c} \tag{D.6}$$

The following Lemma allows partitioning of the matrix B and writing B in terms of the partitioned incidence matrix H given by (D.5).

Lemma 2.2 ([13]).
The matrix B can be rewritten in terms of partitioned incidence matrix (D.5) as $B = \hat{H}_\mathcal{C}^T - \bar{H}_\mathcal{C}^T \bar{H}_\mathcal{T}^{-T} \hat{H}_\mathcal{T}^T$. Then, $B \in M(m - n + c, c; \mathbb{R})$ has a non-trivial kernel, and $\ker(B) = \text{span}\{\mathbb{1}_c\}$.

With the partitioning of H as per (D.5) the network model described by (D.3) and (D.4) can be rewritten as

$$f_\mathcal{T}(q_\mathcal{T}) = \bar{H}_\mathcal{T}^T(\bar{p} + \bar{z}) + \hat{H}_\mathcal{T}^T(\hat{p} + \hat{z}), \tag{D.7a}$$
$$f_\mathcal{C}(q_\mathcal{C}) = \bar{H}_\mathcal{C}^T(\bar{p} + \bar{z}) + \hat{H}_\mathcal{C}^T(\hat{p} + \hat{z}), \tag{D.7b}$$
$$\bar{H}_\mathcal{T} q_\mathcal{T} + \bar{H}_\mathcal{C} q_\mathcal{C} = \bar{d}, \tag{D.7c}$$
$$\hat{H}_\mathcal{T} q_\mathcal{T} + \hat{H}_\mathcal{C} q_\mathcal{C} = \hat{d}. \tag{D.7d}$$

Rewriting (D.7) the following expression describes the non-controlled vertex pressures of the network $\bar{p}$

$$\bar{p} = \bar{H}_\mathcal{T}^{-T} f_\mathcal{T}\left(-\bar{H}_\mathcal{T}^{-1} \bar{H}_\mathcal{C} q_\mathcal{C} + \bar{H}_\mathcal{T}^{-1} \bar{d}\right) - \bar{H}_\mathcal{T}^{-T} \hat{H}_\mathcal{T}^T(\hat{p} + \hat{z}) - \bar{z}, \tag{D.8}$$

and the flow due to the controlled vertices $\hat{d}$ are given by

$$\hat{d} = \left(\hat{H}_\mathcal{C} - \hat{H}_\mathcal{T} \bar{H}_\mathcal{T}^{-1} \bar{H}_\mathcal{C}\right) q_\mathcal{C} + \hat{H}_\mathcal{T} \bar{H}_\mathcal{T}^{-1} \bar{d} \tag{D.9}$$

Since the value of $\bar{p}$, can be measured using a pressure sensor for the vertex (which we are interested in controlling), $\bar{d}$ is measured as consumer demand and $\hat{p}$ is control input due to pumping stations (with $\hat{d}$ being the corresponding water flow from the pumping station), the only unknowns in (D.8) and (D.9) are the chord flows in $q_\mathcal{C}$. We shall

now derive implicit equations from which these unknown chord flows can be obtained. Rearranging the terms in (D.7c), we can obtain the tree flows in spanning tree $q_{\mathcal{T}}$ as

$$q_{\mathcal{T}} = -\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}}q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}. \tag{D.10}$$

Using (D.10), (D.7a) and (D.7b), we can derive the following implicit expression which allows us to calculate the necessary chord flows.

$$f_{\mathcal{C}}\left(q_{\mathcal{C}}\right) - \bar{H}_{\mathcal{C}}^{T}\bar{H}_{\mathcal{T}}^{-T}f_{\mathcal{T}}\left(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}}q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}\right) = \left(\hat{H}_{\mathcal{C}}^{T} - \bar{H}_{\mathcal{C}}^{T}\bar{H}_{\mathcal{T}}^{-T}\hat{H}_{\mathcal{T}}^{T}\right)(\hat{p}+\hat{z}). \tag{D.11}$$

Equations (D.11), (D.8) and (D.9) summarize the partitioned model which will be used for our reference controller.

2.2 Solving implicit equation using Newton method

It is necessary to solve (D.11) for obtaining necessary edge flows which in turn solve (D.8) and (D.9). This is done using Newton's method. We begin by describing the error term ϵ by rearranging (D.11) to obtain

$$\epsilon(q_{\mathcal{C}}) = f_{\mathcal{C}}\left(q_{\mathcal{C}}\right) - \bar{H}_{\mathcal{C}}^{T}\bar{H}_{\mathcal{T}}^{-T}f_{\mathcal{T}}\left(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}}q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}\right) - \left(\hat{H}_{\mathcal{C}}^{T} - \bar{H}_{\mathcal{C}}^{T}\bar{H}_{\mathcal{T}}^{-T}\hat{H}_{\mathcal{T}}^{T}\right)(\hat{p}+\hat{z}). \tag{D.12}$$

$\epsilon(q_{\mathcal{C}}) \approx 0$ implies (D.11) is approximately solved. Let $R(q_{\mathcal{C}}) \in M(m-n+c, m-n+c, \mathbb{R})$ denote a diagonal matrix with diagonal entry being $0.5r_{\mathcal{C}}\left|q_{\mathcal{C}}\right|$ and similarly let $R(q_{\mathcal{T}}) \in M(n-c, n-c, \mathbb{R})$ denote a diagonal matrix with diagonal entry being $0.5r_{\mathcal{T}}\left|q_{\mathcal{T}}\right|$. We further define $G = \bar{H}_{\mathcal{C}}^{T}\bar{H}_{\mathcal{T}}^{-T}$. The derivative of error $\epsilon(q_{\mathcal{C}})$ with respect to $q_{\mathcal{C}}$ is given as

$$\frac{d\epsilon(q_{\mathcal{C}})}{dq_{\mathcal{C}}} = R(q_{\mathcal{C}}) + G^{T}R(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}}q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d})G. \tag{D.13}$$

The cost $V(q_{\mathcal{C}}) = \frac{1}{2}\epsilon^{2}(q_{\mathcal{C}})$ with $\nabla V(q_{\mathcal{C}}) = \frac{d\epsilon(q_{\mathcal{C}})}{dq_{\mathcal{C}}}^{T}\epsilon(q_{\mathcal{C}})$ is minimized using the following algorithm. In Algorithm 14, α is the step-size, β is a regularizing term used for ensuring positive-definiteness of the Hessian, $I_{m\times m}$ is an identity matrix of size $m \times m$ and γ is error tolerance.

3 Control design and Algorithm

In this section, we shall design a decentralized control scheme for the pumping stations. The control objectives (defined in the subsection 3.2) lead to a nonzero-sum game that is computationally intractable in general (see [9]). However, a conservative solution to the aforementioned game can be found by using Minimax or Security strategies (see [8] and [9]). The information structure of the game is summarized in the following assumption.

Algorithm 14 Implicit equation solver

Input: α, β, γ
Initialize $q_{\mathcal{C}} \leftarrow 1$
while $||V_t - V_{t-1}||_2 > \gamma$ **do**
    $q_{\mathcal{T}}^t \leftarrow -\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}}q_{\mathcal{C}}^t + \bar{H}_{\mathcal{T}}^{-1}\bar{d}$
    Update ϵ_t using (D.12)
    Update cost $V_t \leftarrow \epsilon_t^2$
    Newton step $q_{\mathcal{C}}^{t+1} \leftarrow q_{\mathcal{C}}^t - \alpha_t \frac{\nabla V_t(q_{\mathcal{C}}^t)}{(\nabla V_t^2(q_{\mathcal{C}}^t)+\beta I_{m\times m})}$
    Update step size $\alpha_t \leftarrow \frac{1}{2}\alpha_t$
    $t \leftarrow t+1$
end while

Assumption 3.1 A player only knows the possible finite control actions that *can be taken* by the other players.

3.1 Decentralized control by solving Repeated game

The control design is based on solving a static game at each time instant (hence a Repeated game since the same static game is repeated at each time instant). Formally a static game Γ with N players can be defined as a tuple $\Gamma = \{N, (U^1 \times \cdots U^N), (C^1, \cdots, C^N)\}$, where U^k is the finite control space of player k, C^k is the cost operator (matrix in 2-player case) for player k. The finite control space U^k is obtained by discretizing the continuous control space into finite control actions. For the considered WDN, these represent the operating power of pumping stations (for ex. $u^k = 1$ implies that the pumping station k is operating at 10% of its maximum capacity). The cost operator for player k playing the game Γ can be defined as $C^k = [c^k(u^1, \cdots, u^N)]$, where the entry $c^k(u^1, \cdots, u^N)$ represents the instantaneous cost for player k if player 1 plays action u^1, player 2 plays action u^2 and so on i.e. the joint action profile is $u^1, \cdots, u^N$. The cost operator for the next time-step (defined in the subsection 3.2) is constructed using the model (D.8), (D.9), and (D.11). The cost operator serves as the controller's feedback signal as it considers real-time consumer demand and ensures fulfillment of the same. Any nonzero-sum game $\Gamma = \{N, (U^1 \times \cdots U^N), (C^1, \cdots, C^N)\}$ can be solved using Minimax strategies if each player k solves the corresponding zero-sum game $\Gamma' = \{N, (U^1 \times \cdots U^N), (C^k, -C^k)\}$ to obtain their worst-case costs.

Let V_t^k denote the minimax value of the game Γ at time t for a player k and let $\pi_t^k \in \Delta(u^k)$ denote the mixed strategy of player k at time t. Then the following Linear

program can be solved by player k for finding the minimax strategy π_t^k.

$$\min_{\pi_t^k, V_t^k} \quad V_t^k \tag{D.14a}$$

$$\text{s.t.} \quad \sum_{u^k \in U} c_t^k(u^k, u^{-k})\pi_t^k(u^k) \leq V_t^k, \forall u^{-k} \in U, \tag{D.14b}$$

$$\sum_{u^k} \pi_t^k(u^k) = 1, \tag{D.14c}$$

$$\pi_t^k(u^k) \geq 0, \ \forall u^k \in U, \tag{D.14d}$$

In linear program (D.14), the constraints (D.14b) ensures the best response by player k to all possible control actions by the player(s) $-k$. Note that, there will be a constraint (D.14b) for every possible control action by the player(s) $-k$. Constraints (D.14c) and (D.14d) ensure that $\pi_t^k \in \Delta(u^k)$ while we search for optimal π_t^k.

3.2 Formulation of cost function

The control objective for both players consists of a common goal of tracking reference pressure and both players simultaneously have an individual objective of minimizing their energy consumption. The following cost function for player k implements these objectives,

$$c^k = W_1 \left|\bar{p} - p_0\right| + (\bar{p} - p_0)^T W_2 (\bar{p} - p_0) + W_3 \left|\hat{d}^k u^k\right|, \tag{D.15}$$

where $\bar{p}$ is the pressure as per (D.8), p_0 is the reference pressure which we want to maintain, u^k represents the controlled pressure input from k^{th} controller, $\hat{d}^k$ is the flow from k^{th} controller as per (D.9), $|\cdot|$ represents the standard 1-norm, W_1, W_2 and W_3 are normalized weights. The first term in (D.15) represents the absolute mean of pressure difference at the consumer vertex, the second term represents the variance of pressure difference at the consumer vertex and the third term represents the absolute energy consumption (Unit: W) for a pumping station k. We will now state an online algorithm (Algorithm 15) for solving repeated games based on linear program (D.14). Note that (D.11) is used for constructing C_t^k in Algorithm 15 and (D.11) is solved using Algorithm 14.

4 Lab results

Algorithm 15 was applied on Smart Water Lab at Aalborg University (see fig. D.3). The lab has a modular design and we have used 2 pumping station modules, 1 consumer station module, and 2 piping modules for emulating WDN given in fig. D.1. The lab modules communicate with the SCADA (Supervisory Control and Data Acquisition) using MODBUS and further details on lab modules can be found in the paper [15].

Algorithm 15 Online Model-based Repeated games solver

Input: $\bar{p}$, p_0, d_c
Calculate $\bar{d}_c$ by averaging d_c since last decision epoch
for $t = 1, \cdots, T$ **do**
    for All possible control actions of all players **do**
        Construct C_t^k using (D.8), (D.9), (D.11) and (D.15)
    end for
    Solve the game Γ' using (D.14) for π_t^k
    Sample $u_t^k \sim \pi_t^k$
    Apply u_t^k as control input
    $t \leftarrow t + 1$
end for

Fig. D.3: Smart water Lab at Aalborg University with SCADA computer at right, pumping and consumer modules visible in the center, and Piping module at left.

In this study, each pumping station *independently* used Algorithm 15 as a supervisory controller which provided optimal pressure setpoints to the local controllers within the pumping station (which implemented the optimal setpoint) similar to an economic model predictive control. For a 2 player game as per considered WDN in fig. D.1, the cost operator for each player will be a matrix and for the rest of the paper, we will focus on the 2 player case without loss of generality (Note that Minimax strategies exist for an N-player game [8], [9]). As the demand curve (explained in (D.16) and shown in fig. D.4) changes slowly, Algorithm 15 was executed once every 300 seconds and we consider that as the decision epoch. The reference pressure p_0 was chosen to be 0.5 *bar*. For Algorithm 14, we chose $\alpha_0 = 100$, $\beta = 10^{-4}$ and $\gamma = 10^{-7}$. The weights W_1, W_2 and W_3 in (D.15) were set to the same values for both the pumps and their values are as follows; $W_1 = 10^6$, $W_2 = 10^6$, and $W_3 = 1$. These weights were manually

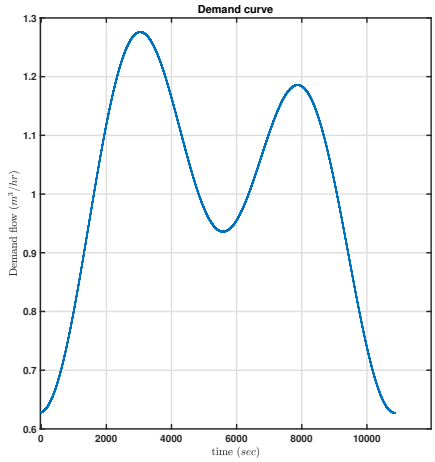


Fig. D.4: The simulated periodic demand trend using (D.16).

tuned to obtain good performance. The consumer demand has been simulated to match the periodic trend discussed in [15] and references therein. Fig. D.7 shows the costs incurred by pump 1 and 2. Decision epochs represent the time when Algorithm 15 was used to update setpoints for the local controllers. It can be observed that both pumps incur almost similar costs. The consumer demand is maximum during the morning hours (around 07h) and falls during midday (around lunchtime 12h) and rises again in the evening (around 17h) before falling to the minimum during the night (around 24h) and the pattern repeats itself. The following Fourier series equation (presented in [18]) was used to simulate the demand curve d_c,

$$d_c = a_0 + a_1 \cos \omega t_h + b_1 \sin \omega t_h + a_2 \cos \omega t_h + b_2 \sin \omega t_h, \tag{D.16}$$

where $a_0 = 1$, $a_1 = -0.155$, $b_1 = 0.044$, $a_2 = -0.217$, $b_2 = -0.005$, $\omega = 0.261$ and t_h is the time instant. The simulated demand trend can be seen in fig. D.4. Fig. D.5 shows the reference tracking despite disturbances due to consumption by consumers. The pressure at node 3 does not perfectly coincide with the reference pressure due to the aforementioned flow disturbances due to consumption at node 3 by the consumer. The demand curve as shown in fig. D.5 is a scaled version of fig. D.4 and Gaussian noise was added to it in order to reflect the real-life consumption. Fig. D.6 shows the control inputs applied by pump 1 and 2. It can be observed that pump 1 is compensating for the disturbance shown in fig. D.5 and pump 2 is supplying almost constant pressure with a magnitude similar to pump 1.

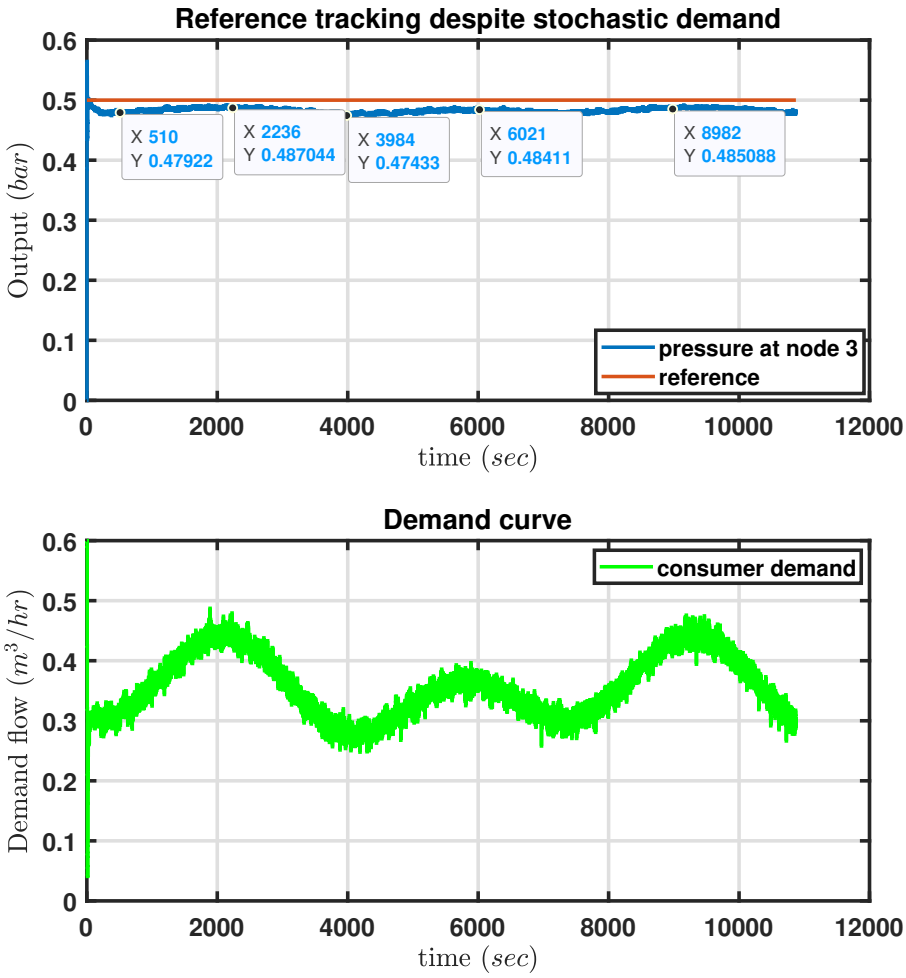


Fig. D.5: The reference is tracked up to a certain error (maximum error being approximately 0.0256 *bar*) despite changing consumer demands. The consumer demand is shown in the bottom subplot.

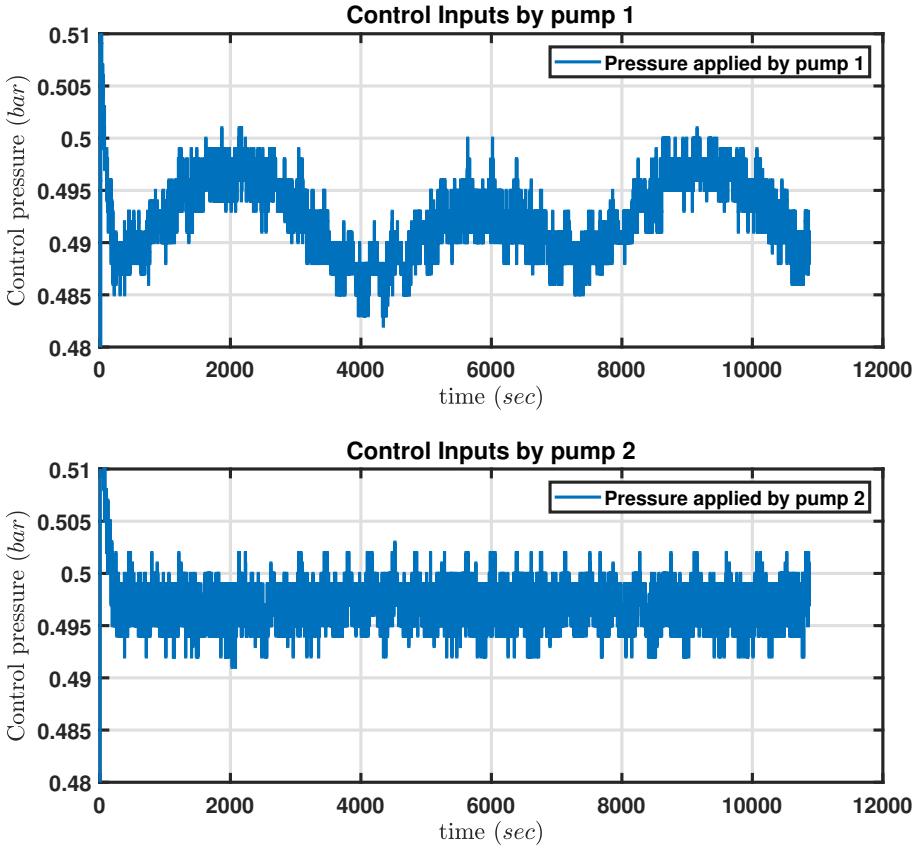


Fig. D.6: The pressure control signal applied by both the pumping stations. Pumping Station 1 applies more pressure when the consumer demand is higher (see fig. D.4) as more water is being taken out of the system by consumers leading to a higher pressure drop. Pumping Station 2 has an almost constant control input on average.

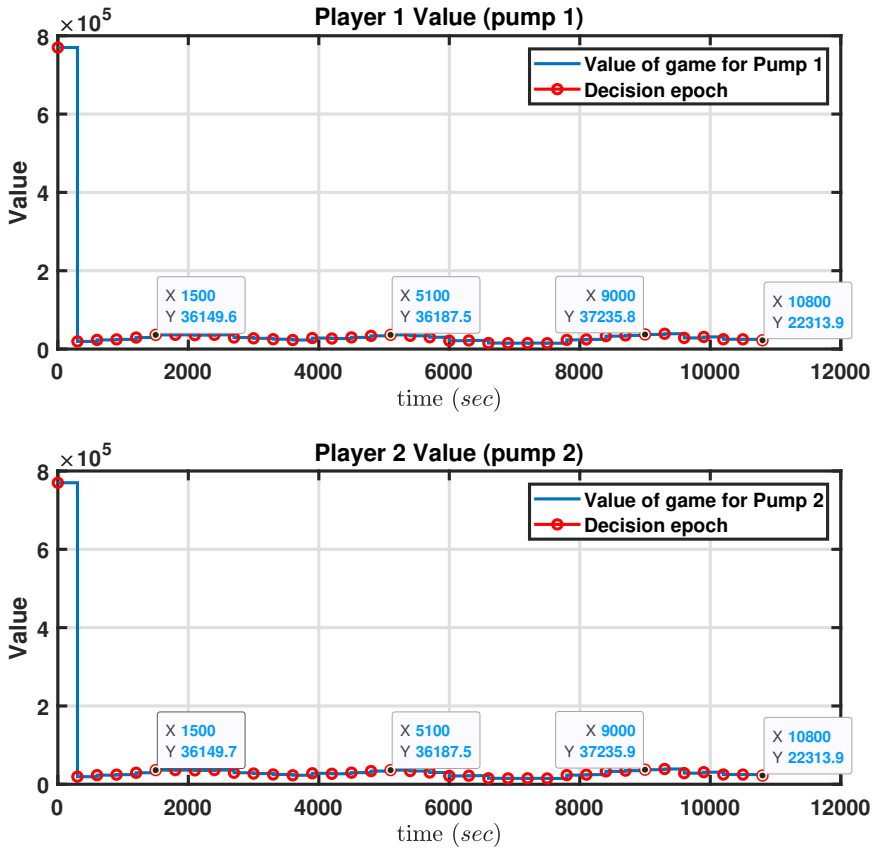


Fig. D.7: The value of the game for each player is almost identical. The red markers indicate the time epochs at which decisions are taken by players.

5 Conclusion

We have presented an Algorithm for Decentralized control of a practical WDN. Game theory and in particular the theory of repeated games is useful for decentralized control of large and complex systems and is sometimes called *engineering agenda* [7]. This work follows the same direction and we hope it inspires more researchers and engineers to use simple control-oriented models for efficient control of large-scale industrial systems. It should be noted that in this work, we calculate minimax optimality which is inefficient

for both the players compared to correlated equilibrium and future research should focus on introducing coordination mechanisms such that both the players converge to a more efficient correlated equilibrium.

6 Acknowledgements

Financial support from the Poul Due Jensen Foundation (Grundfos Foundation) for this research is gratefully acknowledged.

References

[1] J. Thornton and A. Lambert, "Managing pressures to reduce new breaks," *Water*, vol. 21, no. 8, pp. 24–26, 2006.

[2] ——, "Pressure management extends infrastructure life and reduces unnecessary energy costs," in *IWA Conference'Water Loss*, 2007.

[3] V. Puig, C. Ocampo-Martínez, R. Pérez, G. Cembrano, J. Quevedo, and T. Escobet, *Real-time monitoring and operational control of drinking-water systems*. Springer, 2017.

[4] M. M. Polycarpou, J. G. Uber, Z. Wang, F. Shang, and M. Brdys, "Feedback control of water quality," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 68–87, 2002.

[5] C. S. Kallesøe, T. N. Jensen, and J. D. Bendtsen, "Plug-and-play model predictive control for water supply networks with storage," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6582–6587, 2017.

[6] D. Chalchisa, M. Megersa, and A. Beyene, "Assessment of the quality of drinking water in storage tanks and its implication on the safety of urban water supply in developing countries," *Environmental Systems Research*, vol. 6, no. 1, pp. 1–6, 2018.

[7] J. R. Marden and J. S. Shamma, "Game theory and distributed control," in *Handbook of game theory with economic applications*. Elsevier, 2015, vol. 4, pp. 861–899.

[8] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1998.

[9] Y. Shoham, K. Leyton-Brown *et al.*, "Multiagent systems," *Algorithmic, Game-Theoretic, and Logical Foundations*, 2009.

[10] D. Fudenberg and J. Tirole, *Game theory*. MIT press, 1991.

[11] J. Barreiro-Gomez, N. Quijano, and C. Ocampo-Martinez, "Constrained distributed optimization: A population dynamics approach," *Automatica*, vol. 69, pp. 101–116, 2016.

[12] R. Misra, R. Wisniewski, and C. S. Kallesøe, "Approximating solution of stochastic differential games for distributed control of a water network," *IFAC-PapersOnLine*, vol. 55, no. 16, pp. 110–115, 2022.

[13] T. N. Jensen, C. S. Kallesøe, J. D. Bendtse, and R. Wisniewsk, "Plug-and-play commissionable models for water networks with multiple inlets," in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 1–6.

[14] A. Antoniou and W.-S. Lu, *Practical optimization*. Springer, 2007.

[15] J. Val Ledesma, R. Wisniewski, and C. S. Kallesøe, "Smart water infrastructures laboratory: Reconfigurable test-beds for research in water infrastructures management," *Water*, vol. 13, no. 13, p. 1875, 2021.

[16] P. K. Swamee and A. K. Sharma, *Design of water supply pipe networks*. John Wiley & Sons, 2008.

[17] N. Deo, *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2017.

[18] S. S. Rathore, R. Misra, C. S. Kallesøe, and R. Wisniewski, "Leakage diagnosis with a contamination mitigation control framework using a graph theory based model," *Annual Reviews in Control*, 2023.

Paper E

Robust Correlated Equilibrium: Definition and Computation

Rahul Misra, Rafał Wisniewski, Carsten Skovmose Kallesøe, Manuela L. Bujorianu

The paper has been submitted in the IFAC Journal
Automatica

The layout has been revised.

Abstract

We study N-player finite games with costs perturbed due to time-varying disturbances in the underlying system and to that end we propose the concept of Robust Correlated Equilibrium that generalizes the definition of Correlated Equilibrium. Conditions under which the Robust Correlated Equilibrium exists are specified and a decentralized algorithm for learning strategies that are optimal in the sense of Robust Correlated Equilibrium is proposed. The primary contribution of the paper is the convergence analysis of the algorithm and to that end, we propose a modification of the celebrated Blackwell's Approachability theorem to games with costs that are not just time-average as in the original Blackwell's Approachability Theorem but also include time-average of previous algorithm iterates. The designed algorithm is applied to a practical water distribution network with pumps being the controllers and their costs being perturbed by uncertain consumption due to the consumers. Simulation results show that each controller achieves no regret and empirical distributions converge to the Robust Correlated Equilibrium.

Keywords: Game theory; Correlated Equilibrium; Blackwell's Approachability theorem; Robust Control; Adaptive control of multi-agent systems; Optimal control and operation of water resources systems.

1 Introduction

Over the last few decades, the scope of control applications has shifted from optimizing decisions for small-scale systems to large-scale systems with multiple decision-makers [1]. To ensure scalable control design for such large-scale systems, decentralized control schemes are required, and game theory is used to that end [2, 3]. Game theory is a mathematical framework used to study the interactions between multiple decision-makers (also known as players), where a decision made by one player affects not only his/her costs but also the costs of other players. Possible outcomes of the game are characterized by a game-theoretic equilibrium where no player benefits by unilateral deviation of his/her decision. Since the inception of equilibrium concepts such as Nash Equilibrium ($\mathcal{NE}$) [4], it has been argued whether the strategies used by individual rational players will eventually converge to an $\mathcal{NE}$. One challenge with computing a general $\mathcal{NE}$ is that no algorithm can be designed for computing it in *polynomial-time* [5] i.e. the problem scales badly when the number of players or the number of control actions is increased which is a hallmark of large-scale systems. Another fundamental challenge in game theory is *how do players reach an equilibrium?* Fictitious play was introduced as the first algorithm [6] to answer that question. Unfortunately, fictitious play can only be used for zero-sum, cooperative games or games with players having at most 2 actions [7, 8], and there do not exist any simple decentralized algorithms that

ensure convergence to a general $\mathcal{NE}$ [9]. Furthermore, the concept of $\mathcal{NE}$ is incompatible with the Bayesian view of probability, and a player may not choose to follow the $\mathcal{NE}$ strategy after observing the opponent's actions since $\mathcal{NE}$ assumes the worst-case move by opponents [10]. Correlated Equilibrium ($\mathcal{CE}$) which is a convex combination of $\mathcal{NE}$ points is considered to be a more natural alternative to $\mathcal{NE}$ as it assigns subjective probabilities to all possible outcomes of the game [10, 11] (unlike worst-case outcome in $\mathcal{NE}$) as well as is easier to compute as it is a convex polytope [12]. The celebrated Regret matching Algorithms allow decentralized learning of a $\mathcal{CE}$ by a group of agents [13–15]. In this paper, we aim to extend the concept of $\mathcal{CE}$ to games with costs perturbed by disturbances or the so-called time-varying games that have gained a lot of research interest recently [16–18]. To the best of our knowledge, there is a lack of literature on time-varying general games. A related setting is the *contextual games* introduced in [19], where the disturbance can be thought of as a context. However, unlike [19], in this work, the players do not observe the disturbance and we focus on the stricter notion of Robust $\mathcal{CE}$ (instead of coarse-correlated equilibrium in [19]). The motivation behind this work is from practical decentralized optimal control applications where costs could be perturbed by disturbances or unknown dynamics in large-scale cyber-physical systems such as traffic routing, smart grids, pandemics, or energy infrastructure such as water distribution networks [1, 20, 21].

Summary of contributions: 1. We propose a modification of $\mathcal{CE}$ for time-varying games with finite control actions and finite disturbances in the most general uncoupled setting and prove its existence, 2. We propose an algorithm and its convergence analysis for computing the solution, 3. The convergence analysis of the proposed algorithm required a modification of Blackwell's Approachability theorem for the time average of costs as well as the time average of previous iterates of the proposed algorithm, 4. Lastly, we apply our algorithm for decentralized control of a water distribution network.

Paper Outline: We introduce the notation and thereafter introduce the problem setup and our solution concept in Section 3. The existence of the solution is proven in Section 4 followed by an Algorithm with convergence analysis in Section 5. Simulation studies are presented in Section 6 followed by conclusions in Section 7.

2 Notation

- For a finite set A, $\mid A \mid$ is the cardinality of A.
- $\mathbb{P}[\mu]$ represents the probability of event μ occurring.
- $\mathbb{E}_{\mu}$ represents the expectation operator corresponding to the probability measure μ in the subscript.

- $\Delta(A)$ represents a probability simplex in $\mathbb{R}^{|A|}$ where A is a finite set. Formally

$$\Delta(A) := \left\{ x \in \mathbb{R}_+^{|A|} : \sum_{a \in A} x(a) = 1 \right\}.$$

- $[a]_+$ represents nonnegative part of the real number a i.e. $[a]_+ = \max\{a, 0\}$ and for a vector of reals $a = (a_1, \cdots, a_m)$, represents $[a]_+ = ([a_1]_+, \cdots, [a_m]_+)$ and similarly for a matrix A of reals, $[A]_+$ represents matrix consisting of entries $[a_{ij}]_+$.
- $\|a\|_p$ denotes the p-norm of the vector $a \in \mathbb{R}^n$. If no subscript is present, it represents the 2-norm.
- For two vectors $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^n$, $a \cdot b$ represents the standard scalar product between them.
- Given $u = (u^i, u^{-i})$, the notation $\sum_{u \in U : u^i = a} (\cdot)$ is a conditional sum where u^i is fixed as a.

3 Problem setup and formulation

The considered problem setup in this work is similar to [15]. This work will differ from the problem setup considered in [15] in the following ways: 1. The costs will be perturbed by an unknown (possibly adversarial) noise, and therefore, the costs and all associated variables will be time-dependent, 2. Each player can only observe the costs incurred by them (they do not have a cost function for calculating their costs but simply incur a cost at each time instant from the environment) and cannot observe the control actions taken by the other players (each player won't be even aware that they are playing a game and that there are other players i.e. *Unkown Game* setting). We begin by defining a static game Γ.

3.1 Static game preliminaries

Consider a N-player game Γ in strategic (normal) form. The game Γ is defined as a tuple $\Gamma = (N, U, C)$, where

$$U := \prod_{i \in \{1, \cdots, N\}} U^i,$$

is the space of finite joint control actions for all players with U^i being the finite control action space for the player i and $C = \{c^1, \cdots, c^N\}$ are the costs incurred by the players

due to the joint control actions U taken by all the players. Specifically for any player $i \in \{1, \cdots, N\}$, $c^i : U \mapsto \mathbb{R}$. Define U^{-i} as

$$U^{-i} := \prod_{j \in \{1,\cdots,N\}, j \neq i} U^j,$$

which represents the joint control action space of all players except player i. In the sequel, $u \in U$ will represent a joint control action i.e. $u = (u^i, u^{-i})$. Since an equilibrium may not exist in the space of pure control actions but always exists in the space of randomized (or mixed) control actions [4], we define a randomized control policy. A randomized policy for a player $i \in \{1, \cdots, N\}$ is a probability distribution $x^i \in \Delta(U^i)$ over player i's pure control actions U^i. A randomized joint distribution is a probability distribution $x \in \Delta(U)$ over all the player's joint pure control actions U. Given the joint distribution x, we can obtain the marginal distributions $x^i \in \Delta(U^i)$ as $x^i(u^i) = \sum_{u^{-i} \in U^{-i}} x(u^i, u^{-i})$, $\forall u^i \in U^i$ and $x^{-i} \in \Delta(U^{-i})$ as $x^{-i}(u^{-i}) = \sum_{u^i \in U^i} x(u^i, u^{-i}) \quad \forall u^{-i} \in U^{-i}$. The domain of cost functions can be extended multilinearly from joint control action space U to the joint probability distribution $\Delta(U)$ as follows

$$c(x) := \mathbb{E}_x[c] = \sum_{u \in U} c(u) x_u.$$

Similarly, we can write the costs associated with the marginal distributions x^i or x^{-i}.

Definition 3.1 A probability distribution $\psi \in \Delta(U)$ is called a $\mathcal{CE}$ of Γ if it satisfies the following condition for each player $i \in \{1, \cdots, N\}$ and every $a, b \in U^i$,

$$\sum_{u \in U : u^i = a} \psi(u)[c^i(a, u^{-i}) - c^i(b, u^{-i})] \leq 0, \tag{E.1}$$

The $\mathcal{CE}$ condition (E.1) should be interpreted as follows: Suppose a mediator (ψ) for the considered game Γ recommends action a to player $i \in \{1, \cdots, N\}$ i.e. $u^i = a$. Then after observing the recommendation, it is beneficial for player i to follow the recommendation $u^i = a$ i.e. the costs incurred by player i increase if player i deviates from the recommendation $u^i = a$. Specifically, for a two-player game, for every $a, b \in U^1$ and $\beta \in U^2$,

$$\sum_{\beta \in U^2} \psi(a, \beta)[c^1(a, \beta) - c^1(b, \beta)] \leq 0,$$

and for every $e, f \in U^2$ and $\alpha \in U^1$,

$$\sum_{\alpha \in U^1} \psi(\alpha, e)[c^2(\alpha, e) - c^2(\alpha, f)] \leq 0.$$

Thus the distribution ψ is conditioned on the recommendation observed by the player i for any player $i \in \{1, \cdots, N\}$. The set of $\mathcal{CE}$ with cost functions $c : U \to \mathbb{R}^N$ is defined as follows

$$\mathcal{CE}(c) = \left\{ \psi \in \Delta(U) : \sum_{u \in U : u^i = a} \psi(u) c^i(a, u^{-i}) \leq \sum_{u \in U : u^i = a} \psi(u) c^i(b, u^{-i}), \quad \forall i, a, b \right\} \tag{E.2}$$

By Definition 3.1, the correlated equilibrium is robust to any possible unilateral deviation from any player $i \in \{1, \cdots, N\}$ in expectation with respect to the distribution $\psi(u)$. However, what happens if the costs are perturbed due to exogenous disturbances? Is the equilibrium condition (E.1) still robust to perturbations in the cost function itself due to unknown disturbances? This is the primary focus of this paper, and in the following subsection, we define this setting more precisely.

3.2 Static Game with finite Perturbed costs

Suppose that the costs incurred by each player in the Static Game Γ are perturbed due to disturbances. We assume that the number of possible disturbances D is finite and this implies that a finite number of perturbed costs exist such that the disturbance picks one of the costs (from a finite set of costs). Such a Static game will be referred to as a Perturbed Static game in the sequel. Let D be the finite number of costs then formally we can define a N-player Perturbed Static game as $\Gamma' = (N, U, C_D)$ where U is the joint control space, and $C_D : U \to \mathbb{R}^{N \times D}$ maps a control action to a $N \times D$ matrix. Each $c^i : U \to \mathbb{R}^D$ is the N^{th} row of the matrix map C_D with entries c^i_d where $d \in \{1, \cdots, D\}$ (unlike scalar costs in subsection 3.1). We now extend the definition of Correlated equilibrium for Perturbed Static games by introducing the concept of Robust Correlated Equilibrium ($\mathcal{RCE}$).

Definition 3.2 A joint distribution Ψ is called a $\mathcal{RCE}$ for the Perturbed Static Game Γ' if it is an element of the following set

$$\mathcal{RCE} = \left\{ \Psi \in \Delta(U) : \sum_{u \in U : u^i = a} \Psi(u) c^i_d(a, u^{-i}) \leq \sum_{u \in U : u^i = a} \Psi(u) c^i_d(b, u^{-i}), \right.$$
$$\left. \forall i \in \{1, \cdots, N\}, \forall a \in U^i, b \in U^i, \text{ and } \forall d \in \{1, \cdots, D\} \right\}. \tag{E.3}$$

It should be noted that the $\mathcal{RCE}$ condition is a stricter notion compared to the standard Correlated Equilibrium condition (E.2) as

$$\mathcal{RCE} = \bigcap_{d=1}^{D} \mathcal{CE}(c_d(\cdot)),$$

where $c_d : U \to \mathbb{R}^N$ is the D^{th} column of matrix map C_D. Thus, (E.3) is a convex set since it is an intersection of convex sets [22]. More compactly, the $\mathcal{RCE}$ condition can be stated as a probability distribution $\Psi \in \Delta(U)$ such that, for each player $i \in \{1, \cdots, N\}$, having cost vector $c^i : U \to \mathbb{R}^D$ and for every $a, b \in U^i$, the following condition holds

$$\sum_{u \in U : u^i = a} \Psi(u)[c^i_d(a, u^{-i}) - c^i_d(b, u^{-i})] \leq 0, \forall d \in \{1, \cdots, D\}. \tag{E.4}$$

Unlike correlated equilibrium, the existence of $\mathcal{RCE}$ i.e. non-emptiness of (E.3) is non-trivial as shown by the following simple example.

3.3 Motivation for Robustifying $\mathcal{CE}$

We present a simple resource-sharing problem. Consider two farms that use water from a common irrigation channel. Each farm has an automatic controller that directs water from the irrigation channel to the farm. Assume for simplicity, that both the farms continuously require water. The controller can either *open* the valve (action O) or keep it *closed* (action C). Under all conditions, the water in the irrigation channel can supply only one farm at a time. If both farms try to use water from the irrigation channel simultaneously it will be drained for a long time. The cost incurred by the controllers for opening the valve simultaneously is 8 units. The cost incurred by the controller for opening the valve while the other controller's valve is closed is 0 units and the cost incurred for keeping the valve closed while the other farm consumes the water is 5 units. Lastly, if both the controllers do not consume water they pay 1 unit each. The water available in the common irrigation supply depends on the weather conditions (specifically whether there is a drought). Under drought conditions, it becomes more expensive for the farm to not consume water while the other farm consumes it and this is reflected in the costs incurred by the controllers as the cost incurred by a controller for keeping the valve closed, while the other controller opens its valve is increased to 7.5 units form 5 units. The following cost matrices summarize the game. Let $\psi_{CC}, \psi_{CO}, \psi_{OC}$,

$d =$ Normal

Control 1 \ Control 2	C	O
C	$(1, 1)$	$(5, 0)$
O	$(0, 5)$	$(8, 8)$

and ψ_{OO} represent the probabilities of the recommender suggesting the actions $\{C, C\}$, $\{C, O\}$, $\{O, C\}$, and $\{O, O\}$ to controller 1 and 2 respectively. Consider the correlated equilibrium where the public recommendation to both players is as follows,

$$\psi_{CC} = \frac{1}{3}, \quad , \psi_{CO} = \frac{1}{3}, \quad \psi_{OC} = \frac{1}{3}, \quad \psi_{OO} = 0. \tag{E.5}$$

$d =$ Drought

Control 1 \ Control 2	C	O
C	$(1, 1)$	$(7.5, 0)$
O	$(0, 7.5)$	$(8, 8)$

Consider the normal conditions i.e. ($d =$ Normal condition), and suppose that the control 1 is recommended action C, than control 1 can infer that control 2 must have received the recommendation C or O with equal probability and therefore the expected cost for control 1 is,

$$\frac{1}{2}(1) + \frac{1}{2}(5) = 3, \text{ if control 1 follows } C,$$
$$\frac{1}{2}(0) + \frac{1}{2}(8) = 4, \text{ if control 1 switches to action } O,$$

and under recommendation O, the control 1 infers that control 2 must have been recommended to choose C with probability 1, and therefore cost incurred by control 1 is 0 for following the recommendation and 1 unit for not following the recommendation. Thus, the distribution (E.5) is a correlated equilibrium. Now consider the drought conditions i.e. ($d =$ Drought), and suppose that the control 1 is recommended action C, than control 1 can infer that control 2 must have received the recommendation C or O with equal probability and therefore the expected cost for control 1 is,

$$\frac{1}{2}(1) + \frac{1}{2}(7.5) = 4.25, \text{ if control 1 follows } C,$$
$$\frac{1}{2}(0) + \frac{1}{2}(8) = 4, \text{ if control 1 switches to action } O,$$

and the costs remain the same if the recommendation to control 1 was O. Clearly, it is better for control 1 to not follow the recommendation in the case recommendation is C during drought conditions and this demonstrates that the correlated equilibrium distribution (E.5) is not robust to the disturbance d.

4 Existence of $\mathcal{RCE}$

This section focuses on the question of the existence of $\mathcal{RCE}$ for a game. We prove the existence by constructing an auxiliary minimax game with vector payoffs corresponding to each of the disturbances. We construct a mixed strategy for the maximizing player in the auxiliary game using Blackwell's Approachability theorem.

Remark 4.1. *Our proof strategy is similar to [23] but since $\mathcal{RCE}$ (E.4) needs to hold for all d disturbances simultaneously, we use Blackwell's Approachability theorem which in a sense is a generalization of von Neumann's Minimax theorem [24] (although, unlike von Neumann's Minimax theorem, Blackwell's Approachability theorem holds in limit average sense for repeated games and not for static one-shot games [25]).*

We begin by introducing the concepts of *Enforceability* and *Approachability* that will be used in the proof.

Definition 4.1 (Enforceability) Consider a 2-player zero-sum game with scalar payoff $\mathcal{P} : R \times S \to \mathbb{R}$ where player 1 is the minimizer. A payoff p is *p-Enforceable* by player 1, if player 1 has a strategy $q \in \Delta(R)$ such that for any $s \in S$, player 1 can ensure a payoff of $\mathcal{P}(q, s) \leq p$, where $\mathcal{P}(q, s) = \sum_{r \in R} q(r)\mathcal{P}(r, s)$.

Note that *p-Enforceability* can be achieved if p is the value of the saddle point of the game (and the player uses minimax optimal strategy) [15, 26]. Now, consider a 2-player Zero-sum game with **vector payoffs**. Let Player 1 denoted by P1 be the maximizer with control actions in finite set R and Player 2 denoted by P2 be the minimizer with control actions in finite set S. The payoff of this game is denoted by the vector map $\mathcal{P} : R \times S \to \mathbb{R}^D$. Each component $\mathcal{P}_d(r, s)$ of $\mathcal{P}(r, s)$ is determined by the joint actions $(r, s) \in R \times S$ chosen by P1 and P2 respectively. Let us consider a closed convex set $\mathcal{A}$. Define the support function $w_{\mathcal{A}} : \mathbb{R}^D \to \mathbb{R} \cup \{-\infty, \infty\}$ as,

$$\Lambda \mapsto w_{\mathcal{A}}(\Lambda) := \sup\{\Lambda \cdot y : y \in \mathcal{A}\}.$$

Define $\tau \in \mathbb{N}$ and $\mathcal{P}^\tau := \mathcal{P}(r_\tau, s_\tau)$, where (r_τ, s_τ) are the control actions chosen by players at time τ.

Definition 4.2 (Approachability) A set $\mathcal{A} \subset \mathbb{R}^D$ is *Approachable* by P1, if P1 can guarantee the time average of payoff vector $\bar{\mathcal{P}} := (1/t)\sum_{\tau \leq t} \mathcal{P}^\tau$ approaches $\mathcal{A}$ as $t \to \infty$ with probability 1.

We shall now state the Blackwell's Approachability theorem that will be applied to prove the existence of $\mathcal{RCE}$.

Theorem 4.1 (Blackwell's Theorem [24]).
Let $\mathcal{A} \subset \mathbb{R}^D$ be a closed convex set with support function $w_{\mathcal{A}}$. $\mathcal{A}$ is Approachable *by P1 if and only if*

- *for every $\Lambda \in \mathbb{R}^D$ there exists a mixed strategy $q_\Lambda \in \Delta(R)$ such that*

$$\Lambda \cdot \mathcal{P}(q_\Lambda, s) \leq w_{\mathcal{A}}(\Lambda) \quad \textit{for all } s \in S, \tag{E.6}$$

$$\textit{where } \mathcal{P}(q_\Lambda, s) = \mathbb{E}_{q_\Lambda}[\mathcal{P}(r, s)] = \sum_{r \in R} q_\Lambda(r)\mathcal{P}(r, s),$$

with summation being taken component-wise for each component $\mathcal{P}_d(r, s)$ of the vector map $\mathcal{P}$.

- *At time $\tau + 1$, P1 plays the strategy $q_\Lambda(\tau)$ (where $q_\Lambda(\tau)$ is the strategy corresponding to (E.6) for payoff $\mathcal{P}^\tau$) if $\bar{\mathcal{P}} \notin \mathcal{A}$ and plays arbitrarily if $\bar{\mathcal{P}} \in \mathcal{A}$.*

The Blackwell condition (E.6) says that if there exists a strategy q_Λ such that for a static game with scalarized payoffs $\Lambda \cdot \mathcal{P}(q_\Lambda, s)$, the payoff $w_{\mathcal{A}}(\Lambda)$ is $w_{\mathcal{A}}(\Lambda)$-*Enforceable* by the player for any Λ, than the set $\mathcal{A}$ is *Approachable*.

Theorem 4.2.
Every finite perturbed game Γ' has an $\mathcal{RCE}$ as per (E.3).

Proof. — The high-level proof strategy is as follows. Since the set of $\mathcal{RCE}$ is an intersection of finite convex sets $\mathcal{CE}(c_d(\cdot))$, we use the linear duality theorem to show the existence of $\mathcal{RCE}$ or equivalently minimax theorem [22, 25]. Unlike $\mathcal{CE}$, $\mathcal{RCE}$ needs to hold simultaneously for a vector of costs (or payoffs), and therefore, the standard minimax theorem as used in [23] is not sufficient and instead we strive to show the existence in the sense of approachability. We carefully construct an auxiliary zero-sum game corresponding to the considered finite game such that the player 1 in the auxiliary game strives to approach the set of $\mathcal{RCE}$ while player 2 strives to exclude player 1 from the set of $\mathcal{RCE}$. Clearly, if player 1 can find a mixed strategy such that, player 1 approaches the set of $\mathcal{RCE}$ in the auxiliary game, irrespective of player 2's actions, it implies that there exists a mixed strategy for convergence to the set $\mathcal{RCE}$ in the original game. Using Blackwell's Approachability theorem, we construct a mixed strategy in the auxiliary game for player 1 that ensures approachability to the set of $\mathcal{RCE}$. Consider the following auxiliary two-player zero-sum game. P1 chooses an N-tuple of actions $r = (u^1, \cdots, u^N) \in U$, and P2 chooses a triplet $s = (i, a, b)$ where $i \in \{1, \cdots, N\}$, and $a, b \in U^i$ and let $S = \{1, \cdots, N\} \times U^i \times U^i$ be the finite set of all possible triplets which can be chosen by P2. The d^{th} component of payoff from P2 to P1 is defined as follows,

$$\mathcal{P}_d(r, s) = \begin{cases} c_d^i(b, u^{-i}) - c_d^i(u^i, u^{-i}), & \text{if } u^i = a \\ 0, & \text{otherwise.} \end{cases} \tag{E.7}$$

The $\mathcal{RCE}$ of the original game (E.4) corresponds to the mixed strategy for P1 in the auxiliary game (E.7) which guarantees a non-negative time-average of the payoff vector $\bar{\mathcal{P}}$ (note that the sign in (E.7) has been reversed compared to (E.4) since we are working with payoff and not cost in the auxiliary game). More precisely, the $\mathcal{RCE}$ exists if the nonnegative orthant in $\mathbb{R}^D$ is *Approachable* by P1. In the sequel, we shall refer to the nonnegative orthant in $\mathbb{R}^D$ as $\mathcal{S}_+$ where $\mathcal{S}_+ := \{x \in \mathbb{R}^D : x \geq 0\}$. Consider the set $\mathcal{S}_+$ with the support function $w_{\mathcal{S}_+}(\Lambda) = \sup\{\Lambda \cdot s : s \in \mathcal{S}_+\}$ for all $\Lambda \in \mathbb{R}^D$. Given a point $x \in \mathbb{R}^D \backslash \mathcal{S}_+$, let $F(x)$ be the unique point in $\mathcal{S}_+$ that is closest to x in the Euclidean distance and put $\Lambda(x) = x - F(x)$. For $\mathcal{S}_+$ to be approachable by P1, we have

$$w_{\mathcal{S}_+}(\Lambda) = \begin{cases} 0, & \text{if } \Lambda \notin \mathcal{S}_+ \\ \infty, & \text{otherwise.} \end{cases} \tag{E.8}$$

Consider the auxiliary game defined by payoffs (E.7). The $\mathcal{RCE}$ exists if there exists a mixed strategy for P1 such that $\bar{\mathcal{P}}(r, s) \in \mathcal{S}_+$. We shall now construct mixed strategy q_Λ for P1 such that the set $\mathcal{S}_+$ is *Approachable* by P1 by ensuring that condition (E.6) is satisfied by P1. Consider the following cases based on whether, $\mathcal{P}(q_\Lambda, s) \in \mathcal{S}_+$ (i.e. $\Lambda \cdot \mathcal{P}(q_\Lambda, s) \leq w_{\mathcal{S}_+}(\Lambda)$ is $w_{\mathcal{S}_+}(\Lambda)$-*Enforceable*) or not,

- $\mathcal{P}(q_\Lambda, s) \in \mathcal{S}_+$, than by definition $\Lambda > 0$ and $w_{\mathcal{S}_+}(\Lambda) = \infty$. This implies that condition (E.6) is trivially satisfied irrespective of q_Λ.
- $\mathcal{P}(q_\Lambda, s) = 0$, than by definition $\Lambda = 0$ and $w_{\mathcal{S}_+}(\Lambda) = \infty$. This implies that condition (E.6) is trivially satisfied irrespective of q_Λ.
- $\mathcal{P}(q_\Lambda, s) \notin \mathcal{S}_+$, than by definition $\Lambda < 0$ and $w_{\mathcal{S}_+}(\Lambda) = 0$. Thus, in this case, the Blackwell condition (E.6) becomes

$$\sum_{d \in D} \Lambda(d) \cdot \sum_{r \in U} q_\Lambda(r) \mathcal{P}_d(r, s) \leq 0, \tag{E.9}$$

  which is not trivially satisfied. The rest of the proof will focus on finding q_Λ such that (E.9) is satisfied.

Consider the case where $\mathcal{P}(q_\Lambda, s) \notin \mathcal{S}_+$, than we require

$$\sum_{d \in D} \Lambda(d) \cdot \sum_{r \in U} q_\Lambda(r)[c_d^i(b, u^{-i}) - c_d^i(u^i, u^{-i})] \leq 0. \tag{E.10}$$

Recall from (E.7) that P1 can always guarantee payoff 0 if he chooses $u^i \neq a$ (where a is chosen by P2). Thus (E.10) can always be satisfied by P1, if P1's mixed strategy is $q_\Lambda(u^i = a) = 0$ and q_Λ for all other actions is chosen arbitrarily such that $\sum_{r \in R} q_\Lambda(r) = 1$. Therefore the scalarized payoff $\Lambda \cdot \mathcal{P}(q_\Lambda, s) \leq w_{\mathcal{S}_+}(\Lambda)$ is $w_{\mathcal{S}_+}(\Lambda)$-*Enforceable* by P1 for any Λ and therefore, by Theorem 4.1, there exists a strategy $q_\Lambda(\tau)$ for P1 in the Auxiliary game (E.7) such that $\mathcal{S}_+$ is *Approachable* and thus there exists $\mathcal{RCE}$ in the original game.

5 Algorithm for $\mathcal{RCE}$

We have now established the existence of $\mathcal{RCE}$ and this section aims to construct an algorithm that players can use to learn their optimal strategies in the *Unknown Game* setting (i.e. decentralized computation with players unaware of other players' existence in the environment) such that the joint distribution of all the players converges to a $\mathcal{RCE}$. In the sequel, we introduce the learning problem in the *Unknown Game* setting as a repeated game. Thereafter, we introduce the notion of Perturbed conditional Regrets.

5.1 Setting

We begin by formalizing the learning problem as a Repeated Game with Perturbed Costs where the aforementioned Static Game with Perturbed costs is repeated at each time instant: $t = 1, 2, \cdots$. Each player $i \in \{1, \cdots, N\}$ observes the history of play as

$$h_t^i = [u_1^i, c_{d_1}^i, u_2^i, c_{d_2}^i \cdots, u_t^i, c_{d_t}^i],$$

where u_t^i and $c_{d_t}^i$ are defined as the control action taken by player i at time t indicated by the subscript and $c_{d_t}^i$ is the cost incurred by player i at time t indicated by the subscript d_t and further defined as follows. The disturbances d_t should be considered as the random unobservable shocks to the players costs similar to the ones discussed in [8]. However, in contrast to [8], the disturbances are not independent for each player but jointly affect the costs of all the players and are picked from the set $\{1, \cdots, D\}$. As before each player $i \in \{1, \cdots, N\}$ has a randomized policy $x_t^i \in \Delta(U^i)$ based on which control action u_t^i is chosen by the player i and $x_t \in \Delta(U)$ will denote the empirical joint distribution of play (given h_t^i) and is defined as follows.

$$x_t(u) := \frac{1}{t} \mid \{\tau \leq t : u_\tau = u\} \mid . \tag{E.11}$$

Note that the empirical joint distribution is unknown to all the players as players do not observe the control actions taken by other players. Since the costs are affected by the disturbances, we consider *perturbed conditional deviations* which are an extension of conditional deviations defined in [13] and are defined for costs perturbed by disturbance $d_t \in \{1, \cdots, D\}$ at time instant t. The possible perturbed conditional deviation $W^i(a, b)(u_t, d_t)$ for player i is defined as,

$$W^i(a, b)(u_t, d_t) = \begin{cases} c_{d_t}^i(b, u_t^{-i}), & \text{if } u_t^i = a \\ c_{d_t}^i(u_t^i, u_t^{-i}), & \text{otherwise.} \end{cases} \tag{E.12}$$

In contrast to the standard definition of conditional regrets in [13], we consider *perturbed conditional regrets* that are defined for costs perturbed by disturbance $d_t \in \{1, \cdots, D\}$ at time instant t. Since all players fix the history of play, the variables u_t

and d_t are fixed in (E.12) for a given history of play and therefore, in the sequel we have dropped them for notational simplicity. The perturbed conditional regret CR^i_t for using action $a \in U^i$ up to time t is defined as

$$CR^i_t(a,b) = \frac{1}{t}\Big[\sum_{\tau=1}^{t} c^i_{d_\tau}(u^i_\tau, u^{-i}_\tau) - \sum_{\tau=1}^{t} W^i(a,b)\Big],$$

with respect to any action $b \in U^i$. (E.13)

Substituting (E.12) in (E.13) results in

$$CR^i_t(a,b) = \frac{1}{t}\sum_{\tau=1:u^i_\tau=a}^{t} \big[c^i_{d_\tau}(u^i_\tau, u^{-i}_\tau) - c^i_{d_\tau}(b, u^{-i}_\tau)\big],$$

with respect to any action $b \in U^i$. (E.14)

In the sequel, we will consider only the positive part of the perturbed conditional regrets i.e. $CR^i_{t_+} = \max\{CR^i_t, 0\}$. The following theorem implies that if every player i minimizes the positive part of its perturbed conditional regret then the empirical joint distribution of play will approach the set of $\mathcal{RCE}$.

Theorem 5.1.
Define $\epsilon \geq 0$ and suppose at time $t = 1, 2, \cdots$ each player observes their history h^i_t then the corresponding perturbed conditional regret $\limsup_{t\to\infty} CR^i_{t_+}(a,b) \leq \epsilon$ for every player $i \in \{1, \cdots, N\}$ and for every control action pair $a, b \in U^i$ (where $a \neq b$) if and only if the empirical joint distribution of play x_t given by (E.11) converges to the set of $\mathcal{RCE}$ defined by (E.3).

Proof. — Consider the definition of perturbed conditional regret given by (E.14) along with the definition of empirical joint distribution (E.11). Combining them results in

$$CR^i_{t_+}(a,b) = \sum_{u\in U: u^i=a} x_t(u)\big[c^i_{d_t}(u^i, u^{-i}) - c^i_{d_t}(b, u^{-i})\big],$$

with respect to any action $b \in U^i$. (E.15)

Consider the definition of $\mathcal{RCE}$ in(E.4) and for every player $i \in \{1, \cdots, N\}$ and for every control action pair $a, b \in U^i$ (with $a \neq b$), then $x_t(u) \to \Psi \in \mathcal{RCE}$, if for every player $i \in \{1, \cdots, N\}$, $\limsup_{t\to\infty} CR^i_{t_+}(a,b) \leq \epsilon$ as per the definition of $\mathcal{RCE}$. Furthermore, if we substitute $\Psi(u)$ from (E.4) in place of the empirical joint distribution $x_t(u)$ in (E.15) then $CR^i_{t_+}(a,b) \leq \epsilon$ and thus the converse is also true.

As the regrets only make sense given the history of play [27], each player is hindsight rational and that is defined as follows.

Definition 5.1 (Hindsight Rationality) Given the history h_t^i available to player i at time t, player i updates strategy x_{t+1} for $t+1$ instant such that $CR_{t_+}^i \to 0$, as $t \to \infty$ i.e. the player is internally consistent [8].

Remark 5.1. *Similar to Correlated Equilibrium, the Robust Correlated Equilibrium is conditioned on the recommendation action, and therefore, a standard no-regret scheme based on external regret (i.e. comparing the sequence of actions taken against best-fixed action) may not converge to it but instead converge to a robust variant of coarse correlated equilibrium [28]. Convergence to correlated equilibrium requires the minimization of the internal regret matrix* (E.12), (E.14) *(i.e. comparing action taken at time t against best-fixed action at time t for each t and for the entire sequence h_t^i). Blackwell's Approachability theorem is used to show that the minimization of the internal regret matrix by each player implies convergence to the set of correlated equilibria in Theorem A of [13].*

5.2 Algorithm

In this subsection, we present a modification of the regret matching algorithm [13] with perturbed regrets. The motivation behind this modification is to robustify the calculated regrets against disturbances. However, each player i does not observe the disturbance d_t directly but instead observes the costs $c_{d_t}^i$ affected by the disturbances. Therefore, we propose a simple modification that acts like a 'momentum' term [29] and helps reduce variance in regrets due to disturbance as the resulting update is an exponential moving average of past regrets. This smoothens out the effect of perturbations. Thereafter, we apply a regret matching scheme (defined later in this section and in Algorithm 16) to the estimated perturbed conditional regrets. Convergence analysis of the algorithm shows that our proposed scheme converges to the set of $\mathcal{RCE}$. We now introduce the *estimated perturbed conditional regrets* $\widehat{CR}_{t_+}^i$ as follows,

$$\widehat{CR}_{t_+}^i(a,b) = \left(1 - \frac{1}{t}\right)\widehat{CR}_{t-1_+}^i(a,b) + \frac{1}{t}(c_{d_t}^i(a, u_t^{-i}) - c_{d_t}^i(b, u_t^{-i})) + \frac{1}{t}(\widehat{CR}_{t-1_+}^i(a,b) - \widehat{CR}_{t-2_+}^i(a,b)), \quad \text{(E.16)}$$

with a being the control action taken at time $t-1$. The motivation behind adding the term $\widehat{CR}_{t-1_+}^i - \widehat{CR}_{t-2_+}^i$ is to stabilize the perturbed conditional regrets despite perturbations from d. Note that, $\widehat{CR}_{t_+}^i \to 0$ implies $CR_{t_+}^i \to 0$, since we only consider positive

parts of regrets. Furthermore, all the players will simultaneously update their control strategy at each time instant by applying Regret Matching on the estimated perturbed conditional regrets. The regret matching scheme is similar to the one proposed in [13] with the only difference being that the regret matching is applied on $\widehat{CR}^i_{t_+}$ instead of standard regrets in [13]. The regret matching scheme builds a stochastic matrix (i.e. row entries sum up to 1) of size $\mid U^i \mid \times \mid U^i \mid$ that consists of normalized regrets and at each time picks the row corresponding to the action taken at the previous time instant. The row picked is the mixed strategy for the next time instant. Algorithm 16 summarizes this procedure.

5.3 Convergence analysis of Algorithm 16

In this subsection, we shall analyze the convergence of Algorithm 16. Unlike standard no-regret analysis, where it suffices to show that $Regret/t \to 0$, as $t \to \infty$ [19, 28], the estimated $\widehat{CR}^i_{t_+}$ are not just an average of regrets and this is explicitly shown in the equations (E.17), (E.28) where y represents $\widehat{CR}^i_{t_+}$ (simplified notation). Thus, we need to modify Blackwell's Approachability theorem to handle iterates that are not just time-average and this is the main focus of this subsection. The subsection is divided into two parts where in the first part we propose a modification of Blackwell's Approachability theorem and in the second part we apply the extended Approachability theorem to derive transition matrix $\pi(a, b)$ and prove that the strategy update x^i results in convergence of $\widehat{CR}^i_{t_+} \to 0$. By Theorem 5.1, we know that if the perturbed conditional regrets converge to 0 then the empirical joint distribution of play converges to the set of $\mathcal{RCE}$ (provided every player uses a strategy which ensures that their regrets converge to 0) and thus our goal is to show that the estimated perturbed conditional regrets $\widehat{CR}^i_{t_+} \to 0$. Note that we cannot directly use the results from [13, 30] as the estimated perturbed conditional regrets $\widehat{CR}^i_{t_+}$ also considers regret due to deviation from previous estimates. Specifically in [13, 30], the losses (or regrets) are time-averaged, whereas in (E.16) the dynamics of $\widehat{CR}^i_{t_+}$ are not time-averaged losses alone but a combination of the previous iterates of the algorithm along with the time-averaged losses. Therefore, we modify Blackwell's Approachability Theorem for the dynamics (E.16) instead of time-averaged dynamics.

Modifying Blackwell's Approachability theorem

Consider a 2-player perturbed Zero-Sum finite repeated game $\Gamma_2' = (2, U^1, U^2, l, D)$ with losses $l : U^1 \times U^2 \times D \to \mathbb{R}^L$ (with L being the arbitrary dimension of the loss vector) incurred by Player 1. Let $l_{t+1} = l(a_{t+1}, b_{t+1}, d_{t+1})$ represent the bounded loss incurred by player i at time $t+1$ if the players choose actions $(a_{t+1}, b_{t+1}) \in U^1 \times U^2$ with the losses being perturbed by disturbance $d_{t+1} \in \{1, \cdots, D\}$. Consider the following update

Algorithm 16 Perturbed Conditional Regret Matching

Input: Control space U^i, initial strategy x_1^i is uniform distribution over U^i.
Obtain $u_t^i \sim x_t^i$ and get corresponding $c_{d_t}^i(u_t^i, u_t^{-i})$
Let $u_t^i = a$
Set $\alpha_t = \frac{1}{t}$
for $u^i = 1, \cdots, b, \cdots, \mid U^i \mid$ **do**
    Define the vector $q_\pi \in \Delta(U^i)$ as

$$q_\pi(b) = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{otherwise .} \end{cases}$$

    Update estimated perturbed regrets $\widehat{CR}_{t_+}^i$

$$\widehat{CR}_{t_+}^i(a,b) = (1-\alpha_t)\widehat{CR}_{t-1_+}^i(a,b) + \alpha_t(c_{d_t}^i(a, u_t^{-i}) - c_{d_t}^i(b, u_t^{-i})) + \alpha_t(\widehat{CR}_{t-1_+}^i(a,b) - \widehat{CR}_{t-2_+}^i(a,b))$$

end for
Define normalizing constant $\mu = \sum_{b \neq a} \widehat{CR}_{t_+}^i(a,b)$
for $u^i = 1, \cdots, b, \cdots, \mid U^i \mid$ **do**
    Update entries of the transition matrix $\pi_t^i(a,b)$

$$\pi_t^i(a,b) = \begin{cases} \frac{1}{\mu}\widehat{CR}_{t_+}^i(a,b), & \text{if } a \neq b, \\ 1 - \sum_{b' \neq a} \frac{1}{\mu}\widehat{CR}_{t_+}^i(a,b'), & \text{if } a = b, \\ \frac{1}{|U^i|}, & \text{if } \mu = 0 \end{cases}$$

    Update strategy $x_{t+1}^i(b) = q_\pi(a)\pi_t^i(a,b)$
end for

equation,

$$y_{t+1} = \left(1 - \frac{1}{t+1}\right) y_t + \frac{1}{t+1} l_{t+1} + \frac{1}{t+1} \left(y_t - y_{t-1}\right), \tag{E.17}$$

where $t \in \mathbb{N}$, $y_1 = l_1$, $y_0 = 0$, and we change the time index to $t+1$ since y_{-1} is undefined.

Definition 5.2 (Approachability of y_t) The set $\mathcal{A} \subset \mathbb{R}^L$ is Approachable by player 1 if the player 1 has a randomized strategy such that no matter how player 2 plays,

$$\lim_{t\to\infty} \operatorname{dist}\left(y_t, \mathcal{A}\right) = 0, \quad \text{almost surely,} \tag{E.18}$$

where $\operatorname{dist}(\mathbf{j}, \mathcal{A}) = \inf_{\mathbf{k}\in\mathcal{A}} \|\mathbf{j} - \mathbf{k}\|$ is the Euclidean distance of $\mathbf{j}$ from the set $\mathcal{A}$.

Note that Approachability of y_t implies Approachability of $CR^i_{t_+}$ as the form of y_t is same as (E.16) and convergence of y_t implies that the term $y_t - y_{t-1} \approx 0$ and observe that in this case (E.17) is approximately equivalent to

$$y_{t+1} \approx \left(1 - \frac{1}{t+1}\right) y_t + \frac{1}{t+1} l_{t+1},$$

which is same as the positive part of (E.14) (with time index t instead of $t+1$). The following theorem generalizes Blackwell's Approachability theorem for iterates of the form (E.17).

Theorem 5.2.
Consider the game Γ_2' and let y_t be updated by the Player 1 as per (E.17). y_t Approaches to the set $\mathcal{A}$ as per (E.18) if and only if the following conditions are satisfied:

- *For every $\Lambda \in \mathbb{R}^L$ there exists a strategy $q_\Lambda \in \Delta(U^1)$ for Player 1 such that*

$$\Lambda \cdot l(q_\Lambda, b, d) \leq w_{\mathcal{A}}(\Lambda), \ \forall b \in U^2, d \in \{1, \cdots, D\}, \tag{E.19}$$

  where $l(q_\Lambda, b, d) = \mathbb{E}_{q_\Lambda}[l(a, b, d)] = \sum_{a\in U^1} q_\Lambda(a) l(a, b, d)$.

- *At time $t+1$, player 1 plays the strategy $q_\Lambda(t)$, where $q_\Lambda(t)$ is the strategy corresponding to (E.19) for y_t if $y_t \notin \mathcal{A}$, and plays arbitrarily if $y_t \in \mathcal{A}$.*

Proof. — Let the loss incurred at time instant t be denoted by l_t. Since the losses are bounded at each time instant, l_t can be normalized by dividing with the maximum possible loss such that $\|l_t\| \leq 1$ without loss of generality. Let $P(y_t)$ denote the projection of y_t on the set $\mathcal{A}$,

$$P(y_t) = \arg\min_{z \in \mathcal{A}} \|y_t - z\| .$$

Since $\mathcal{A}$ is a convex set, the projection $P(y_t)$ is a unique point on $\mathcal{A}$ [22]. Define Λ_{t-1} as the unit vector pointing in the direction of the set $\mathcal{A}$,

$$\Lambda_{t-1} := \frac{y_{t-1} - P(y_{t-1})}{\|y_{t-1} - P(y_{t-1})\|}. \tag{E.20}$$

The squared distance $\text{dist}^2(y_t, \mathcal{A})$ is calculated as,

$$\text{dist}^2(y_t, \mathcal{A}) = \|y_t - P(y_t)\|^2 \leq \|y_t - P(y_{t-1})\|^2 , \tag{E.21}$$

where the inequality is due to the optimality of the projection operator for each time step. Substituting (E.17) in the last term of the above equation results in,

$$\begin{aligned}
&\|y_t - P(y_{t-1})\|^2 = \left\| \frac{t-1}{t} y_{t-1} + \frac{l_t(a,b)}{t} + \frac{1}{t}(y_{t-1} - y_{t-2}) - P(y_{t-1}) \right\|^2 \\
&= \left\| \frac{t-1}{t} \left(y_{t-1} - P(y_{t-1})\right) + \frac{l_t(a,b) - P(y_{t-1})}{t} + \frac{1}{t}(y_{t-1} - y_{t-2}) \right\|^2 \\
&= \left\| \frac{t-1}{t} \left(y_{t-1} - P(y_{t-1})\right) + \frac{l_t(a,b) - P(y_{t-1})}{t} \right\|^2 + \frac{1}{t^2} \|(y_{t-1} - y_{t-2})\|^2 + \\
&\frac{2}{t^2} \Big[(t-1)(y_{t-1} - P(y_{t-1})) + l_t(a,b) - P(y_{t-1}) \Big] \cdot [y_{t-1} - y_{t-2}] .
\end{aligned} \tag{E.22}$$

The first term in the above equation (in the last step) can be expanded as follows

$$\begin{aligned}
&\left\| \tfrac{t-1}{t} \left(y_{t-1} - P(y_{t-1})\right) + \tfrac{l_t(a,b) - P(y_{t-1})}{t} \right\|^2 = \left(\tfrac{t-1}{t}\right)^2 \|y_{t-1} - P(y_{t-1})\|^2 \\
&+ \tfrac{1}{t^2} \|l_t(a,b) - P(y_{t-1})\|^2 + 2\tfrac{t-1}{t^2} \left[y_{t-1} - P(y_{t-1})\right] \cdot \left[l_t(a,b) - P(y_{t-1})\right]
\end{aligned}$$

Remaining terms of the equation (E.22) are simplified as,

$$\begin{aligned}
&\tfrac{1}{t^2} \|(y_{t-1} - y_{t-2})\|^2 + \tfrac{2}{t^2} \Big[(t-1)(y_{t-1} - P(y_{t-1})) + l_t(a,b) - P(y_{t-1}) \Big] \cdot [y_{t-1} - y_{t-2}] \\
&= \tfrac{1}{t^2} [y_{t-1} - y_{t-2}] \cdot [y_{t-1} - y_{t-2} + 2(t-1)\left(y_{t-1} - P(y_{t-1})\right) + l_t(a,b) - P(y_{t-1})]
\end{aligned}$$

Furthermore as $\|l_t\| \leq 1$, the $\|l_t(a,b) - P(y_{t-1})\|$ will be ≤ 2. Thus the inequality given by (E.21) can be simplified by multiplying by t^2 and rearranging the terms as,

$$t^2 \|y_t - P(y_t)\|^2 - (t-1)^2 \|y_{t-1} - P(y_{t-1})\|^2 \leq \\ 4 + 2(t-1)[y_{t-1} - P(y_{t-1})] \cdot [l_t(a,b) - P(y_{t-1})] + \\ [y_{t-1} - y_{t-2}] \cdot [y_{t-1} - y_{t-2} + 2(t-1)(y_{t-1} - P(y_{t-1})) + l_t(a,b) - P(y_{t-1})]. \quad \text{(E.23)}$$

Summing up both the sides of the inequality (E.23) for time $t = 1, \cdots, T$ results in a telescopic sum of the left-hand side of the inequality (E.23) as $T^2 \|y_T - P(y_T)\|$

$$T^2 \|y_T - P(y_T)\|^2 \leq \sum_{t=1}^{T} 4 + \sum_{t=1}^{T} \Bigg(2(t-1)[y_{t-1} - P(y_{t-1})] \cdot [l_t(a,b) - P(y_{t-1})] \\ + [y_{t-1} - y_{t-2}] \cdot [y_{t-1} - y_{t-2} + 2(t-1)(y_{t-1} - P(y_{t-1})) + l_t(a,b) - P(y_{t-1})] \Bigg), \quad \text{(E.24)}$$

and on the right-hand side the term $y_{t-1} - y_{t-2}$ telescopes to y_{T-1}. Denote $(t-1)\|y_{t-1} - P(y_{t-1})\|$ as K_{t-1} and dividing both sides of (E.24) by T^2 results in

$$\|y_T - P(y_T)\|^2 \leq \frac{4}{T} + \frac{2}{T} \sum_{t=1}^{T} K_{t-1}\Lambda_{t-1}[l_t(a,b) - P(y_{t-1})] + \\ \frac{y_{T-1}}{T} \left[\frac{y_{T-1}}{T} + \frac{2}{T} \sum_{t=1}^{T} \Big(K_{t-1}\Lambda_{t-1} + l_t(a,b) - P(y_{t-1}) \Big) \right]. \quad \text{(E.25)}$$

Using the definition of the support function given by (E.8) and (E.20), the condition given by (E.19) can be written as

$$\Lambda_{t-1} \cdot l_t(q_{\Lambda_{t-1}}, b) \leq \Lambda_{t-1} \cdot P(y_{t-1}), \ \forall b \in U^2, \ \forall d \in \{1, \cdots, D\}. \quad \text{(E.26)}$$

Substituting (E.26) in (E.25) leads to

$$\|y_T - P(y_T)\|^2 \leq \frac{4}{T} + \frac{2}{T} \sum_{t=1}^{T} K_{t-1}\Lambda_{t-1}[l_t(a,b) - l_t(a, q_{\Lambda_{t-1}})] + \\ \frac{y_{T-1}^2}{T^2} + \left[\frac{2y_{T-1}}{T^2} \sum_{t=1}^{T} \Big(K_{t-1}\Lambda_{t-1} + l_t(a,b) - P(y_{t-1}) \Big) \right]. \quad \text{(E.27)}$$

The terms K_{t-1} and $l_t(a,b) - P(y_{t-1})$ are bounded between 0 and 2 whereas the term $l_t(a,b) - l_t(q_{\Lambda_t}, b)$ is a Martingale Difference Sequence (see definition A.1 in Appendix) with respect to the random variable a_t which is sampled from the distribution $q_{\Lambda_{t-1}}$ i.e.

$$\mathbb{E}_{q_{\Lambda_{t-1}}}[l_t(a_t, b) - l_t(q_{\Lambda_t}, b) \mid a_1, \cdots, a_t] = 0,$$

holds almost surely. Convergence of remaining terms of (E.27) is studied in the following Lemma.

Lemma 5.1.
The component-wise value of y_T generated by (E.17) is bounded by $\mathbb{E}_{q_\Lambda}[l_t(a,b)]$ almost surely.

Proof. — Recall that each component of the vector $y_t \in \mathbb{R}_+$ and y_t is updated as per (E.17),

$$\begin{aligned} y_{t+1} =& \left(1 - \frac{1}{t+1}\right) y_t + \frac{1}{t+1} l_{t+1} + \frac{1}{t+1}\left(y_t - y_{t-1}\right), \\ =& y_t - \frac{1}{t+1} y_{t-1} + \frac{1}{t+1} l_{t+1}, \end{aligned} \tag{E.28}$$

where $t \in \mathbb{N}$, $y_1 = l_1$, $y_0 = 0$. The outline of the proof is as follows. In the sequel, all the analysis will be component-wise for the vector y_t. The analysis of sequence (y_t) generated by (E.28) is separated into analysis of the sequence generated by an autonomous component of (E.28) denoted by $\tilde{y}_{t+1}$ and the sequence generated by (E.28) with input loss l_{t+1}. Firstly, we consider the sequence generated by the autonomous component of the equation (E.28) i.e. (E.28) without input loss l_{t+1} (defined more precisely in (E.29)). We observe that the sequence $(\tilde{y}_t)$ is decreasing and bounded from below by 0, so it is convergent. Then we prove by contradiction that its limit is 0. Thereafter, we show that the sequence (y_t) generated by the non-autonomous system will be bounded by $\mathbb{E}_{q_\Lambda}[l_{t+1}]$. Consider the autonomous component of the equation (E.28) (everything except $(1/(t+1))l_{t+1}$ in (E.28)) as follows

$$y_{t+1} = y_t - \frac{1}{t+1} y_{t-1}. \tag{E.29}$$

We claim that the sequence generated by (E.29), $\tilde{y}_\tau \to 0$ for some finite time τ. The proof of this claim is as follows. Suppose that the sequence generated by (E.29) has a positive limit. Specifically for any $\epsilon > 0$, assume that there exists $\tau := \tau(\epsilon)$ such that $\tilde{y}_\tau > \epsilon$. Than from (E.29),

$$\tilde{y}_\tau = \tilde{y}_{\tau-1} - \alpha_\tau \tilde{y}_{\tau-2}, \quad \text{where } \alpha_\tau = \frac{1}{\tau}.$$

$$\text{If } \tilde{y}_\tau > \epsilon \text{ than } \tilde{y}_{\tau-1} > \epsilon + \alpha_\tau \tilde{y}_{\tau-2}.$$

Since $\tilde{y}_\tau \in \mathbb{R}_+$, $\forall \tau$, we can iterate backwards the previous inequality as

$$\begin{aligned} &\text{If } \tilde{y}_{\tau-1} > \epsilon \text{ than } \tilde{y}_{\tau-2} > \epsilon + \alpha_\tau \tilde{y}_{\tau-3}, \\ &\quad \cdots \qquad \cdots \\ &\qquad\qquad \tilde{y}_0 > \epsilon. \end{aligned}$$

But $\tilde{y}_0$ cannot be greater than ϵ since we initialize (E.28) in Algorithm 16 with $\tilde{y}_0 = 0$ (contradiction). Since this is true for any $\epsilon > 0$ and $\tilde{y}_\tau \in \mathbb{R}_+$, $\forall \tau$, we can conclude that $\tilde{y}_\tau \to 0$ in some finite time τ. Now consider the non-autonomous system given by (E.28) and modify the equation as follows

$$y_{t+1} = y_t - \frac{1}{t+1} y_{t-1} + \frac{1}{t+1}(l_{t+1} - \mathbb{E}_{q_\Lambda}[l_{t+1}]) + \frac{1}{t+1}\mathbb{E}_{q_\Lambda}[l_{t+1}].$$

Recall that $\mathbb{E}_{q_\Lambda}[l_{t+1}(a,b)] = l(q_\Lambda, b)$ as defined in (E.19). Note that the term $l_{t+1} - \mathbb{E}_{q_\Lambda}[l_{t+1}]$ is a Martingale Difference Sequence. Therefore we can conclude $y_T \leq \mathbb{E}_{q_\Lambda}[l_{t+1}]$ almost surely.

Thus all the terms in (E.27) are either bounded or are a Martingale Difference Sequence. Let the Martingale Difference Sequence generated by $\|y_T - P(y_T)\|^2$ be bounded by constants $0 < k_1, k_2, \cdots k_t, \cdots < \infty$, then for any $m > 0$, we have the following concentration inequality due to Azuma–Hoeffding inequality (see Theorem A.1 in Appendix) we have

$$\mathbb{P}[\|y_T - P(y_T)\|^2 > m] \leq \exp\left(\frac{-2m^2}{\sum_{t=1}^{T} k_t^2}\right) < \infty,$$

and thus by Borel-Cantelli Lemma (see Lemma A.1 in Appendix), we can conclude that for sufficiently large T, $\|y_T - P(y_T)\|^2 \to 0$ almost surely.

Convergence of Algorithm 16 using Theorem 5.2

Proposition 5.1.
The estimated perturbed conditional regret $\widehat{CR}^i$ and the corresponding perturbed conditional regret CR^i in Algorithm 16 converges to 0 for all actions $a, b \in U^i$ and for all players $i \in N$ if all the players use Algorithm 16 in a decentralized manner.

Proof. — Consider the static game at time t and fix the control actions $u_t = [u_t^i, u_t^{-i}]$ taken by all players, and fix the disturbance d_t experienced by each of the players. Then an auxiliary stage game is defined as follows. Let l_t^i denote the perturbed conditional

regret incurred by player i at time t (i.e. regret per time instant for not choosing control action b instead of control action a) as

$$l_t^i(u_t, d_t)(a,b) := \begin{cases} c_{d_t}^i(a, u_t^{-i}) - c_{d_t}^i(b, u_t^{-i}), & \text{if } u_t^i = a \\ 0, & \text{otherwise.} \end{cases} \tag{E.30}$$

Consider Theorem 5.2 with (E.17). We consider the Approachability of Auxiliary Game loss (E.30) to the set $\mathcal{S}_- := \{x \in \mathbb{R}^L : x \leq 0\}$, where L is given as follows,

$$L := \{(a,b) \in U^i \times U^i : a \neq b\}. \tag{E.31}$$

Consider y_t (recall that it represents $\widehat{CR}^i$ in (E.16)) as per (E.17) and note that $P(y_t) = 0$, where $P(y_t)$ is the projection of y_t on $\mathcal{S}_-$. Define

$$\pi_t := \frac{y_t}{\|y_t\|_1}, \quad \text{where } \|y_t\|_1 \text{ is 1-norm of } y_t,$$

The condition (E.19) can now be written as

$$\pi_t(a,b) \cdot l_t^i(a, q_{\pi_t}) \leq 0 \ \forall a \in U^i, \ \forall d \in \{1, \cdots D\},$$
$$\pi_t(a,b) q_{\pi_t}(b)[c_{d_t}^i(a, u_t^{-i}) - c_{d_t}^i(b, u_t^{-i})] \leq 0, \ \forall a \in U^i, \ \forall d \in \{1, \cdots D\}.$$

Choosing $q_\pi \in \Delta(U^i)$ as

$$q_\pi(b) = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{otherwise} \end{cases}, \quad \forall b \in |\ U^i\ |,$$

results in the condition (E.19) being satisfied for all $a \in U^i$ and for all $d \in \{1, \cdots D\}$ as an equality.

6 Simulation studies

In this section, we apply the algorithm to a decentralized control problem for a water distribution network with pumping stations acting as controllers. The goal of each decentralized controller is to ensure tracking of the desired pressure set point in the system while minimizing its own energy consumption. The decentralized optimal control problem is described precisely in our previous work [21], where we propose model-based minimax strategies for each player and implement them on a laboratory setup that emulates a real-life water distribution network. Firstly, we present the estimated perturbed regrets $\widehat{CR}_{t_+}^i$ calculated by each player using Algorithm 16 on semi-logarithmic plots in Fig. E.1 and Fig. E.2. The x-axis is on the logarithmic scale while the y-axis is

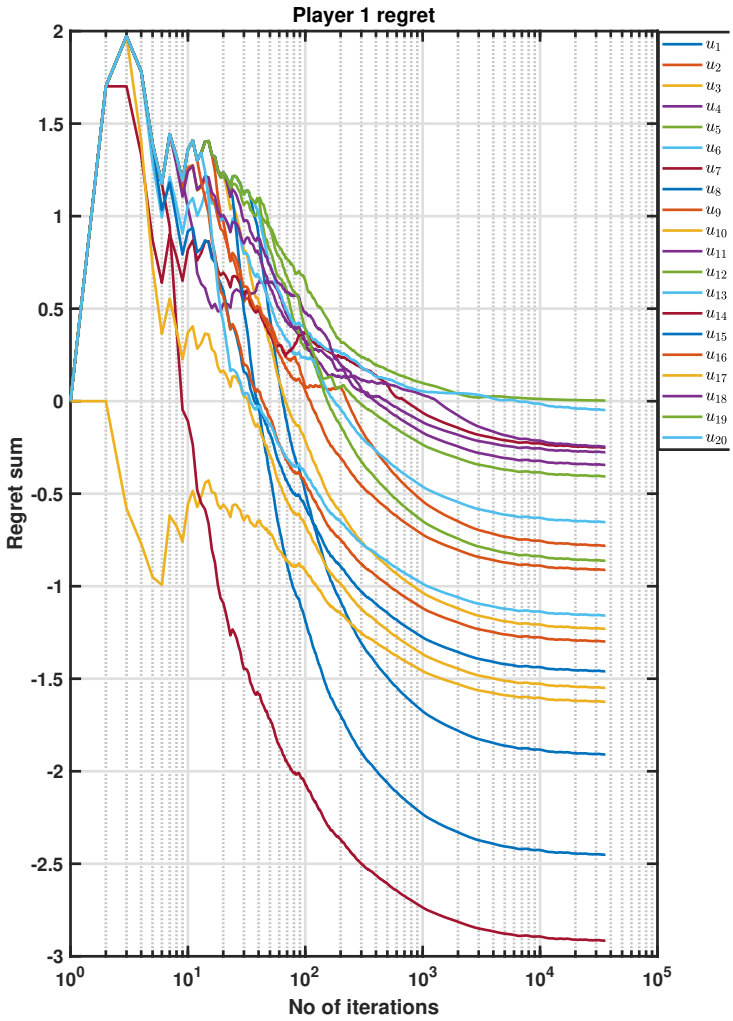


Fig. E.1: Regrets experienced by Player 1.

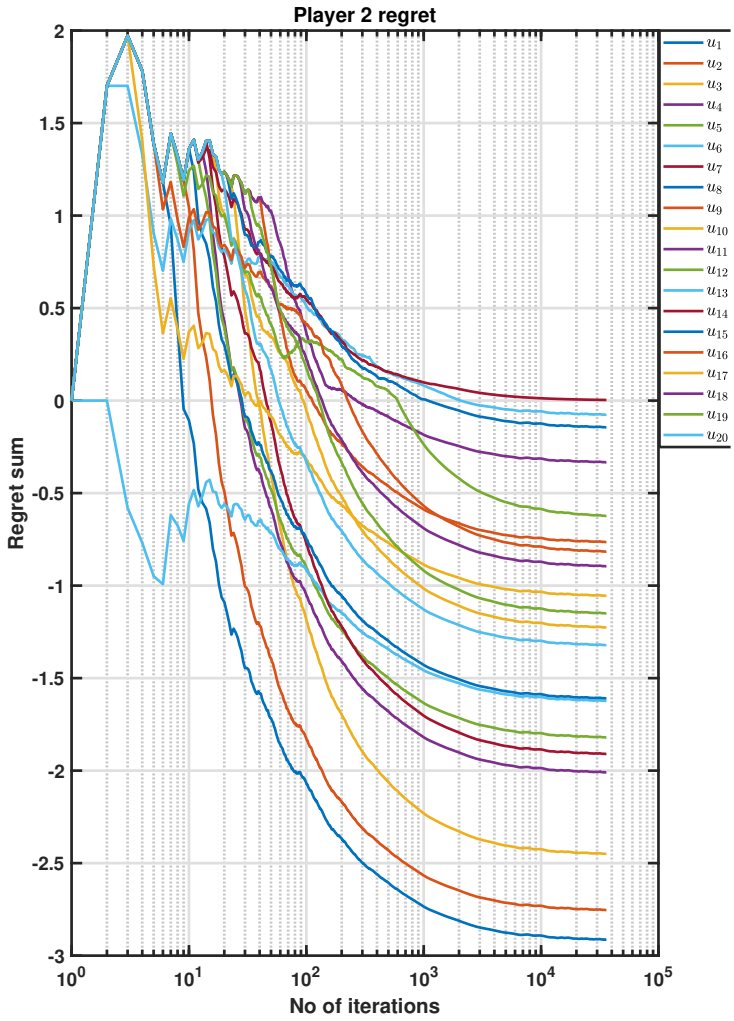


Fig. E.2: Regrets experienced by Player 2.

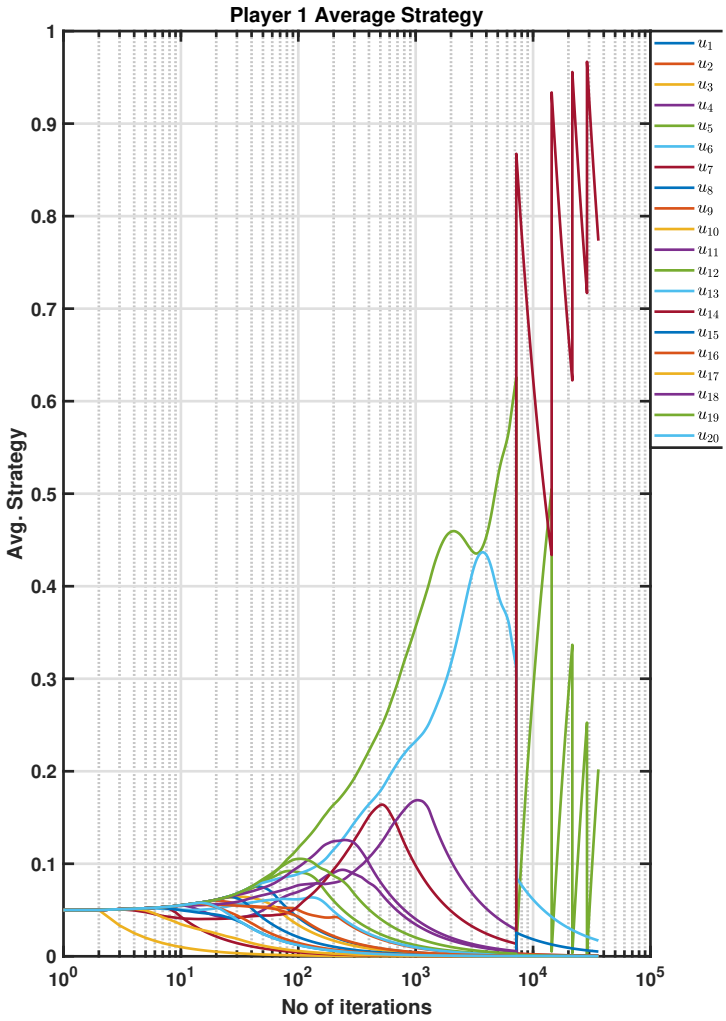


Fig. E.3: Averaged empirical mixed strategy for Player 1.

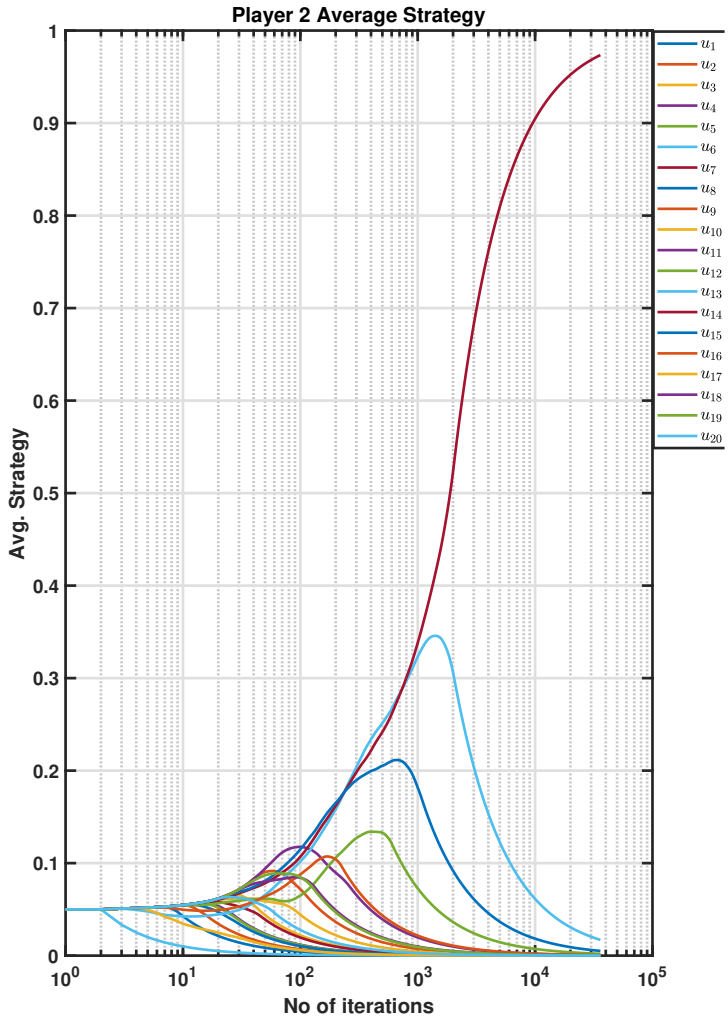


Fig. E.4: Averaged empirical mixed strategy for Player 2.

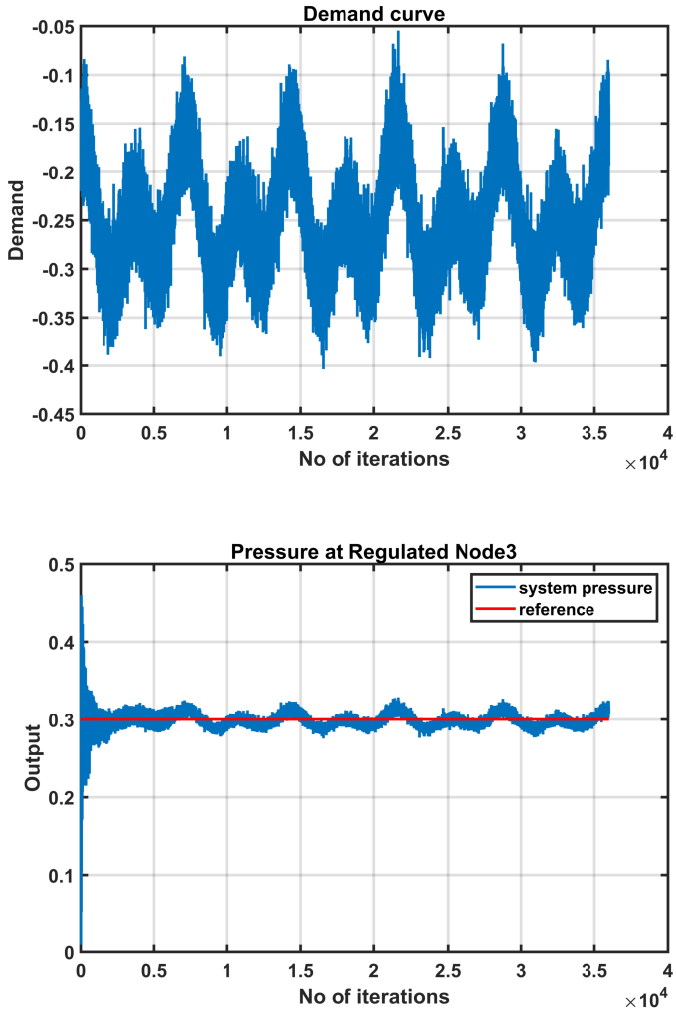


Fig. E.5: Pressure is regulated despite consumer disturbances.

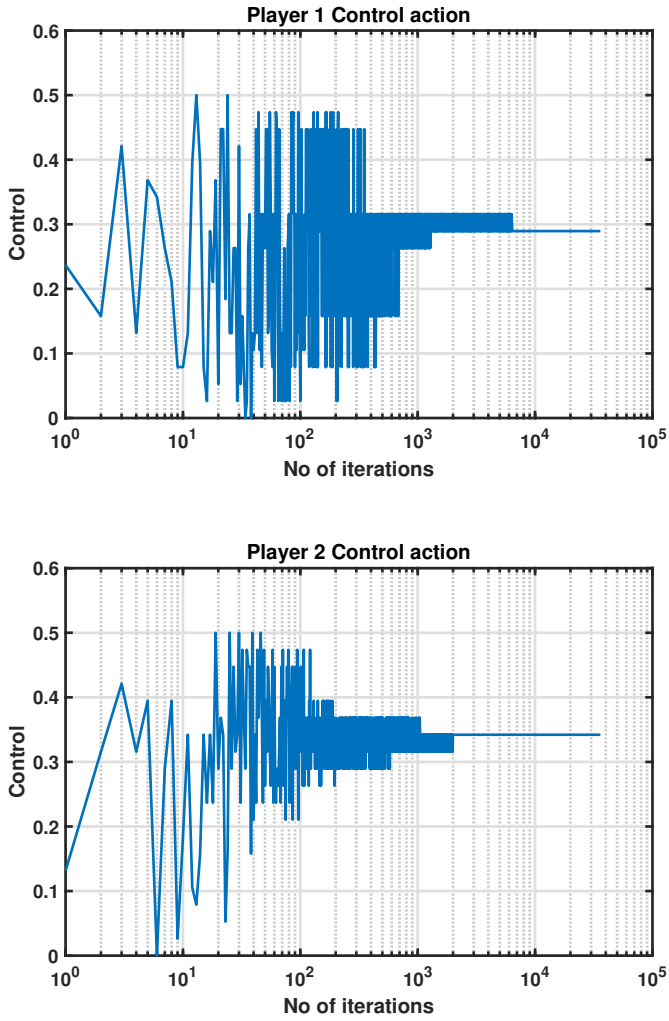


Fig. E.6: Realized control actions by both the players.

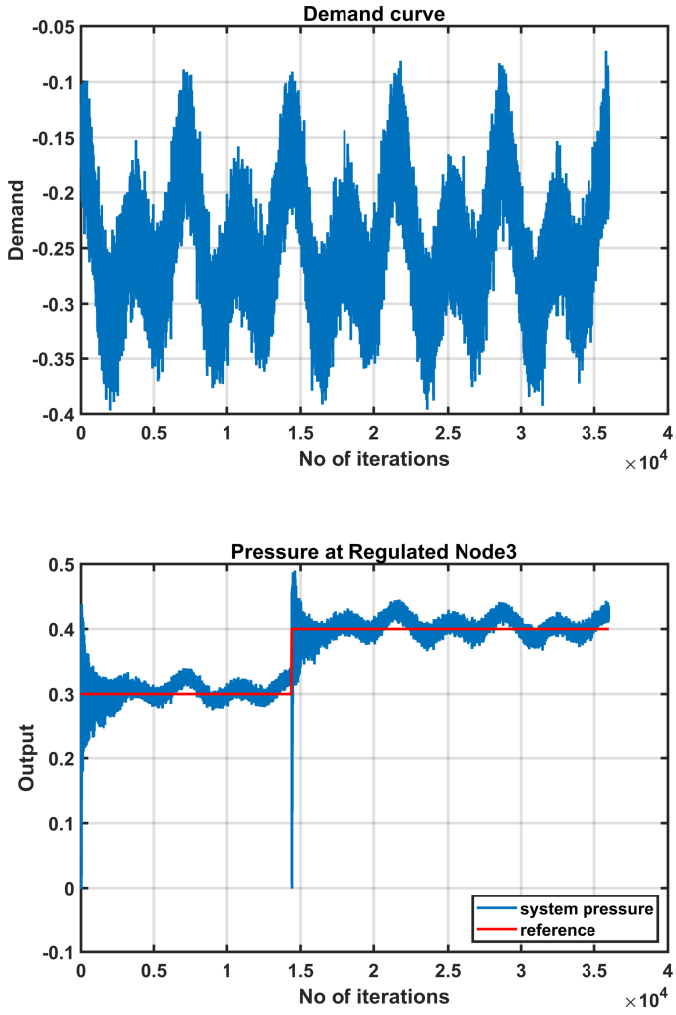


Fig. E.7: The players handle changing set points satisfactorily.

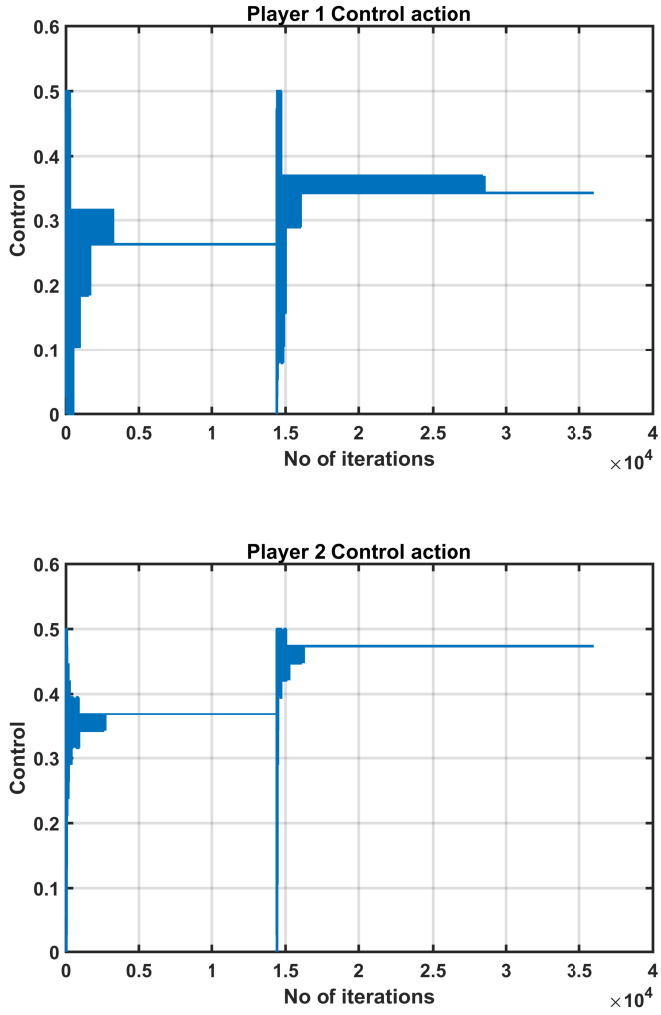


Fig. E.8: Realized control actions with change in the set point.

on a regular scale. Next, we present the average of the empirical mixed strategy of the players (which jointly converge to the $\mathcal{RCE}$) and the realized control actions on semi-logarithmic plots in Fig. E.3, fig. E.4 and Fig. E.6. Simulation results show that the controllers calculate an optimal solution in the sense of a $\mathcal{RCE}$ while being game agnostic i.e. they are unaware of the existence of the other controllers and the dynamics of the system. The results presented here are an improvement on the work presented in [21] as our solution is more efficient for individual players and better regulation of pressure is achieved as shown in Fig. E.5. Further experiments demonstrate that our algorithm can also handle changing set points (by re-initializing if it detects a change in the set point). This is shown in the plots in Fig. E.7 and Fig. E.8.

7 Conclusion and Future Work

We have successfully extended the concept of $\mathcal{CE}$ to time-varying games by introducing $\mathcal{RCE}$ and a decentralized algorithm that each player can use to learn $\mathcal{RCE}$ in the Unkown game setting. The convergence analysis of the algorithm and simulation studies validate our approach. We conjecture that Theorem 4.2 can be extended for games with continuous but bounded disturbances as per the simulation results and a proof for bounded disturbances is an interesting topic for future work.

Acknowledgements

Financial support from the Poul Due Jensen Foundation for this research is gratefully acknowledged.

References

[1] A. M. Annaswamy, K. H. Johansson, G. J. Pappas *et al.*, "Control for societal-scale challenges: Road map 2030," *IEEE Control Systems Society Publication: Piscataway, NJ, USA*, 2023.

[2] E. Mageirou and Y.-C. Ho, "Decentralized stabilization via game theoretic methods," *Automatica*, vol. 13, no. 4, pp. 393–399, 1977.

[3] J. R. Marden and J. S. Shamma, "Game theory and control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 105–134, 2018.

[4] J. Nash, "Non-cooperative games," *Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951. [Online]. Available: http://www.jstor.org/stable/1969529

[5] Y. Shoham, K. Leyton-Brown *et al.*, "Multiagent systems," *Algorithmic, Game-Theoretic, and Logical Foundations*, 2009.

[6] G. W. Brown, "Iterative solution of games by fictitious play," *Act. Anal. Prod Allocation*, vol. 13, no. 1, p. 374, 1951.

[7] M. Benaım and M. W. Hirsch, "Mixed equilibria and dynamical systems arising from fictitious play in perturbed games," *Games and Economic Behavior*, vol. 29, no. 1-2, pp. 36–72, 1999.

[8] D. Fudenberg and D. K. Levine, *The theory of learning in games*. MIT Press, 1998, vol. 2.

[9] S. Hart and A. Mas-Colell, "Uncoupled dynamics do not lead to nash equilibrium," *American Economic Review*, vol. 93, no. 5, pp. 1830–1836, 2003.

[10] R. J. Aumann, "Correlated equilibrium as an expression of bayesian rationality," *Econometrica: Journal of the Econometric Society*, pp. 1–18, 1987.

[11] ——, "Subjectivity and correlation in randomized strategies," *Journal of mathematical Economics*, vol. 1, no. 1, pp. 67–96, 1974.

[12] C. Papadimitriou and T. Roughgarden, "Computing correlated equilibria in multi-player games," *Journal of the ACM (JACM)*, vol. 55, no. 3, pp. 1–29, 2008.

[13] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[14] ——, *A reinforcement procedure leading to correlated equilibrium*. Springer, 2001.

[15] ——, *Simple adaptive strategies: from regret-matching to uncoupled dynamics*. World Scientific, 2013, vol. 4.

[16] M. Zhang, P. Zhao, H. Luo, and Z.-H. Zhou, "No-regret learning in time-varying zero-sum games," in *International Conference on Machine Learning*. PMLR, 2022, pp. 26 772–26 808.

[17] Z. Wang, Y. Shen, M. M. Zavlanos, and K. H. Johansson, "Convergence analysis of the best response algorithm for time-varying games," *arXiv preprint arXiv:2309.00307*, 2023.

[18] Y. Feng, H. Fu, Q. Hu, P. Li, I. Panageas, B. Peng, and X. Wang, "On the last-iterate convergence in time-varying zero-sum games: Extra gradient succeeds where optimism fails," *arXiv preprint arXiv:2310.02604*, 2023.

[19] P. G. Sessa, I. Bogunovic, A. Krause, and M. Kamgarpour, "Contextual games: Multi-agent learning with side information," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 912–21 922, 2020.

[20] O. Karaca, P. G. Sessa, A. Leidi, and M. Kamgarpour, "No-regret learning from partially observed data in repeated auctions," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14–19, 2020.

[21] R. Misra, C. S. Kallesøe, and R. Wisniewski, "Decentralized control of a water distribution network using repeated games," in *2023 27th International Conference on Methods and Models in Automation and Robotics (MMAR)*. IEEE, 2023, pp. 181–186.

[22] S. Boyd and L. Vandenberghe, "Convex optimization, cambridge univ," *Press, UK*, 2004.

[23] S. Hart and D. Schmeidler, "Existence of correlated equilibria," *Mathematics of Operations Research*, vol. 14, no. 1, pp. 18–25, 1989.

[24] D. Blackwell, "An analog of the minimax theorem for vector payoffs." *Pacific Journal of Mathematics*, vol. 6, no. 4, pp. 1–8, 1956.

[25] J. Abernethy, P. L. Bartlett, and E. Hazan, "Blackwell approachability and no-regret learning are equivalent," in *Proceedings of the 24th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, 2011, pp. 27–46.

[26] J. Von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton university press, 1944.

[27] V. Perchet, "Approachability, regret and calibration: Implications and equivalences," *Journal of Dynamics and Games*, vol. 1, no. 2, pp. 181–254, 2014.

[28] T. Roughgarden, "Cs364a: Algorithmic game theory lecture# 17: No-regret dynamics," 2013.

[29] V. Borkar, *Stochastic Approximation: A Dynamical Systems Viewpoint: Second Edition*, ser. Texts and Readings in Mathematics. Hindustan Book Agency, 2022. [Online]. Available: https://books.google.dk/books?id=k_ChEAAAQBAJ

[30] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

A Some useful results from Probability theory

Consider a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ where Ω is the sample space, $\mathcal{F}$ is the increasing σ-algebra and $\mathbb{P}$ is the probability measure on the measurable space $(\Omega, \mathcal{F})$. We can now define the Martingale Difference Sequences.

Definition A.1 (Martingale Difference Sequence) A sequence of random variables $X_1, X_2, \cdots$ is a Martingale Difference sequence with respect to the filtration $\{\mathcal{F}_0, \mathcal{F}_1, \cdots\}$ if for every $t > 0$, X_t is $\mathcal{F}_t$-measurable and

$$\mathbb{E}_{\mathbb{P}}[X_{t+1} \mid \mathcal{F}_t] = 0, \text{ almost surely.}$$

We now state the Azuma–Hoeffding inequality which is essentially a concentration inequality that helps in deriving probabilistic bounds for a random process.

Theorem A.1 (Azuma–Hoeffding inequality).
Let $X_0, X_1, \cdots$ be a Martingale Difference Sequence with respect to the filtration $\{\mathcal{F}_0, \mathcal{F}_1, \cdots\}$ with $X_t \in [Z_t, Z_t + c_t]$ for some random variable Z_t which is also measurable with respect to the filtration $\{\mathcal{F}_0, \cdots, \mathcal{F}_{t-1}\}$ and constants $0 < c_1, c_2, \cdots c_t, \cdots < \infty$. Let $S_t = \sum_{t=1}^{T} X_t$, than for any $m > 0$ we have the following inequality

$$\mathbb{P}[S_t > m] \leq exp\left(\frac{-2m^2}{\sum_{t=1}^{T} c_t^2}\right)$$

We now state the Borel–Cantelli Lemma which reasons about the probability of events occurring infinitely often.

Lemma A.1 (Borel–Cantelli Lemma).
Let $E_1, E_2, \cdots$ be a sequence of events in some probability space. Then if

$$\sum_{t=1}^{\infty} \mathbb{P}[E_t] < \infty,$$

then the probability that infinitely many of them occur is

$$\mathbb{P}[\limsup_{t \to \infty} E_t] = 0.$$

This dissertation explores how learning-based adaptive controllers can be designed to operate in environments with other controllers, meeting multiple objectives while remaining robust to uncertainties and disturbances. The central hypothesis is that optimal control for large, complex systems can be broken down into a series of decentralized optimization problems, solved independently by each controller at each time step. Under appropriate assumptions, these controllers can learn solutions that ensure system safety, optimality, and resilience to disturbances.