Machine Learning Approaches to Multi-Agent Inverse Learning Problems

---

A Dissertation

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment
of the requirements for the degree

Doctor of Philosophy

by

Xiaomin Lin

December 2017

APPROVAL SHEET

This Dissertation
is submitted in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Author Signature: Xiaomin Lin

This Dissertation has been read and approved by the examining committee:

Advisor: Peter Beling

Committee Member: Zongli Lin

Committee Member: Matthew Gerber

Committee Member: Laura Barnes

Committee Member: Nicola Bezzo

Committee Member: Stephen Adams

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, School of Engineering and Applied Science

December 2017

UNIVERSITY OF VIRGINIA

DOCTORAL THESIS

Machine Learning Approaches to Multi-Agent Inverse Learning Problems

---

Author:

Xiaomin LIN

Supervisor:

Dr. Peter A. BELING

A dissertation submitted in fulfillment of the requirements

for the degree of Doctor of Philosophy

to the

Department of Systems and Information Engineering

School of Engineering and Applied Science

December 11, 2017

Approved by the Guidance Committee:

Peter Beling
Professor, SIE, Advisor

Date

Matthew Gerber
Assistant Professor, SIE, Committee Chair

Date

Zongli Lin
Professor, ECE, Committee Member

Date

Laura Barnes
Assistant Professor, SIE, Committee Member

Date

Nicola Bezzo
Assistant Professor, SIE, Committee Member

Date

Stephen Adams
Senior Scientist, SIE, Committee Member

Date

UNIVERSITY OF VIRGINIA

Abstract

Department of Systems and Information Engineering

School of Engineering and Applied Science

Doctor of Philosophy

Machine Learning Approaches to Multi-Agent Inverse Learning Problems

by Xiaomin LIN

The problem to infer the goals of an agent on the basis of the observation of its actions has been framed in the context of inverse reinforcement learning (IRL) and has been extensively studied in recent decades. However, this model is valid only when no other adaptive agents exist or their interference can be neglected. Otherwise, a new model taking other agents into account needs to be created in place of IRL. To this end, this dissertation proposes a multi-agent inverse reinforcement learning (MIRL) model, using the framework of stochastic games, which generalize Markov decision processes to game theoretic scenarios. We develop algorithms for two fundamental classes of MIRL problems: two-agent zero-sum and two-agent general-sum. For the first class, we develop a Bayesian solution approach in which the generative model is based on an assumption that the two agents follow a minimax bi-policy. For the second, we consider five variants: *uCS-MIRL*, *advE-MIRL*, *cooE-MIRL*, *uCE-MIRL*, and *uNE-MIRL*, each distinguished by its solution concept. Problem uCS-MIRL is a cooperative game in which the agents employ cooperative strategies that aim to maximize the total game value. In problem uCE-MIRL, agents are assumed to follow strategies that constitute a correlated equilibrium while maximizing total game value. The

uNE-MIRL is similar to uCE-MIRL in total game value maximization but a Nash equilibrium is assumed to employ. The advE-MIRL and cooE-MIRL problems assume agents constitute an adversarial equilibrium and coordination equilibrium, respectively. We propose novel approaches to address these five problems under the assumption that the game observer either knows or is able to accurately estimate the policies and solution concepts for players. For uCS-MIRL, we first develop a characteristic set of solutions ensuring that the observed bi-policy is a uCS and then apply a Bayesian inverse learning method. For uCE-MIRL, we develop a linear programming problem subject to constraints that define necessary and sufficient conditions for the observed policies to be correlated equilibria. The objective is to choose a solution that not only minimizes the total game value difference between the observed bi-policy and a local uCS, but also maximizes the scale of the solution. We apply a similar treatment to the problem of uNE-MIRL. We demonstrate these algorithms on multiple grid-world experiments, concluding: 1) all these algorithms are able to recover high-quality rewards comparable to ground truths; 2) perform better than other methods, such as decentralized-MIRL and IRL.

Acknowledgements

During my PhD studies, many people offer tremendous help. The first one to whom I would like to express my deepest appreciation is my academic advisor Prof. Peter Beling. He not only offers me this great research opportunity, but also teaches me the way to become a good researcher. He directs my research towards interesting topics and works together with me on challenging problems. When I feel frustrated, he encourages patiently and brings back my self-confidence. Beyond academics, he spent a lot of time helping me greatly improve my writing skills. All in all, to me, he is not simply a teacher or professor, but more like a life coach, because under his supervision, I can feel great improvement in almost every aspect, way beyond academics.

Next, I would like to thank my Ph.D. advisory committee members: Prof. Zongli Lin, Prof. Laura Barnes, Prof. Matthew Gerber, Prof. Nicola Bezzo and Dr. Stephen Adams. Their valuable advice is an indispensable component in this thesis. I would like to thank Randy Cogill, Stephen Adams and Benjamin Choo. They have either advised me or collaborated with me in my research. Their ideas and encouragements refresh my mind and speed up my research progress.

I also want to say thank you to administrative officers in our department, particularly Jayne Weber. She treats every student like her kid, making every effort to help us with whatever difficulties we have.

Lastly I would like to thank my parents Lingzhu Zhu and Yizheng Lin. Their spiritual and financial support is of critical help in my studies and research.

Contents

Abstract iii

Acknowledgements v

1 Introduction 1

1.1 Motivation 5

1.1.1 Sports 5

1.1.2 Manufacturing 6

1.1.3 Difficulties and Proposed Solution 7

1.2 Contributions 8

1.3 Dissertation Organization 11

2 Background 13

2.1 Multi-agent Systems 13

2.2 Game Theory 16

2.2.1 Strategic Form Game 16

2.2.2 Nash Equilibrium 18

2.2.3 Correlated Equilibrium 20

2.3 Bayesian Inference 23

3 Related Work 25

3.1 IRL 25

3.2 MRL Summary 27

3.3 zero-sum MRL 28

3.4 Nash-Q MRL 30

3.5 MIRL Summary 31

4 Zero-sum MIRL **35**

4.1 Introduction 35

4.2 Preliminaries 36

4.2.1 Zero-sum Stochastic Games 36

4.3 Bayesian MIRL 41

4.3.1 Prior Distributions on Rewards 42

4.3.2 Likelihood Function (Unique Minimax bi-policy) . . . 43

4.3.3 MAP Estimation Model 43

4.3.4 Discussion on Nonunique bi-policies 47

4.3.5 Uniqueness of bi-policy 48

4.4 Linear d-MIRL 48

4.5 Bayesian IRL 49

4.6 Numerical Example 50

4.6.1 Game and Model 50

4.6.2 Specification of Prior Information 52

Mean of the Prior 53

Covariance Matrix 54

4.6.3 Results Evaluation Metric 54

4.6.4 Results 58

4.6.5 Analysis of Results 59

4.7 Monte Carlo Simulation using Recovered Rewards 61

4.8 Additional Experiments 64

4.9 Conclusions 66

5 General-sum MIRL **69**

5.1 Introduction 69

5.2 Preliminaries 71

5.2.1 General-sum Stochastic Game 71

5.2.2 MRL 72
5.2.3 Cooperative Strategy 74
5.3 Conventional MIRL Approaches 75
5.4 MIRL Model Development 77
5.4.1 Extension to stochastic games 79
5.4.2 uCS-MIRL 79
5.4.3 advE-MIRL 82
5.4.4 cooE-MIRL 84
5.4.5 uCE-MIRL 86
5.4.6 uNE-MIRL 90
5.5 Numerical Examples I: GridWorld 91
5.6 Numerical Examples II: Abstract Soccer Game 97
5.6.1 Prior Specification 98
5.6.2 Monte Carlo Simulation using Recovered Rewards . . 100
5.7 Conclusions 102

6 Conclusions **105**

Bibliography **109**

List of Figures

2.1 Normal form game example: Prisoner's Dillema 17
2.2 Chicken Game 20
2.3 Traffic light control 22

4.1 Soccer game: initial board 52
4.2 Inferred rewards and PSS: weak mean & weak covariance . . 55
4.3 Inferred rewards and PSS: weak mean & strong covariance . . 56
4.4 Inferred rewards and PSS: median mean & weak covariance . 56
4.5 Inferred rewards and PSS: median mean & strong covariance . 57
4.6 Inferred rewards and PSS: strong mean & weak covariance . . 57
4.7 Inferred rewards and PSS: strong mean & strong covariance . 58
4.9 Two evaluation metrics comparison 63
4.10 Soccer game: 5*5 board 65

5.1 (A) describes the relationship between uCE, uCS and other CEs. (B) explains local uCS. 89
5.2 Grid games. The circle indicates A's goal and the hexagon indicates B's goal. 92
5.3 The uCS-MIRL results 96
5.4 cooE-MIRL experiment result 96
5.5 Soccer game: initial board 98

List of Tables

4.1 Original PSS distribution of each player 52
4.2 BMIRL results summary 59
4.3 Numerical results comparison 60
4.4 Comparison between prior mean and posterior 60
4.5 B vs A games simulation results 63
4.6 B vs C games simulation results 63
4.7 B vs D games simulation results 64
4.8 B vs B_p games simulation results 65
4.9 B_{5*5} vs A_{5*5} games simulation results 66
4.10 Policy difference w.r.t. β 67

5.1 NAED results for reward values comparison 95
5.2 total game value comparison for uCS 95
5.3 Original PSS distribution of each player 97
5.4 C vs D 101
5.5 C vs E 101
5.6 C vs F 102
5.7 C vs G 102

List of Abbreviations

MAS	**M**ulti-agent **S**ystem
RL	**R**inforcement **L**earning
IRL	**I**nverse **R**inforcement **L**earning
MRL	**M**ulti-agent **R**inforcement **L**earning
MIRL	**M**ulti-agent **I**nverse **R**inforcement **L**earning
MDP	**M**arkov **D**ecision **P**rocess
uCS	**U**tilitarian **C**ooperative **S**trategy
advE	**A**dversarial **E**quilibrium
cooE	**C**oordination **E**quilibrium
uCE	**U**tilitarian **C**orrelated **E**quilibrium
uCE	**U**tilitarian **N**ash **E**quilibrium

Chapter 1

Introduction

Artificial intelligence (AI), informally speaking, aims to construct machines that are capable of executing tasks and solving problems in ways normally attributed to humans. A machine, once programmable to have an ability to make right decisions in a specific environment, can be recognized as an “intelligent agent". There is a fundamental difference between automation and AI. Automation is basically making a hardware or software that is capable of doing things automatically. An example of automation is a fire alarm system. Once the smoke sensor is activated, water starts pouring down the pipes. this example, the fire alarm system has the ability to do things automatically. We cannot claim, however, it is an AI system, due to the insufficient proof of intelligence. Simply put, automation relies primarily on pre-programmed controls and AI is capable of self-learning and evolving like human beings.

In academia, AI research is motivated by three philosophies (Millingto, 2009): 1. understanding the nature of thoughtïijN̆as well as intelligence; 2. understanding the mechanics of the human brain and mental processes, and; 3. implementing software to model the way of thinking and algorithms to perform human-like tasks. This distinction is central to the view and activities of AI researchers (Millingto, 2009).

Human beings are cognitive agents. Cognitive agents think about the environment, evaluate various aspects of it and act upon their responsive decisions. This is known as the *doxast-conative* loop. The defining characteristic

of cognitive agents is that they employ doxast-conative by thinking about the environment and acting uopon their perseptions (Pollock, 2006). This decision-making process also partially defines rationality. An autonomous agent envisaged in AI is considered as a cogitative agent if it posses this characteristic. Hence the key aspect of AI research is to study, understand and model the cognition, or more specifically, the *rationality*.

AI has revolutionized many areas since its inception. One emerging domain where AI has made breakthrough contributions is computer games. The gaming industry has seen great strides in recent decades, with more and more complex and intelligent games springing up. On the one hand, game developers face the challenge of creating games that are increasingly compelling (Greene, 2017; Lou, 2017). On the other hand, the development of multi-core processors and other hardware advancements help computer-aided AI evolve rapidly and thus profoundly change the industry. The most recent achievement astonishing the whole world is AlphaGo, developed by Google DeepMind (Mozur, 2017).

Learning from demonstrations (LD) is a traditional line of research in behavior learning, and attracts attention from AI community. In LD, policy learning directly from observations has achieved remarkable success in large part because it can benefit from advanced supervised learning techniques. For example, Runarsson and Lucas use preference learning for policy learning (Runarsson and Lucas, 2014). The most recent work is to adopt a deep convolutional neural network as the basis for policy learning (Maddison et al., 2015).

Researchers from the machine learning community believe that behavior is mainly reward-driven (Sutton and Barto, 1998; Ng and Russell, 2000; Russell, 1998). On the basis of this philosophy, Reinforcement Learning (RL) has been studied extensively and has become one of the three pillars in machine learning (Sutton and Barto, 1998). RL solves sequential decision problems

by interacting with an environment. The model RL adopts for the sequential decision process is a Markov Decision Process (MDP). For a single agent, RL aims to find an optimal policy of actions for the purpose of maximizing its total reward. Its "inverse" version, termed Inverse Reinforcement Learning (IRL) aims to recover reward (equivalently, payoff or cost) functions given measurements of an agent's behavior over time as well as a model of the environment. IRL was introduced by Russell (Russell, 1998) and then formalized by Ng and Russell (Ng and Russell, 2000) in the context of several linear programming algorithms. One can view the IRL problem as being that of learning the reward structure for a game given observations of the play of an expert. The key assumption in IRL is that the agent has a clear reward perception (though not available to us) and takes sequential actions in order to maximize its total reward in the long run.

One major advantage of IRL, as pointed out by Ng and Russell (Ng and Russell, 2000), is that in many applications, the reward function provides a parsimonious description of behavior that is succinct, robust, and transferable with respect to changes in the environment. The comparison between policy learning and reward learning is discussed by Abbeel and Ng (Abbeel and Ng, 2004).

IRL has found many applications such as control and transportation and achieved solid success. Examples include simulated driver-less car (Abbeel and Ng, 2004; Syed and Schapire, 2007), traffic navigation (Abbeel, Dolgov, and Ng, 2008; Ziebart et al., 2008), path planning (Mombaur, Truong, and Laumond, 2010) and human goal inference (Qiao and Beling, 2013). More details about IRL can be found in Section 3.1.

A multi-agent system (MAS) aims to provide both principles for construction of complex systems involving multiple agents and mechanisms of interaction among independent agents (Stone and Veloso, 2000). MASs can be used to solve problems that are either difficult for an individual agent

system to solve, or result in a poor performance if using a single agent approach. Two fundamental classes of MAS problems are of interest: 1. build or optimize a MAS to meet some pre-determined requirements (Parka and Sugumaran, 2005), and 2. learn some information from an established MAS (Todd, Beling, and Scherer, 2016). The first one has been studied extensively, while the second one lags behind. More details of MAS can be found in Section 2.1.

RL can be extended to multi-agent reinforcement learning (MRL), where a framework of stochastic game instead of MDP is adopted (Littman, 1994). A major difference between IRL and MRL, as Hu and Wellman point out, is that the concept of "optimality" loses its meaning in MRL as any agent's reward depends on others' actions (Hu and Wellman, 1998). In a stochastic game, equilibrium plays a vital role and is used as a solution concept, which ensures that every agent achieves highest reward given others do not change their strategies. Hence no one has the incentive to deviate from its equilibrium strategy.

One difficulty MRL has is that for a general game, there could exist different types of equilibriums with different properties. For example, for every game, there exists at least one or more correlated and Nash equilibriums. The details of these two types of equilibrium concepts can be found in Chapter 2. Therefore, the issue of convergence is still a roadblock to a general MRL problem (Shoham, Powers, and Grenager, 2003).

Another class of important research problems, which is an "inverse" version of MRL, is termed multi-agent inverse reinforcement learning (MIRL). Specifically, in a multi-agent system, given all participants' activities, we may want to characterize/distinguish each individual for the purpose of: 1. learn and understand all individuals' behaviors, and; 2. use what we have learned to make predictions. Representative works on MIRL include (Natarajan et al., 2010; Waugh, Ziebart, and Bagnell, 2011; Reddy et al., 2012). On the whole,

unfortunately, not much literature discusses this topic.

This dissertation focuses on multi-agent inverse learning problems and handles them from a machine learning perspective. Specifically, we model such problems as MIRL, adopting stochastic games as a framework and developing various algorithms to enable inference of their goals in different situations given the observation of their behaviors. A basic assumption of a certain degree of rationality of all agents with respect to some agreement or equilibrium is required. Though we neither offer a universal algorithm that is applicable to all MASs, nor propose a complete set of algorithms that cover all situations, we contribute a major addition to the current theory of MIRL and point out promising directions for further efforts.

1.1 Motivation

There are many real problems associated with agents that need to be addressed. We explore two typical problems in details here.

1.1.1 Sports

In competitive sports games, how to effectively *learn your opponent's preferences* is an open-ended research question in sports psychology (Hodges, 2016; Borum, 2009). At an individual level, an athlete's preferences or subjective utilities, depending on how he or she perceives the condition, determines his or her actions. Suppose you play against your opponent. The advantages of inferring his or her preferences include:

- If the opponent is an individual, you are able to understand his or her *tactics* more deeply and adjust your own more effectively;
- If your opponent is a team, what you will learn is the coach's preferences. In practice, each coach has his or her own style, strategies, or

biases, all of which may plausibly remain stable long enough to be exploited if properly inferred.

There exists many statistical methods to measure the skills of an athlete or a team (McGuigan, 2017; J. Albert and Koning, 2016) by observing their actions. However, the task of inferring preferences and subjective utilities from observed actions in a sports game is beyond the capability of the existing literature.

1.1.2 Manufacturing

Growth in intelligent manufacturing is a clear trend for the next couple of decades. Many countries impose great importance in pursuit of this trend. For example, Germany proposed Industry 4.0 Action Plan (Earls, 2015) and China launched a even greater plan, called "Made in China 2025" (Kuo, 2017). A key component of the intelligent manufacturing is intelligent robotics (Kopacek, 1999). As an example, Foxconn Technology Group, a major manufacturer of Apple's products, has replaced $60,000$ factory workers with robots (Wakefield, 2016).

Intent inference is important to multi-robot control problems (Valtazanos and Ramamoorthy, 2011; Kirchner et al., 2016). For example, one method for training a robot to perform a task like a human is to let it infer the human expert's intent by observing his or her actions. Another interesting but more realistic question is, for example, for a complex task that two human experts are required to work together to accomplish, is it possible for two robots to infer the intents from human experts and learn to cooperate?

1.1.3 Difficulties and Proposed Solution

IRL is a good candidate approach for addressing agent learning problems. However, an implicit assumption IRL requires is that only one agent is involved in the problem and only its decision will impact the environment and trigger a state transition dynamic. If more adaptive agents are involved, the system becomes an MAS and creates more complicated issues with which current IRL algorithms may fail to deal, such as:

- The state transition dynamic is controlled by all agents instead of any individual's action;
- When taking an action, each agent needs to take others' actions/responses into consideration, and;
- The relationship between agents can be complicated. For example, one common situation is that every agent is selfish and just tries to maximize her own utility. Another circumstance could be that all agents cooperate in order to achieve optimal social welfare. There are even cases where agents are semi-cooperative or semi-competitive.

Obviously, the sports and manufacturing problems motivating our research are both multi-agent problems. IRL may not be applicable a MAS. To see this, consider a simple one-state example. In the single state, agent A can take action X or Y and agent B is also allowed take action X or Y. Their rewards are:

- If they both take action X, A will get 3 and B will get 0;
- If they both take action X, A will get 0 and B will get 3;
- If A takes action X and B takes action Y, A will get 1 and B will get 2;
- If A takes action Y and B takes action X, A will get 2 and B will get 1.

Given A is rational, it cannot make decisions without taking B into account. Although its largest possible reward is 3, A cannot take X without hesitation, otherwise may probably end up with receiving only 1. The essential reason for this phenomenon, is that the “optimality" concept in IRL does not hold in multi-agent conditions.

As a summary of all the above discussions, MIRL, if perfectly developed, will be a suitable method to deal with the sports and manufacturing problems. However, comparing to RL and IRL, little progress has been made to the theory of MIRL, as well as applications. That motivates our ambition to build a solid theoretical foundation for MIRL.

1.2 Contributions

We address the problem of multi-agent learning from demonstrations. To begin with, we build a new multi-agent based model in place of IRL, termed multi-agent inverse reinforcement learning (MIRL). Though there are many equilibriums, the MRL community has found several that have been proved unique or empirically unique and lead to the corresponding MRL problems being solvable with satisfying results (Hu and Wellman, 2003; Littman, 2001; Greenwald and Hall, 2003). To our knowledge, they are:

1. **Minimax equilibrium**. For a *fully* competitive game where two agents compete with each other and their rewards/payoffs sum to zero for every game.

2. **Coordination equilibrium (cooE)**. It belongs to a win-win-or-lose-lose game where agents employ a special Nash equilibrium and fully cooperate for the sake of themselves.

3. **Adversarial equilibrium (advE)** It belongs to a competitive game where agents play against each other. The game differs from a zero-sum game

in the sense that it relaxes the zero-sum requirement of rewards, though still a win-or-lose game.

4. ***utilitarian*** **correlated equilibrium (uCE)** is such an equilibrium that agents employ one correlated equilibrium (a third-party mediator sends private recommendation of action to each agent) which generates the largest total game value among all CEs.

5. ***egalitarian*** **correlated equilibrium (eCE)** is such an equilibrium that agents employ one correlated equilibrium which maximizes the minimum of the agents' rewards.

6. ***republican*** **correlated equilibrium (rCE)** is such an equilibrium that agents employ one correlated equilibrium which maximizes the maximum of the agents' rewards.

7. ***libertarian*** **correlated equilibrium (lCE)** is such an equilibrium that agents employ one correlated equilibrium which maximizes the maximum of the agents' rewards.

In addition, we develop another two equilibriums:

1. ***utilitarian*** **Cooperative Strategy (uCS)** belongs to a cooperative game, by employing which agents *fully* cooperate with each other to achieve a pre-determined goal.

2. ***utilitarian*** **Nash equilibrium (uNE)** is such an equilibrium that agents employ one Nash equilibrium which generates the largest total game value among all NEs.

From the above nine equilibriums, we select six: minimax, cooE, advE, uCE, uCS and uNE and develop novel algorithms to address six MIRL problems associated with those equilibriums. Though the specific algorithms we develop for these problems are different from each other, the general idea to

treat these problems, in a high level, is similar: taking the advantage of their uniqueness property, we first characterize a set of unknown rewards in which each solution point is consistent with the observed polices, and then develop algorithms to pick the most “reasonable" solution. This *two-step* paradigm can be extended to other MIRL problems.

It is worth emphasizing the rationale for us to address the above six problems. First, zero-sum MIRL is interesting because in reality, zero-sum games are very popular. For example, a basketball game, football game or other purely win-or-lose competitive game is either actual or can be treated as a zero-sum game. One potential advantage of zero-sum MIRL brings to the table is, it is able to learn the opponents' preferences, as is discussed in Section 1.1.1 We show this in an abstract soccer game example in Chapter 4.

The uCS-MIRL is a *fully* cooperative game and an obvious application is the manufacturing problem described in Section 1.1.2. The cooE-MIRL applies in a all- win or all-lose situation. It also applies in a manufacturing task that can be accomplished only when workers coordinate their jobs closely and carefully. The application for advE-MIRL is straightforward, and is applicable to more than two agents. One of its applications can be marketing: by observing the competition among several local major automobile dealers and applying advE-MIRL, a new dealer would be able to have a better insight into not only the whole market (which can be obtained through a statistical analysis), but also each of its potential opponents' strategies. The uCE- and uNE-MIRL can be regarded as a constrained uCS-MIRL problem. In the manufacturing problem, they are particularly useful when the coordinated robots are imposed a resource constraint.

There are two reasons that we do not select the remaining three equilibriums, eCE, rCE and lCE. First, we do not think any of these equilibriums has many real applications. Second, the way we treat the other six equilibriums can be applied to them and thus this is our future work.

To conclude, we are the first, to our knowledge, to propose a set of MIRL algorithms covering all types of games, from cooperative to non-cooperative games. In addition to the theory of MIRL we develop in this dissertation, our contributions to the MIRL community also include:

- Formalize a stochastic game based framework for MIRL problems;
- Propose a two-step treatment that can be applicable to other MIRL problems,and;
- Develop a novel way, Monte-Carlo simulation, to evaluate the quality of solutions for MIRL problems.

1.3 Dissertation Organization

The remainder of this dissertation is organized into five chapters. In Chapter 2, we give a brief introduction of multi-agent systems and game theory, which are prerequisites of our MIRL algorithms. Chapter 3 reviews MRL research findings and current MIRL research progress. Chapter 4 proposes a Bayesian algorithm for zero-sum MIRL problems. Chapter 5 covers five general-sum MIRL cases, and proposes five algorithms for the five cases, respectively. Finally, in Chapter 6, we offer conclusions and suggestions for future research.

Chapter 2

Background

Agent interaction and learning relies upon existing concepts and ideas from machine learning, statistics, economics and sociology. Two major theoretical background of this dissertation, in addition to machine learning and other statistical techniques, are multi-agent system and game theory. The former one builds the system basis and the latter provides a rigorous mathematical description of interactions between multiple agents.

2.1 Multi-agent Systems

Since the early 1980's the *multi-agent system* MAS became an increasing popular architecture for solving computational problems of a distributed nature. Although today's AI problems are increasingly complex with high computational expenses, the development of salable and reliable distributed system offers the possibility to handle these problems. Hence MAS plays a more and more vital role in AI. Although the MAS has found a huge number of applications, many of them can be classified in terms of objectives into three main categories:

- *Simulation and Prediction*. A MAS can be constructed by simulating tens of thousands of individual agents. The goal is to search for explanatory insight into emergent properties of the group of agents. For example,

local police can treat a local resident as an agent, assigning various attributes to him or her with some randomness added to model his or her behavior (Zhang and Brown, 2014). This may yield insight into societal trends or provide a forecast of potential events.

- *Optimization*. This category applies when the research question is focused on resource allocation, coalition formation and cooperative decentralized decision making. For example, a robotic soccer team design is to find state-of-the-art algorithms to guide individual activities and thus optimize team performance (Ould-Khessal, 2005).

- *Learning*. To understand how the MAS works or evolves, one way is to break the whole system down and either focus on individual agents (for a distributed MAS), or consider their mutual interactions as well (for a centralized MAS). Learning from them help us understand some phenomenon arised from the system and predict the system evolvement.

The literature offers a variety of definitions of *agent*, without a universal understanding of the term. Definitions tend to be strongly biased by the fields in which they arise, such as artificial intelligence or cognitive science. In our opinion, the following definition given by Maes (Maes, 1995) represents the perspective of AI community.

"Autonomous agents are computational systems that inhabit some complex, dynamic environment, sense and act autonomously in this environment, and by doing so realize a set of goals or tasks for which they are designed."

The above definition holds for a MAS. Each agent in a MAS takes actions purposely and interact with each other. Stone and Veloso define a MAS as follows (Stone and Veloso, 2000).

"A multi-agent system is a loosely coupled network of problem-solving entities (agents) that work together to find answers to problems that are beyond the individual capabilities or knowledge of each entity (agent)."

As few system involves exactly only one agent, modeling most problems as MASs leads to a higher accurate solution than modeling them as single-agent systems; in our opinion, it has the following advantages:

- From a perspective of resource distribution, a single-agent system may suffer from resource limitation, which potentially leads to performance performance bottlenecks or even failures. In contrast, an MAS is able to break through this restriction in the sense that one agent's failure does not cause the callapse of the whole system (Lynch, 2009).

- A single-agent system, in many cases, approximates an MAS by treating other *adaptive* agents as passive objects or part of the environment. This approximation simplifies the model at the cost of losing accuracy.

- In a highly large complex system where interconnection of multiple sub-systems exists, an MAS provides such a two-layer hierarchical solution that in the upper layer, treating each sub-system as a single agent and building a wrapper around it while in the bottom layer, unwrapping them and applying a MAS treatment.

The main objective of the agents in an MAS is to learn how to act for some purpose. For any agent, however, learning is more difficult in an MAS than in a single agent system because of the presence of multiple decision makers, which raises issues such as interferences, communications and coordinations. Since its inception, a lot of research efforts have been put into the field of MAS and thus enables us to apply MAS approaches to enormous fields, such as AI (Stone and Veloso, 2000), machine learning (Hu and Wellman, 1998; Littman, 1994), distributed systems (dInverno et al., 2004), communications (Giles and Jim, 2002; Pitt and Mamdani, 2000), robotic systems

(Liu and Wu, 2002), operations research (Gabel and Riedmiller, 2007) and economics (Bajo, Mathieu, and Escalona, 2017). There are many frameworks and schematics to model an MAS, and one of them is to borrow ideas from Reinforcement Learning and game theory. This is the perspective adopted in this dissertation.

2.2 Game Theory

Borrowing the definition of *agent* in Section 2.1, *Game Theory* is a mathematical framework for studying the interactions between rational agents. In a game, a number of agents interact, in either a cooperative or noncooperative fasion, take actions accordingly, and eventually receive some benefit or loss upon joint actions (Ferguson, 2008). Game theory has been widely used in economics, social science, psychology, biology and computer science (Owen, 1968). By studying game theory, we are able to:

- Find the best actions to taken in consistent with our objectives, and;
- Understand what is happening in order to make better predictions about the future.

Another topic in game theory attracs our attention is *mechanism design*, which is also called “inverse game theory". Mechanism design aims to design economic mechanisms or incentives to achieve some desired objective (for example, system-wide goal or designer’s selfish objective), based on the assumption that players act rationally (Hurwicz and Reiter, 2006).

2.2.1 Strategic Form Game

In game theory, *Strategic Form*, or *Normal Form* is a *simultaneous* game represented by a matrix where the rows denote actions of one agent and the columns denote actions of other agents (Fudenberg and Tirole, 1991). When

there are only two agents involved the game is known as a *bimatrix* game. For example, the famous *Prisoner's Dilemma* (Amadae, 2016), shown in Figure 2.1, is a normal form game in which players act simultaneously (or at least do not observe the other player's act before taking their own) and receive payoffs as specified for the joint actions taken. In each cell, the first number represents the payoff to the row player (in this case player #1), and the second number represents the payoff to the column player (in this case player #2). For example, if both players take *Cooperate* action, both of them will receive a payoff of -1. In each game, we have to define

- The set of players;
- The strategy of each player, and;
- The payoffs to each player depending their joint actions.

		Player #2	
		Cooperate	*Defect*
Player #1	*Cooperate*	(-1, -1)	(-5, 0)
	Defect	(0, -5)	(-2, -2)

FIGURE 2.1: Normal form game example: Prisoner's Dillema

A more rigorous and formal definition of normal form game is (Ozdaglar, 2010):

Definition 2.1. *A strategic form game is a triplet* $\langle \mathcal{I}, (\mathcal{A}_i)_{i\in\mathcal{I}}, (u_i)_{i\in\mathcal{S}} \rangle$ *where* $\mathcal{I}$ *is the finite set of players;* $\mathcal{A}_i$ *is the set of available actions for player* i*;* $a_i \in \mathcal{A}_i$ *is an action for player* i*;* $u_i : \mathcal{A} \to R$ *is the payoff (utility) function of player* i *where* $\mathcal{A} = \prod_i \mathcal{A}_i$ *is the joint action.*

One more important concept in game theory is *strategy* (Ozdaglar, 2010). A *strategy* is a complete description of how to play the game. A *pure* strategy determines an action that a player will always take. A *mixed* strategy

determines a probability distribution over all pure strategies and according to that distribution, a player is allowed to select a pure strategy. Since probabilities are continuous, there are infinitely many mixed strategies available to a player. In addition, a set of strategies of all players is a *strategy profile*.

Finally, it is worth noting that in Section 2.1, the sums of payoffs of the two players are *not* all 0 for all cells. This type of game is called a *general-sum* game. Otherwise the game is called a *zero-sum* game and in this case, we usually only put player #1's payoff in each cell.

2.2.2 Nash Equilibrium

In game theory, a major class of games is *Noncooperative Game*. In general, communication is not allowed for players in a noncooperative game so that no binding agreements can be achieved. Thus the only agreements that may to occur are those that are *self-enforcing*, in which no player is able to gain by unilaterally violating the agreements. Such an agreement is called *strategic equilibria* or *Nash equilibria* (NE) (Ferguson, 2008). The formal definition of NE is:

Definition 2.2. *A mixed strategy profile π is a (mixed strategy) Nash equilibrium if and only if for each player i,*

$$R_i\left(\pi_i, \pi_{-i}\right) \geq R_i\left(a_i, \pi_{-i}\right), a_i \in \mathcal{A}_i \tag{2.1}$$

where π_i and a_i denote a mixed and pure strategy of player #1, respectively, and π_{-i} denotes the joint strategies of others.

To better illustrate NE, we now use the Prisoner's Dilemma example in Figure 2.1. To find a NE in that game, we examine each pure strategy profile in turn, as follows (C and D are short for $Cooperate$ and $Defect$, respectively):

- (C, C): player #1 tends to choose D rather than C because she would obtain a payoff of 0 rather than -1, given player #2 does not change. Thus this strategy profile is not a NE.
- (C, D): player #1 tends to choose D rather than C because she would obtain a payoff of -2 rather than -5, given player #2 does not change. Thus this strategy profile is not a NE.
- (D, C): player #2 tends to choose D rather than C because she would obtain a payoff of -2 rather than -5, given player #1 does not change. Thus this strategy profile is not a NE.
- (D, D): Neither player can increase theirs payoffs by changing strategies. Thus this strategy profile is a NE.

The most important theorem regarding to the existence of NE is as follows (Nash, 1951; Owen, 1968):

Theorem 2.3. *(Nash existence) Every finite game has at least one mixed strategy Nash equilibrium.*

A rigorous proof of this theorem can be found in several works (Ozdaglar, 2010; Owen, 1968). Note that in the above Prisoner's Dilemma game, there is only one pure NE (Defect, Defect). It is worth emphasizing that a *pure* NE is not guaranteed to exist for a finite game. Theorem 2.3 is important because

- It is difficult to understand its properties without knowing the existence of NE, and;
- We can focus on developing algorithms of finding the NEs.

We have just introduced the Nash's existence theorem applied for finite games. There are also similar Nash's existence theorems (for pure and mixed) for infinite games (Ozdaglar, 2010), but that is beyond the scope of this dissertation.

Although the problem of finding NE has found increasing applications in many fields, it has been shown that it is a NP-hard problem (Daskalakis, Goldberg, and Papadimitriou, 2009). So far, many algorithms, such as Lemke-Howson Algorithm, have been developed to address this problem. More details of the Nash-finding works include (Abbot, Kane, and Valiant, 2004; Daskalakis, Goldberg, and Papadimitriou, 2009).

2.2.3 Correlated Equilibrium

To interpret Correlated Equilibrium (CE), it is better to first go through an example. Consider the *Chicken game* (Rapoport and Chammah, 1966) shown in Figure 2.2. This is a two-player general-sum game. Player #1 (row player) has the option to play C or D and and so does Player #2 (column player). In each cell, the first number is the payoff to player #1 and the second number is the payoff to player #2. More formally, the cell indexed by row x and y represents a payoff pair (a, b), where $a = u_1(x, y)$ and $b = u_2(x, y)$.

	C	D
C	6, 6 (π_{CC})	2, 7 (π_{CD})
D	7, 2 (π_{DC})	0, 0 (π_{DD})

FIGURE 2.2: Chicken Game

Assume there is a *trusted mediator* devising a *joint* bi-strategy π for Player # 1 and Player # 2 as the following:

1. Player # 1 takes action C and Player # 2 takes action C, with probability π_{CC};

2. Player # 1 takes action C and Player # 2 takes action D, with probability π_{CD};

3. Player # 1 takes action D and Player # 2 takes action C, with probability π_{DC};

4. Player # 1 takes action D and Player # 2 takes action D, with probability π_{DD};

where $\pi_{CC} + \pi_{CD} + \pi_{DC} + \pi_{DD} = 1$. The mediator picks one of the above four pure bi-strategies according to this probability distribution. Once a pure bi-strategy is picked, the mediator will make a recommendation to both players accordingly. However, both players receive the recommendation regarding their own actions without knowing what recommendation the other player receives. Each player has the option to accept the recommendation or not.

A CE is such an equilibrium that each player will accept the recommendation with the belief that the other player will also accept the recommendation. In other words, in the case that the other player takes the recommended action, one will not benefit from deviating from his recommendation.

In the above game, a trusted party designs a probability distribution over joint actions, $\pi = (\pi_{CC}, \pi_{CD}, \pi_{DC}, \pi_{DD})$. Then it pick one among the four, according to π, say (C, C) to player 1 and 2. If π is a CE, the following inequality must hold:

$$\begin{aligned} 6\pi_{CC} + 2\pi_{CD} &\geq 7\pi_{CC} + 0\pi_{CD} \\ 6\pi_{CC} + 2\pi_{DC} &\geq 7\pi_{CC} + 0\pi_{DC} \end{aligned} \tag{2.2}$$

Similarly, we will have

$$\begin{aligned} 7\pi_{DC} + 0\pi_{DD} &\geqslant 6\pi_{DC} + 2\pi_{DD}, \\ 6\pi_{CC} + 2\pi_{DC} &\geqslant 7\pi_{CC} + 0\pi_{DC}, \\ 7\pi_{CD} + 0\pi_{DD} &\geqslant 6\pi_{CD} + 2\pi_{DD} \end{aligned} \tag{2.3}$$

Armed with some intuition of CE given by the preceding example, we now formally define CE as follows:

Definition 2.4. *Let $\Delta(\mathcal{A})$ denote the set of probability distribution over $\mathcal{A}$, and R be a random variable taking values in $\mathcal{A} = \prod_{i\in\mathcal{I}} \mathcal{A}_i$ distributed according to a $\pi \in \Delta(\mathcal{A})$. Then π is a correlated equilibrium if and only if*

$$\Sigma_{a_{-i}\in A_{-i}} P(R = a|R_i = a_i)\left[u_i(a_i, a_{-i}) - u_i(\check{a}_i, a_{-i})\right] \geq 0,$$

for all $a_i \in \mathcal{A}_i$ such that $P(R_i = a_i) > 0$ and all $\check{a}_i \mathcal{A}_i \setminus a_i$.

One real application of CE is the traffic light control shown in Figure 2.3. A traffic light sends a private message with action recommendation to each car ("stop" for A and C and "go" for B and D). Each car can decide whether to accept it or not. Rationally, no car will reject the recommendation from the traffic lights by assuming that all other cars will obey the rules.

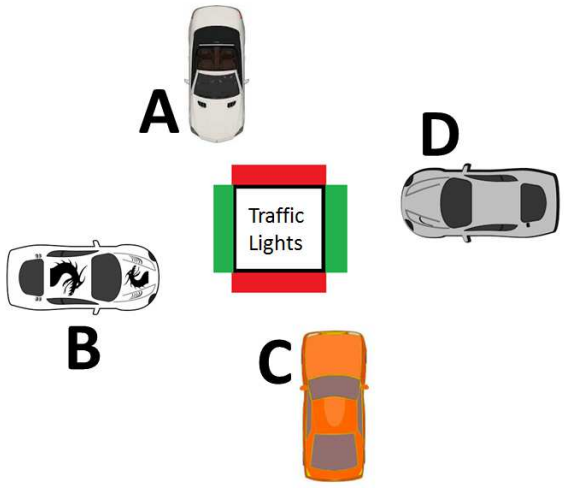


FIGURE 2.3: Traffic light control

Similar to NE, an important theorem regarding to the existence of CE is as follows (Hart and Schmeidler, 1989):

Theorem 2.5. *(Correlated equilibrium existence) Every finite game has at least one Correlated equilibrium.*

In fact, CE is a superset of NE (Aumann, 1974), and hence for any general sum game, the number of CEs is larger than or equal to that of NEs. Take the above chicken game as an example. There are three NEs, as the following:

1. pure strategy - (C, D), with game values $R_1 = 2, R_2 = 7$
2. pure strategy - (C, D), with game values $R_1 = 7, R_2 = 2$
3. mixed strategy - $\pi_1 : \left(\frac{2}{3}C, \frac{1}{3}D\right), \pi_2 : \left(\frac{2}{3}C, \frac{1}{3}D\right)$, with game values $R_1 = R_2 = \frac{14}{3}$

It is easy to show that $\pi = \left\{\pi_{CC} = \frac{1}{2}, \pi_{DC} = \frac{1}{4}, \pi_{CD} = \frac{1}{4}\right\}$ is a CE but not a NE. Each player's game value, under this equilibrium, is 5.25.

In contrast to NE, for which no efficient method of computation is known, finding CEs can be done in polynomial time via linear programming. However, the non-convergence issue still occurs for MRL unless a particular and unique CE is agreed.

2.3 Bayesian Inference

Throughout this dissertation we extensively use an approach to inverse learning problems that is grounded in the framework of Bayesian statistical inference. Bayesian statistical inference provides a probabilistic method for problems involving the estimation of unknowns, where the uncertainty in the values of these unknowns can be characterized by probability models. There are two major methods for Bayesian estimation, maximum likelihood (ML) and maximum a-posteriori (MAP) (Casella and Berger, 2001). While both methods use observations to make estimations, MAP offers an additional benefit of taking advantage of the prior knowledge of the unknowns, usually in the form of a probability distribution over possible of values of the unknowns. Therefore, MAP is a compromise between the prior and the likelihood. MAP performs better when an informative prior is available.

We use an example for illustration. Suppose a scalar parameter θ needs to be estimated. Our initial uncertainty in the value of θ, which is the prior, could be described by a probability distribution $f(\theta)$. We then receive a sequence of observations $x_1, \ldots, x_n$ that are generated from some random process related to θ. Each observation is often considered i.i.d. We also assume that a conditional PDF $f(x_i \,|\, \theta)$ that characterizes the likelihood of observing a specific value x_i given θ is available to us. The MAP of θ given the sequence of observations can be expressed as

$$\theta_{\text{MAP}} = \text{argmax}_\theta \left\{ f(\theta \,|\, x_1, \ldots, x_m) \right\}$$

where

$$f(\theta \,|\, x_1, \ldots, x_m) \propto f(\theta) f(x_1 \,|\, \theta) \cdots f(x_m \,|\, \theta)$$

The PDFs of many common random variables, particularly if they belong to the exponential family, are log-concave (Marshall and Olkin, 1988). Thus the above optimization problem is often convex.

Chapter 3

Related Work

This chapter reviews IRL, MRL, and the most recent MIRL research results.

3.1 IRL

A finite-state, infinite horizon *Markov decision process (MDP)* is defined as a tuple $M = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \gamma, r)$, where $\mathcal{S} = \{s_1, s_2, \cdots, s_n\}$ is a set of n states; $\mathcal{A} = \{a_1, a_2, \cdots, a_m\}$ is a set of m actions; $\mathcal{P} = \{P_a\}_{a=1}^{m}$ is a set of state transition probabilities; γ is a discount factor; r is a state dependent reward vector of length n such that $r(s)$ is the immediate reward received upon arriving state s. For any $a \in \mathcal{A}$, each row of the $n \times n$ matrix, P_a, denoting as P_{as}, is the probability distribution over all next states transiting from current state s upon taking action a (Ng and Russell, 2000).

Let π be a policy of actions to take over all states, the *value function* at state s with respect to policy π is defined as $V^{\pi}(s) = E[\sum_{t=0}^{\infty} \gamma^t r^{\pi}(s^t) | s = s^0]$, where the expectation is over the distribution of the state sequence $\{s^0, s^1, \dots\}$ (superscripts index time) given policy π. The Q-function $Q^{\pi}(s, a)$ is defined, as a function of state s and action a under policy π, to be the expected return from state s, taking action a and thereafter following policy π. Given a policy

π, we can have

$$V^{\pi}(s) = r_s + \gamma \sum_{s'} P_{\pi}(s') V^{\pi}(s')$$

$$Q^{\pi}(s,a) = r_s + \gamma \sum_{s'} P_{as}(s') V^{\pi}(s')$$

for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$. The well-known Bellman optimality conditions state that π is optimal if and only if, $\forall s \in S$, we have $\pi(s) \in \arg\max_{a \in \mathcal{A}} Q^{\pi}(s,a)$ (Bellman, 1957).

An *inverse Markov decision process (IMDP)* $M_I = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \gamma, \mathcal{O})$ is defined as a tuple includes the states, actions, and state transition dynamics and a reward discount factor. While it lacks a specification of the reward vector, M_I includes a set of observations $\mathcal{O}$ of state-action pairs generated. We can define the *inverse reinforcement learning* (IRL) problem associated with $M_I = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \gamma, \mathcal{O})$ to be that of finding a reward vector r such that the observations are results of an optimal policy for $M = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \gamma, r)$. The IRL problem is, in general, ill-posed in nature, which has motivated researchers to develop various models for restricting the set of feasible solutions (Abbeel and Ng, 2004; Neu and Szepesvári, 2007; Syed, Michael, and E., 2008; Krishnamurthy and Todorov, 2010). Ng and Russel (Ng and Russell, 2000) characterize the feasible reward vectors that are consistent with an observed policy π, as $(P_{\pi} - P_a)(I_n - \gamma P_{\pi})^{-1} r \geq 0, \forall a \in \mathcal{A}$, where P_{π} is the transition probability matrix relating to observed policy π and P_a denotes the transition probability matrix for any other action. Note that the trivial solution $r = 0$ satisfies these constraints, which highlights the underspecified nature of the problem and the need for reward selection mechanisms. Ng and Russel (Ng and Russell, 2000) propose the idea of selecting a reward which maximizes the margin between the optimal and suboptimal policies. Another typical idea is to take advantage of MAP estimation or other Bayesian methods (Qiao and Beling, 2011; Ramachandran and Amir, 2007; Levine, Popović, and Koltun, 2011). Recent advancements in IRL include apprenticeship learning via IRL

(Abbeel and Ng, 2004), policy matching (Neu and Szepesvári, 2007), the treatment of partial policy observation (Choi and Kim, 2009) and linearly-solvable stochastic optimal control (Krishnamurthy and Todorov, 2010).

3.2 MRL Summary

One shortcoming of IRL that is particularly relevant to games is that it assumes no other adaptive agents exist in the environment. However, many games are multi-agent, mutually influential systems. To jointly consider the decision making processes of interacting rational agents, we need different models and techniques. In the forward direction, *multi-agent reinforcement learning* (MRL), proposed by Littman (Littman, 1994), extends RL to a multi-agent framework. Littman makes use of stochastic games (Owen, 1968) to model MRL, limiting consideration to the special case of *two-player zero-sum games*, in which one agent's gain is always the other's loss, and applies this algorithm in a simple grid-world soccer game. Hu and Wellman (Hu and Wellman, 1998) extend Littman's work, proposing a *two-player general-sum* stochastic game framework for the MRL problem. They point out that the concept of optimality loses its meaning in MRL problems since any agent's payoff depends on the action choices of others. Consequently, they adopt as a solution concept the *Nash equilibrium*, in which each agent's choice is the best response to other agents' choices. Later MRL work has focused on the development of solution concepts and methods, in competing games as well as cooperative games, including (Abdallah and Lesser, 2008; Ghavamzadeh, Mahadevan, and Makar, 2006; Patek, Beling, and Zhao, 2007; Zhao, Patek, and Beling, 2008). Representative applications include traffic control (Bazzan, 2009) and robotics (Duan, Cui, and Xu, 2012).

3.3 zero-sum MRL

The simplest MRL problem, proposed by Littman (Littman, 1994), is that two agents play a zero-sum stochastic game. Though simple enough, there are still many applications in reality. A two-person zero-sum stochastic is defined as follows:

Definition 3.1. *A two-player zero-sum stochastic game* Γ *is a* 6*-tuple* $\{S, A_1, A_2, r, P, \gamma\}$*, where* S *is the common discrete state space, which is finite;* A_k *is the discrete action space of player* k *for* $k = 1, 2$*;* $r : S \times A_1 \times A_2 \mapsto r$ *is a reward function mapping state and joint actions to a scalar, for player* 1 *(*$-r$ *is the reward for player* 2*);* $P : S \times A_1 \times A_2 \mapsto \Delta$ *is the transition probability map, where* Δ *is the set of probability distributions over state space* S *conditioning on different joint actions; and* $\gamma \in (0, 1)$ *is a reward discounted factor.*

Like Markov Decision Processes (MDP) in RL, two assumptions are implicitly made:

- **Stationary**: for every $s, s' \in S$, the transition probability from s to s' given that the players take actions $a_1 \in A_1$ and $a_2 \in A_2$, is independent of time, i.e., $p\left(s'|s, a_1, a_2, t\right) := \mathbb{P}\left(S_{t+1} = s'|S_t = s, A_1^t = a_1, A_2^t = a_2\right)$, for all $t = 0, 1, 2, ...$

- **Markovian**: for each player, the transition probability to s_t, besides A_1^t and A_2^t, only depends on s_{t-1}, i.e.,

$$\begin{aligned}&\mathbb{P}\left(S_{t+1} = s'|S_t = s, S_{t-1} = s_{t-1}, \cdots, S_0 = s_0, A_1^t = a_1, A_2^t = a_2\right)\\ =&\mathbb{P}\left(S_{t+1} = s'|S_t = s, S_{t-1} = s'_{t-1}, \cdots S_0 = s'_0, A_1^t = a_1, A_2^t = a_2\right).\end{aligned}$$

Recall that the *stationary* and *Markovian* assumptions guarantee that a traditional RL problem is solvable, i.e., a convergence solution exists for a MDP. Likewise, for MIRL, a *stationary* solution a zero-sum stochastic game, i.e.,

each player selects a stationary policy over the state space, taking the other' strategies into account, in order to maximize her expected discounted sum of rewards. The minimax equilibrium catches our eye because of the following theorem (Ferguson, 2008):

Theorem 3.2. ***Minimax Theorem.*** *For every finite two-person zero-sum game*

1. there is a number V, called the value of the game,
2. there is a mixed strategy for Player I such that $I's$ average gain is at least V no matter what II does, and
3. there is a mixed strategy for Player II such that $II's$ average loss is at most V no matter what I does.

The algorithm is described in Algorithm 1. The strength of the minimax criterion is that it allows each agent to find a "stationary" strategy that is guaranteed to exist such that a minimum payoff would be achieved no matter what its opponent behaves. Another advantage is that due to linear programming, even large size problems can be solved efficiently.

Algorithm 1 Zero-sum Q-Learning Algorithm (Littman, 1994)

Initialize:
For all $s \in \mathcal{S}, a_1 \in \mathcal{A}_1, a_2 \in \mathcal{A}_2$, let $Q(s, a_1, a_2) := 1$
For all $s \in \mathcal{S}$, let $V(s) := 1$
For all $s \in \mathcal{S}, a_1 \in \mathcal{A}_1$, let $\pi(s, a_1) := \frac{1}{|\mathcal{A}|}$
Choose an action:
With probability $explor$, return an action uniformly at random. Otherwise, if the current state is s, return action a_1 with probability $\pi_1(s, a_1)$.
Learn:
After receiving reward r_1 for moving from s to s' via action a_1 and opponent's action a_2
Let $Q_1(s, a_1, a_2) := (1 - \alpha) Q_1(s, a_1, a_2) + \alpha [r_1 + \gamma V_1(s')]$
Use linear programming to find $\pi(s, \cdot)$ such that
$\pi_1(s, \cdot) :=$ argmax $\{\pi_1'(s, \cdot), \min \{a_2', \text{sum} \{a_1', \pi(s, a_1') Q_1(s, a_1', a_2')\}\}\}$
Let $V_1(s) := \min \{a_2', \text{sum} \{a_1', \pi(s, a_1') Q_1(s, a_1', a_2')\}\}$
Let $\alpha = alpha * \text{decay}$

3.4 Nash-Q MRL

The Nash-Q algorithm, proposed by Hu and Wellman (Hu and Wellman, 1998), is an extension of Q-learning to a competitive general-sum MRL problem. Like Littman's work (Littman, 1994), a stochastic game is adopted as the basic framework in place of MDP. With an assumption that all agents, due to their rationality, will behave to achieve a Nash equilibrium, each agent maintains Q-functions over joint actions and updates the values in an exploration and exploitation fashion.

There are many algorithms to calculate Nash equilibria in a general-sum game. For a bi-matrix game, Lemke-Howson algorithm (Lemke and Howson, 1964) is most popular, and it is said to be "the best known among the combinatorial algorithms for finding a Nash equilibrium". However, it is only applicable for two player games. Though computing Nash equilibria in small games (two players and only a handful of actions for each player) using sequential algorithms is tractable, real-life situations can rarely be modeled using small size games. Rather, the involvement of many players and many actions makes finding the Nash equilibria quite difficult. In the worst case, current algorithms cannot guarantee better than exponential time to find even one Nash equilibrium.

Due to this reason, a major drawback of the Nash Q-learning algorithm is the lack of convergence guarantee. For example, if multiple Nash equilibria exist in a game, which is often the case, agents may not pick the same Nash equilibrium because they update their Q-values in a decentralized way. But authors demonstrate that if each game has a global optimal point or a saddle point which is also a Nash equilibrium, empirical convergence is able to achieve. The Nash-Q algorithm, demonstrated using two players, can be summarized in Algorithm 2 (Hu and Wellman, 1998).

Algorithm 2 Multi-agent Q-Learning Algorithm (Hu and Wellman, 1998)

Initialize:
Let $t = 0$, start from s_0
For all $s \in \mathcal{S}, a_1 \in \mathcal{A}_1, a_2 \in \mathcal{A}_2$, let $Q_1^t(s, a_1, a_2) = 1, Q_2^t(s, a_1, a_2) = 1$,
Loop:
Choose action a_1^t based on $\pi_1(s_t)$, which is a mixed strategy Nash equilibrium solution of the bimatrix game $(Q_1(s_t), Q_1(s_t))$.
Observe r_1^t, r_2^t, a_2^t and s_{t+1}
Update Q_1 and Q_2 such that
$Q_1^{t+1}(s, a_1, a_2) = (1 - \alpha_t) Q_1^t(s, a_1, a_2) + \alpha_t [r_1^t + \beta \pi_1^t(s_{t+1}) Q_1^t(s_{t+1}) \pi_2^t(s_{t+1})]$
$Q_2^{t+1}(s, a_1, a_2) = (1 - \alpha_t) Q_2^t(s, a_1, a_2) + \alpha_t [r_1^t + \gamma \pi_1^t(s_{t+1}) Q_2^t(s_{t+1}) \pi_2^t(s_{t+1})]$
Let $t := t + 1$

However, as the authors point out, this Nash-Q algorithm is not applicable to a general case because the convergence depends on certain restrictions on the bimatrix games during learning. Therefore, the application of Nash-Q learning in reality is limited. In addition, even if the convergence is achieved, it is required that every action has been tried and every state has been visited. Though it is still not close to be able to solve real applications, Nash-Q learning algorithm is an important addition to the theory of MRL.

3.5 MIRL Summary

Inverse learning problems for MRL, which we term MIRL, include the problem of estimating the game payoffs being played, given only observations of the actions taken by the players. Compared to the IRL problem, MIRL is more challenging in that it is formalized in the context of a stochastic/Markov game (Shapley, 1953; Owen, 1968) rather than a MDP. Games bring two primary challenges: First, the concept of optimality, central to MDPs, loses its meaning and must be replaced with a more general solution concept, such as the Nash equilibrium. Second, the non-uniqueness of equilibria means that in MIRL, in addition to multiple reasonable solutions for a given inversion model, there may be multiple inversion models that are all equally sensible approaches to solving the problem. IRL is a special or approximate version

of MIRL in the sense that the former treats other agents in the system as part of the environment, ignoring the difference between responsive agents and passive environment.

MIRL can be potentially useful in many real applications. For example, the use of IRL is proposed in quantitative finance, specifically, to understand the behaviour of stock trading algorithms and furthermore identify some particular type of trading (Yang et al., 2015). This treatment is reasonable in a typical trading market. Usually there are many traders involved and their activities give rise to "cancellation effects". Thus for a particular trader, she does not need to take into account what every other trader is doing (and impossible) but can regard the whole market as a noisy system. In some unusual cases, however, it may not work. For example, if there are two traders with high volumes of trades and dominate the price trend, the market can no longer be able to regarded as a passive system. Rather, one dominator has to take the other's possible strategies into account before making decisions. Obviously, MIRL fit more into in this circumstance than IRL.

Recently MIRL has attracted some interest from the machine learning research community, but the research findings are quite limited. Natarajan *et al.* address MIRL using an IRL model for multiple agents without dealing with interactions or interference among agents (Natarajan et al., 2010). Waugh *et al.* (Waugh, Ziebart, and Bagnell, 2011) contribute to the inverse equilibrium problem, but in the context of simultaneous one-stage games, rather than the sequential stochastic games that are the subject of MIRL. Reddy *et al.* (Reddy et al., 2012) use the concept of subgame perfect equilibrium (SPE) (Maskin and Tirole, 2001), a refinement of NE used in dynamic games, to address MIRL for general-sum stochastic games that have the property that each player's rewards do not depend on the actions of the others. Hadfield-Menell *et al.* (Hadfield-Menell et al., 2016) introduce a cooperative IRL problem, motivated from an autonomous system design problem, where the robot

is required to align its value with those of the humans in its environment in such a way that its actions contribute to the maximization of values for the humans. Their problem is not modeled as a MIRL problem in a stochastic game context.

Chapter 4

Zero-sum MIRL

4.1 Introduction

This chapter proposes a novel Bayesian approach to MIRL. We establish a theoretical foundation for competitive two-agent zero-sum MIRL problems and describe *Bayesian MIRL* (BMIRL), a Bayesian solution approach in which the generative model is based on an assumption that the two agents follow a minimax bi-policy. To our knowledge, this topic has not been deeply studied in the literature. Natarajan *et al.* (Natarajan et al., 2010) present an inverse reinforcement learning model for multiple agents. However, that paper does not consider competing agents or game-theoretic models, a key characteristic of our work. Waugh *et al.* (Waugh, Ziebart, and Bagnell, 2011) do consider a form of the inverse equilibrium problem. However, that paper considers simultaneous one-stage games, rather than the sequential stochastic games we consider here. A similar method, termed *decentralized MIRL* (d-MIRL), is a decentralized linear IRL approach based off work by Reddy *et al.* (Reddy et al., 2012), while our work is centralized and set in a Bayesian framework.

Several numerical experiments are performed in the setting of an abstract soccer game with simple grid structure and movement actions and probability models governing ball exchange and the outcomes of ball kicks at the goal (the agents' *shoot* action). For the inverse learning problem, the unknown rewards correspond to location of goals and player perception of a successful

shot from each position on the field. Investigation centers on relationships between the extent of prior information and the quality of learned rewards. The quality of learned rewards is measured by distance metrics in reward and probability space and by the game playing success of agents that use the rewards as the basis for an equilibrium policy. The weakest priors result in learned rewards that would give an agent using them no chance of winning the game, while the strongest priors result in learned rewards essentially as good as ground truth. Additionally, results suggest that covariance structure is more important than mean value in reward priors.

The remainder of this chapter is structured as follows: Section 4.2 introduces notation, terminology, definitions, and some basic properties needed for later work. Section 4.3 provides the main technical results, including a Bayesian framework for MIRL and formulation of a convex optimization problem for learning rewards. Section 4.4 and Section 4.5 extend the BIRL and d-MIRL approaches to the case where reward is also action dependent. Section 4.6 introduces the soccer model and compares the results generated from the three methods. Section 4.7 provides evaluation of learned rewards of our BMIRL method in terms of game playing success in simulations of the soccer game. Section 4.9 offers concluding remarks.

4.2 Preliminaries

4.2.1 Zero-sum Stochastic Games

A two-player zero-sum *discounted stochastic game* is played as follows. The game begins in one of finitely many states. There is a reward for each player. In each state, each player simultaneously selects one of finitely many actions, and hence receives a reward that associates with current state and sometimes, as well as the actions selected by one or both players. The game then makes

a stochastic transition to a new state, where the transition is dependent on the starting state and the jointly selected actions. This process is repeated over an infinite time horizon, where geometrically discounted rewards are accrued additively.

Under these rules, we can specify an instance of a two-person zero-sum discounted stochastic game in terms of the state space $\mathcal{S} = \{1, 2, \cdots, N\}$, the action spaces $\mathcal{A}_1 = \mathcal{A}_2 = \{1, 2, \cdots, M\}$ (Note that it is not required for the two agents share the same action space), two reward vectors r_1 and r_2 of the two agents involved, state transition probabilities $p\left(s'|s, a_1, a_2\right)$, and a reward discount factor $\gamma \in [0, 1)$. Reward values are assumed to be dependent on state and the actions taken by the two agents. Hence, the dimension of $r_1\left(r_2\right)$ depends on the the size of S, $\mathcal{A}_1$ and $\mathcal{A}_2$. We use $r_1\left(\cdot\right)\left(r_2\left(\cdot\right)\right)$ to denote a scalar; e.g., $r_1\left(s, a_1, a_2\right)$ represents the reward value gained by agent 1 when the two agents take actions a_1 and a_2, respectively, in state s. As it is zero-sum, $r_1\left(s, a_1, a_2\right) = -r_2\left(s, a_1, a_2\right)$. The symmetry of rewards between the two players allow to use r to denote r_1.

A solution to a stochastic game is a *bi-policy*, which provides the rules that each player follows when selecting actions at each state. Without loss of generality, a bi-policy can be specified by a collection of conditional probability mass functions π_1 and π_2, where player k selects action a^k in state s with probability $\pi^k(a^k|s)$. Each $\pi^k(\cdot|s)$ is referred to as the *strategy* played by player k in state s.

Given that each player can select from among M actions, the strategy followed by player k in state s can be represented by the $M \times 1$ vector $\pi^k\left(s\right)$. The bi-policy for state s is the set of two column vectors that denote the strategies employed by player 1 and player 2 in state s,

$$\pi\left(s\right) = \left\{\pi_1\left(s\right), \pi_2\left(s\right)\right\}.$$

In this notation, the bi-policy is defined as the set of all bi-strategies over all states,

$$\pi = \{\pi(1), \pi(2), \cdots, \pi(N)\}.$$

We use $\tilde{r}_\pi(s)$ to denote the single-stage *expected reward value* received by agent 1 at state s under bi-policy π. Then $\tilde{r}_\pi$ is a column vector with its ith component $\tilde{r}_\pi(s)$. Define $\tilde{r}_\pi(s)$ to be

$$\begin{aligned} \tilde{r}_\pi(s) &= \sum_{a_1,a_2} \pi_1(a_1|s)\,\pi_2(a_2|s)\,r(s,a_1,a_2) \\ &= [\pi_1(s)]^T\, r(s)\,\pi_2(s), \end{aligned} \tag{4.1}$$

where $r(s)$ is a $M \times M$ matrix, whose entries are independent of $\pi(s)$. We can express this relationship in matrix notation as

$$\tilde{r}_\pi = B_\pi r, \tag{4.2}$$

where B_π is a $N \times NM^2$ matrix constructed from bi-policy π, whose kth row is:

$$\left[\Phi^\pi_{1,1}(k), \Phi^\pi_{1,2}(k), \cdots, \Phi^\pi_{M,M}(k)\right],$$

where

$$\Phi^\pi_{i,j}(k) = \left[\underbrace{0,\cdots,0}_{k-1}, \phi^\pi_{i,j}(k), \underbrace{0,\cdots,0}_{N-k}\right],$$

and

$$\phi^\pi_{i,j}(k) = \pi_1(i|k)\,\pi_2(j|k).$$

The concepts of the *value function* and *Q-function* in MDPs have natural analogs in zero sum stochastic games. In particular, let us define the value function to be the bi-policy-dependent, discounted expected sum of rewards

of player 1 as a function of the initial state s:

$$V_{\pi}(s) = \sum_{t=0}^{\infty} \gamma^{t} E\left(\tilde{r}_{\pi}(s_t) | s_0 = s\right), \tag{4.3}$$

where s_t denotes the state of the game at stage t and $\tilde{r}_{\pi}^{t}$ denotes player 1's expected reward under bi-policy π at that stage. Note that the superscript t can be removed because of the Markov property. V_{π} denotes the column vector with ith component $V_{\pi}(i)$.

In addition, we define player 1's Q-function of state s and action pair (a_1, a_2), under bi-policy π, as

$$Q_{\pi}(s, a_1, a_2) = r(s, a_1, a_2) + \gamma \sum_{s'} p(s'|s, a_1, a_2) V_{\pi}(s'). \tag{4.4}$$

Over all states and actions, we can write equation (5.4) in matrix notation as

$$Q_{\pi} = r + \gamma P V_{\pi}, \tag{4.5}$$

where P is a $NM^2 \times N$ matrix with $p(s'|s, a_1, a_2)$ as its elements.

Let G_{π} denote transition matrix under bi-policy π. Specifically, G_{π} is the $N \times N$ matrix with elements

$$g_{\pi}(s'|s) = \sum_{a_1, a_2} \pi_1(a_1|s) \pi_2(a_2|s) p(s'|s, a_1, a_2). \tag{4.6}$$

Note that

$$\begin{aligned} V_{\pi}(s) &= \tilde{r}_{\pi}(s) + \sum_{t=1}^{\infty} \gamma^{t} E\left(\tilde{r}_{\pi}(s_t) | s_0 = s\right) \\ &= \tilde{r}_{\pi}(s) + \gamma \sum_{s'} g_{\pi}(s'|s) V_{\pi}(s'). \end{aligned} \tag{4.7}$$

This equation can be written in matrix notation as

$$V_{\pi} = \tilde{r}_{\pi} + \gamma G_{\pi} V_{\pi}. \tag{4.8}$$

Thus

$$V_\pi = (I - \gamma G_\pi)^{-1} B_\pi r, \tag{4.9}$$

where $(I - \gamma G_\pi)$ is always invertible for $\gamma \in [0, 1)$ since G_π is a transition matrix. The value function $V_\pi(s)$ can be expressed in terms of the Q-function as

$$V_\pi(s) = [\pi_1(s)]^T Q_\pi(s) \pi_2(s), \tag{4.10}$$

where $Q_\pi(s)$ is a $M \times M$ matrix for agent 1, whose (i, j) element is given by $Q_\pi(s, i, j)$. Note that while $Q_\pi(s)$ is a matrix, the Q_π introduced in (5.11) is an $NM^2 \times 1$ vector. We will use this relationship between the Q-function and the value function to define a *minimax bi-policy* for a stochastic game.

We will assume that rational agents playing two-player zero-sum stochastic games seek a minimax bi-policy. A minimax bi-policy is an equilibrium, in that it has the property that neither player can change the game value in their favor given that the other player holds their policy fixed. To give a precise definition of a minimax bi-policy, we will start by reviewing the notion of a minimax bi-strategy for a static game (Neumann and Morgenstern, 1944).

First consider a static (single-stage) zero-sum game, where two players simultaneously choose an action and both players receive a reward determined by the joint choice of actions. The minimax theorem states that for every two-person zero-sum game with finitely many actions, there exists a value V and a mixed strategy for each player such that

- Given player 2's strategy, the best expected reward possible for player 1 is V.
- Given player 1's strategy, the best expected reward possible for player 2 is $-V$.

As before, the strategies played by both players in a certain state s can be expressed in terms of probability mass functions $\pi_1(s)$ and $\pi_2(s)$. Expressing

the reward received by player 1 as an $M \times M$ matrix $Q_\pi(s)$, the value of the game for player 1 under a minimax bi-strategy is given by

$$\text{value}\left(Q_\pi(s)\right) = \max_{\pi_1(s)} \left\{ \min_{\pi_2(s)} \left\{ \left[\pi_1(s)\right]^T Q_\pi(s)\, \pi_2(s) \right\} \right\}.$$

A pair $\pi_1(s)$ and $\pi_2(s)$ that achieves this value is called a *minimax bi-strategy*. For zero-sum games, a minimax bi-strategy is also a Nash equilibrium.

The concept of a minimax bi-strategy can be extended to two-player discounted stochastic games via the following theorem (Shapley, 1953).

Theorem 4.1 (Shapley's Theorem)**.** *There exists a bi-policy π such that*

$$V_\pi(s) = \text{value}\left(Q_\pi(s)\right) \tag{4.11}$$

for all $s \in \mathcal{S}$.

A bi-policy that satisfies Theorem 4.1 is called a *minimax bi-policy*. For a minimax bi-policy, $V_\pi(s)$ gives the game value from each initial state $s \in \mathcal{S}$. Throughout the following sections it is assumed that agents are observed playing a game according to a minimax bi-policy and that the complete bi-policy is observable. The minimax nature of the bi-policy can then be used to infer the reward structure of the game.

4.3 Bayesian MIRL

We will formulate two-agent MIRL problems in a Bayesian setting. Bayesian methods have been widely adopted for IRL problems (Baker, Saxe, and Tenenbaum, 2009; Choi and Kim, 2011; Dimitrakakis and Rothkopf, 2011; Engel,

Mannor, and Meir, 2005; Michini and How, 2012; Qiao and Beling, 2011; Ramachandran and Amir, 2007). In a Bayesian setting, we assign a prior distribution to the reward functions. This prior distribution encodes the learner's initial belief about the reward functions before any observations are made.

Given an observed bi-policy, we can generate a point estimate of the reward function from the posterior distribution over reward functions. To construct this point estimate, we must know the likelihood of observing each bi-policy for each given reward function. So, consideration must be given to determining the appropriate likelihood function for the MIRL problem and to the development of optimization models that can be used to generate point estimates of the reward function.

The BMIRL approach we propose is a maximum a posteriori probability (MAP) estimate of reward under a likelihood function that encodes the notion of a minimax equilibrium. Let $f(r)$ denote the prior distribution on the reward of agent 1 (recalling that we denote $r = r_1$ and $r_1 = -r_2$ for zero-sum games). We will discuss the selection of prior distributions further in Section 4.6.2. Also, let $p(\pi|r)$ denote the likelihood of observing a bi-policy π when the true reward is r. Hence now our objective is to maximize $f(r|\pi)$, the posterior of rewards given an observed bi-policy, as follows,

$$f(r|\pi) \propto p(\pi|r) f(r).$$

4.3.1 Prior Distributions on Rewards

In BMIRL, we use prior distributions over reward functions to model our initial uncertainty in the reward. Although any prior may be used, in this chapter we prefer Gaussian priors for rewards. Gaussians are a reasonable choice of prior since they provide a straightforward model for representing uncertainty around a nominal choice of reward function, and have the added

benefit of leading to analytically tractable inference procedures.

Specifically, we model $r \sim \mathcal{N}(\mu_r, \Sigma_r)$, where μ_r is the mean of r and Σ_r is the covariance matrix. The probability density function of r is

$$f(r) = \frac{1}{(2\pi)^{N/2} |\Sigma_r|^{1/2}} \exp\left(-\frac{1}{2}(r-\mu_r)^T \Sigma_r^{-1}(r-\mu_r)\right). \tag{4.12}$$

4.3.2 Likelihood Function (Unique Minimax bi-policy)

To model the likelihood function $p(\pi|r)$, we assume that the bi-policy which the two agents follow is a unique minimax bi-policy given r. The likelihood is then a probability mass function given by

$$p(\pi|r) = \begin{cases} 1, & \text{if } \pi \text{ is minimax for } r \\ 0, & \text{otherwise.} \end{cases} \tag{4.13}$$

4.3.3 MAP Estimation Model

The posterior distribution of rewards for a given observed bi-policy is now

$$f(r|\pi) \propto p(\pi|r) f(r) = \begin{cases} f(r), & \text{if } \pi \text{ is minimax for } r \\ 0, & \text{otherwise.} \end{cases}$$

The MAP estimate of rewards is the vector r that maximizes $f(r|\pi)$. Thus we wish to solve the problem

$$\begin{aligned} \text{maximize:} \quad & f(r) \\ \text{subject to:} \quad & p(\pi|r) = 1. \end{aligned} \tag{4.14}$$

The remainder of this section will be devoted to developing a tractable characterization of the set of feasible r. Consider, as a first step, the class of static, single-stage, zero-sum games. In these games, minimax strategies satisfy the

conditions of the following theorem (Neumann and Morgenstern, 1944; Ferguson, 2008).

Theorem 4.2 (Minimax Theorem)**.** *Consider a two-person zero-sum game with $M \times M$ payoff matrix A. There exists a value V, a mixed strategy p for player 1, and a mixed strategy q for player 2 such that*

$$\begin{aligned} A^T p &\geq V 1_M \\ Aq &\leq V 1_M, \end{aligned} \tag{4.15}$$

where 1_M is the $M \times 1$ vector in which every element is 1. Moreover, p and q are an equilibrium bi-strategy and V is the game value if and only if (4.15) holds.

This theorem has direct implications for inverse learning problems. Consider a static game as a special case of the MIRL problem, where the goal is to recover a A such that the given bi-strategy (p, q) is a minimax bi-strategy. Hence, the linear constraints (4.15) give a characterization of the desired constraint set for a two-person zero-sum static game.

We will now extend this approach to a multi-stage stochastic game. Combining Theorem 4.1 with Theorem 4.2, a bi-policy π is a minimax bi-policy if and only if

$$\begin{aligned} \left[Q_\pi(s)\right]^T \pi_1(s) &\geq V_\pi(s) 1_M \\ Q_\pi(s) \pi_2(s) &\leq V_\pi(s) 1_M, \end{aligned} \tag{4.16}$$

for all $s \in \mathcal{S}$. The linear inequalities (4.16) provide conditions that must hold for the Q-function and value function of a stochastic game if π is a minimax bi-policy.

Since our ultimate goal is to estimate the reward function of a stochastic game, we must introduce additional constraints relating the Q-function and

value function to rewards. From (5.11) and (5.10), recall that

$$\begin{aligned} Q_\pi &= r + \gamma P V_\pi \\ V_\pi &= (I - \gamma G_\pi)^{-1} B_\pi r, \end{aligned} \tag{4.17}$$

and from (5.1), (5.2) and (5.8), we can deduce that

$$V_\pi = B_\pi Q_\pi. \tag{4.18}$$

Let $B_{\pi_1|a_2=j}$ denote the B_π obtained when π_1 is used as player 1's policy, and player 2 selects action $a_2 = j$ in all states. In this notation, the inequalities (4.16) can be expressed as

$$\begin{aligned} B_{\pi_1|a_2=j} Q_\pi &\geq B_\pi Q_\pi, \forall j \in \mathcal{A}_2 \\ B_{\pi_2|a_1=i} Q_\pi &\leq B_\pi Q_\pi, \forall i \in \mathcal{A}_1. \end{aligned} \tag{4.19}$$

Substituting the expression for V_π into the expression for Q_π in (4.17), we obtain

$$Q_\pi = r + \gamma P (I - \gamma G_\pi)^{-1} B_\pi r \tag{4.20}$$

$$= \left(I + \gamma P (I - \gamma G_\pi)^{-1} B_\pi\right) r. \tag{4.21}$$

Finally, letting

$$D_\pi = \left(I + \gamma P (I - \gamma G_\pi)^{-1} B_\pi\right), \tag{4.22}$$

the inequalities (5.34) can be expressed as

$$\begin{aligned} \left(B_{\pi_1|a_2=j} - B_\pi\right) D_\pi r &\geq 0, \forall j \in \mathcal{A}_2 \\ \left(B_{\pi_2|a_1=i} - B_\pi\right) D_\pi r &\leq 0, \forall i \in \mathcal{A}_1. \end{aligned} \tag{4.23}$$

Now we can formulate a convex quadratic program equivalent to (4.14). Recall that we use a Gaussian prior in this chapter, so the objective function

in (4.14) is log-concave. To obtain an equivalent convex optimization problem, we will instead minimize $-\ln(f(r))$. Combining (4.23) with the negative log-prior objective, the optimization problem (4.14) can be solved as the following equivalent convex quadratic program:

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}(r-\mu_r)^T \Sigma_r{}^{-1}(r-\mu_r) \\ \text{subject to:} \quad & \left(B_{\pi_2|a_1=i} - B_\pi\right) D_\pi r \leq 0 \\ & \left(B_{\pi_1|a_2=j} - B_\pi\right) D_\pi r \geq 0, \end{aligned} \tag{4.24}$$

for all $i \in \mathcal{A}_1$ and $j \in \mathcal{A}_2$.

The optimization problem (4.24) is specific to two-person zero-sum MIRL problems, which is a class of problems in which the reward value depends on both state and bi-actions. The equivalent problem for the case where reward values only depend on state is as follows:

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}(r-\mu_r)^T \Sigma_r^{-1}(r-\mu_r) \\ \text{subject to:} \quad & \left(G_\pi - G_{\pi_2|a_1=i}\right)(I-\gamma G_\pi)^{-1} r \geq 0 \\ & \left(G_\pi - G_{\pi_1|a_2=j}\right)(I-\gamma G_\pi)^{-1} r \leq 0 \end{aligned}$$

for all $i \in \mathcal{A}_1$ and $j \in \mathcal{A}_2$.

It is worth discussing the scalability of the optimization problem. When the problem size, n, is large, the inversion of the covariance matrix, which is usually sparse, is computationally expensive ($O(n^3)$). And even if we obtain the inverse of the covariance matrix (which generally will not be sparse), the objective of this problem includes $O(n^2)$ quadratic monomials, which may not fit into memory. One way to tackle this problem is to first compute the Cholesky upper-triangle factorial R of Σ, which often is sparse as Σ itself is sparse. Then let $e = R^T(r-\mu_r)$ and add it to the constraints. Finally, we rewrite our objective as $\frac{1}{2}e^T e$. This reformulation helps avoid the memory issue.

4.3.4 Discussion on Nonunique bi-policies

In the definition of the likelihood function and the convex program (4.24) we have implicitly assumed that the stochastic game has a unique minimax bi-policy. It is important to note that this assumption need not hold. Indeed for a *static* two-person zero-sum games there may exist an infinite number of minimax bi-strategies, even though each such game has a unique Nash equilibrium value. In (Ferguson, 2008), a sufficient condition for the existence of unique bi-strategy for a static matrix game is given: the square game matrix A is nonsingular and $1^T A^{-1} 1 \neq 0$.

So it is clear we must consider cases where multiple minimax bipolices exist. For ease of expression in doing so, define the following notation:

- $\mathcal{G}(r)$: the stochastic game given one agent's reward vector is r.
- $\mathcal{U}(r)$: the set of r in which the necessary condition (5.34) is satisfied.
- $\mathcal{U}^*(r)$: a subset of $\mathcal{U}(r)$ where π is a unique minimax bi-policy for $\mathcal{G}(r)$.
- $\mathcal{M}(\mathcal{U}(r))$: the optimization problem (4.24) where $r \in \mathcal{U}(r)$.
- $\mathcal{M}(\mathcal{U}^*(r))$: a subproblem of (4.24) constrained by $r \in \mathcal{U}^*(r)$.

We would like to solve the MAP problem for $\mathcal{G}(r)$ that accounts for the possibility of multiple minimax strategies. Even with a generative notion such as the idea that agents will select among equal-value equilibrium strategies with uniform probability, however, it is difficult to develop a likelihood for this problem because we cannot easily characterize the set of minimax equilibrium strategies as a function of r. As a surrogate, one might adopt $\mathcal{M}(\mathcal{U}^*(r))$, but again this problem is difficult to define directly. An alternate approach is to first solve $\mathcal{M}(\mathcal{U}(r))$. Let $\tilde{r}$ be the optimal solution to this problem. If $\tilde{r} \in U^*$ then $\tilde{r}$ is optimal for $\mathcal{M}(\mathcal{U}^*(r))$. If $\tilde{r} \notin U^*$ then form $\hat{r} = \tilde{r} + \epsilon$, for small random perturbation ϵ. With high probability $\hat{r} \in U^*(r)$ (cf. (Rudelson and Vershynin, 2014)) and will be nearly optimal for $\mathcal{M}(\mathcal{U}^*(r))$.

4.3.5 Uniqueness of bi-policy

For a *static* two-person zero-sum games there may exist multiple minimax bi-strategies, even though each such game has a unique Nash equilibrium. In (Ferguson, 2008), a sufficient condition for the existence of unique bi-strategy for a matrix game is given: the square game matrix A is nonsingular and $1^T A^{-1} 1 \neq 0$. Note that this is not the necessary condition for the existence of unique minimax bi-strategy. Rudelson and Vershynin (Rudelson and Vershynin, 2014) show that a perturbation of any fixed square matrix by a random unitary matrix is well invertible with high probability. From these findings, we can come to conclusion that in a real world two-person zero-sum MIRL problem, a unique bi-policy exists with high probability.

4.4 Linear d-MIRL

In this section, we extend the decentralized MIRL approach (Reddy et al., 2012) to a two-person zero-sum MIRL problem in which each agent's reward depends on state and the actions of both agents. As before, let r denote player 1's reward vector.

In (Reddy et al., 2012), the assumption is made that in a multi-agent system all agents reach a *Markov Perfect Equilibrium* (MPE). This implies that, for all $s \in \mathcal{S}$ and all $i \in \mathcal{A}_1$,

$$Q_\pi(s) \geqslant Q_{\pi|a_1=i}(s).$$

In (Reddy et al., 2012), rewards are selected to maximize the difference between the Q value of the observed policy and those of pure strategies, which is analogous to the classical approach to single-agent IRL given in (Ng and Russell, 2000). For our notation, the equivalent problem for agent 1 is the

following linear program:

$$\begin{aligned} \text{maximize:} \quad & \sum_{s=1}^{N} \min{}_{i \in \mathcal{A}_1} \left(\tilde{r}_\pi(s) - \tilde{r}_{\pi|a_1=i}(s) \right) \\ & + \gamma \left(G_\pi(s) - G_{\pi|a_1=i}(s) \right) (I - \gamma G_\pi)^{-1} B_\pi r \\ & - \lambda \|r\|_1 \\ \text{subject to:} \quad & \left(B_{\pi_2|a_1=i} - B_\pi \right) D_\pi r \leq 0, \end{aligned}$$

where λ is an adjustable penalty coefficient for having too many non-zero values in the reward vector.

4.5 Bayesian IRL

In this section, we will model the two-person zero-sum multi-agent inverse problem as an IRL problem, by focusing on one agent, which can be called the *agent of interest* and regarding the other agent as part of the inadaptive environment. We extend the BIRL approach developed in (Qiao and Beling, 2011), which is only applicable to state-dependent reward recovery, to our case where the reward depends on both state and the action of the agent of interest. Note that the reward we want to recover is $r(s, a_1)$ instead of $r(s, a_1, a_2)$, or $r(s, a_1, j) = r(s, a_1)$ for all $j \in \mathcal{A}_2$. Although we now turn to the MDP framework, the terminology and notation introduced in Section 4.2 will be used here, unless otherwise specified.

In (Qiao and Beling, 2011), rewards are selected to maximize the posterior of the observed state-action pairs given a reward vector r, with the likelihood being 1 if the observed actions are optimal and 0 otherwise for r. For our notation, the equivalent problem for agent 1 is the following linear program:

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2} (r - \mu_r)^T {\Sigma_r}^{-1} (r - \mu_r) \\ \text{subject to:} \quad & \left(F^{\pi_1}_{a_1=i} - C_{a_1=i} \right) r \geqslant 0, \end{aligned} \tag{4.25}$$

for all $i \in \mathcal{A}_1$, where

$$F^{\pi_1}_{a_1=i} = \left[\gamma \left(G_\pi - G_{\pi_2|a_1=i}\right)\left(I - \gamma G_\pi\right)^{-1} + I\right] C_{\pi_1},$$

and where C_{π_1} is a $N \times NM$ sparse matrix constructed from π_1, whose ith row is,

$$\left[\underbrace{0, \cdots, \pi_1(i,1), \cdots, 0}_{N}, \underbrace{\cdots}_{(M-2)N}, \underbrace{0, \cdots, \pi_1(i,M), \cdots, 0}_{N}\right],$$

and $C_{a_1=i}$ is conceptually similar to C_{π_1}, except for being constructed from a pure policy.

In the above formulation, μ_r is the mean of the unknown reward vector as a prior, and Σ_r is its covariance matrix. Note here we use the notation introduced in Section 4.6.2.

4.6 Numerical Example

In this section, we illustrate the BMIRL method developed in the previous sections on a two-player stochastic game modeled on soccer, and compare results with those obtained from d-MIRL and IRL. Though styled after soccer abstractions in (Littman, 1994), the game considered here is richer in that it models an action *shoot*, which is a direct attempt to score through a ball kick.

4.6.1 Game and Model

The game is played on a 4×5 grid as depicted in Figure 4.1. We use A and B to denote two players, and the circle in the figures to represent the ball. Each player can either stay unmoved or move to one of its neighborhood squares by taking one of 5 actions in each turn: *N* (north), *S* (south), *E* (east), *W* (west), and *stand*. If both players land on the same square in the same time period,

the ball is exchanged between the two players with some probability. In addition, the player who has the ball can *shoot*, which is to kick the ball toward their opponent's goal, with a *probability of successful shot* (PSS) distribution shown in Table 4.1. A shot can be taken from any field position, and the PSS is independent of opponent's position. It is worth noting that the PSS at one spot is the probability that the agent believes she would make a successful shot if she kicked the ball at that spot, rather than the actual probability of success she achieves during the play. Otherwise the PSS can be statistically calculated easily through observations once we have inferred the goal area by applying an appropriate MIRL approach.

In the game setting, both players act simultaneously in each time period. Player A attempts to score by reaching with the ball or shooting the ball into squares 6 or 11, and player B attempts to score by reaching with the ball or shooting the ball into squares 10 or 15. Once a point is scored or a shooting is missed, the players take the positions shown in Figure 4.1 and ball possession is assigned randomly.

As a third-party observer, we have very limited knowledge about the game they play. We know that this is a zero-sum game. We also know that both players aim to score points by taking or kicking the ball to somewhere in the field. Assume that we watch their playing sufficiently long so that we can statistically calculate their complete policies and their ball exchange rate $\beta = 0.6$ with a perfect accuracy. We will infer which squares each player must reach in order to score a point (the goal squares), as well as the PSS of each player, by means of recovering their reward vector. For example, the PSS of A in position pos ($pos = 1, 2, \cdots, 20$) equals the corresponding reward value because

$$\begin{aligned} r\left(s, a_1 = \text{kick}, a_2\right) &= 0 \times \left(1 - PSS^1_{pos}\right) + 1 \times PSS^1_{pos} \\ &= PSS^1_{pos}, \end{aligned}$$

where s is the state where A's position is pos. There are in total 800 states in this model, corresponding to the positions of the players and ball possession. Since each player has 6 different actions to choose, each one has a reward vector with a length of $800 \times 6 \times 6 = 28800$. Both players aim to maximize their own total expected points scored, subject to discount factor of $\gamma = 0.9$.

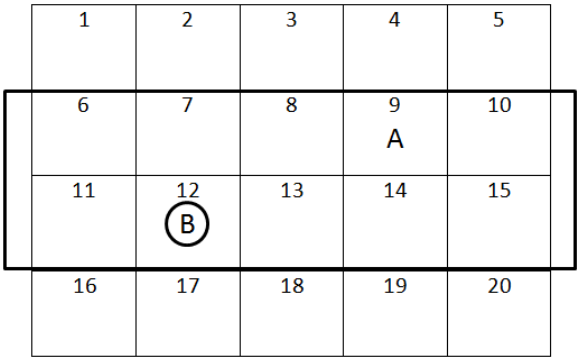


FIGURE 4.1: Soccer game: initial board

	PSS = 0.7	PSS = 0.5	PSS = 0.3	PSS = 0.1	PSS = 0
A	1, 7, 12, 16	2, 8, 13, 17	3, 9, 14, 18	4, 10, 15, 19	5, 20
	PSS = 0.7	PSS = 0.5	PSS = 0.3	PSS = 0.1	PSS = 0
B	5, 9, 14, 20	4, 8, 13, 19	3, 7, 12, 18	2, 6, 11, 17	1, 16

TABLE 4.1: Original PSS distribution of each player

It is worth mentioning that in the simulations done in Section 4.7 the PSS of the two agents happens to be symmetric. As there is some possibility this structure might give rise to confusion with the negative symmetry property of rewards, note that reward symmetry is due to the precondition of zero-sum and is unrelated to the PSS distributions of the agents. The experiments could be performed with arbitrary PSS and ball exchange probabilities.

4.6.2 Specification of Prior Information

Recall that the MIRL optimization program requires the specification of two Gaussian prior parameters for A, the mean of the rewards vector μ_r and the covariance matrix Σ_r. Below we define a concept of strength for prior information that can be expressed independently in the mean and covariance

matrix. Later subsections focus on the impact of different priors on the quality of learned rewards.

Mean of the Prior

We will use three types of mean reward vectors, namely *weak mean*, *median mean* and *strong mean*, respectively. Note that since this is a zero-sum game, the rewards assigned to B are the negatives of these rewards assigned to A.

- *Weak Mean*: we assign 0.8 point to player A in every state where A has possession of the ball and -0.8 point in every state where player B has possession of the ball;

- *Median Mean*: guessing that A's goal might be among the rightmost squares, or squares 5, 10, 15 and 20, and symmetrically, B's goal might be among the leftmost squares, or squares 1, 6, 11 and 16, we assign 1 point to A whenever A has the ball and is in the four leftmost squares, and -1 point to A whenever B has the ball and is in four rightmost squares. Also, when A has the ball and takes a shot, no matter where she is, we assign 0.5 point to A. Similarly, we assign -0.5 point to A when B has the ball and takes a shot. Otherwise, no points will be assigned to A.

- *Strong Mean*: we have a foresight to predict where the goals are for both players, but cannot make a good guess of their PSS distributions. So comparing to *median mean*, the only difference is that now the potential goal area includes only 2 squares (square 6 and 11 for A and square 10 and 15 for B), rather than 4 squares, for both players.

Covariance Matrix

The covariance matrix of the reward vector encodes our belief of the structure of the prior. Based off of our knowledge of this soccer game, we can develop two types of covariance matrices.

- *Weak Covariance Matrix*: an identity matrix, indicating that the reward vector is assumed independently distributed. This is a universal covariance matrix suitable for those MIRL problems in which we neither have knowledge of the structure of unknowns, nor want to make a guess.
- *Strong Covariance Matrix*: a more complex matrix encapsulating some internal information of the reward structure subject to our following beliefs.
    1. When A has the ball and takes a shot, the PSS depends only on A' s position in the field; likewise for B.
    2. In any state, the reward for A for any non-*shoot* action is a state-dependent constant; likewise for B.

Note that the strong covariance matrix can be constructed from the correlation matrix, by assuming that the standard deviation of each random variable in the unknown reward vector is the same. In order to avoid singularity, we will add a small perturbation α to the diagonal of the covariance matrix.

4.6.3 Results Evaluation Metric

To evaluate a recovered result, we simply compute its *average reward distance* (ARD), which is the average *Euclidean distance* from the true rewards as follows:

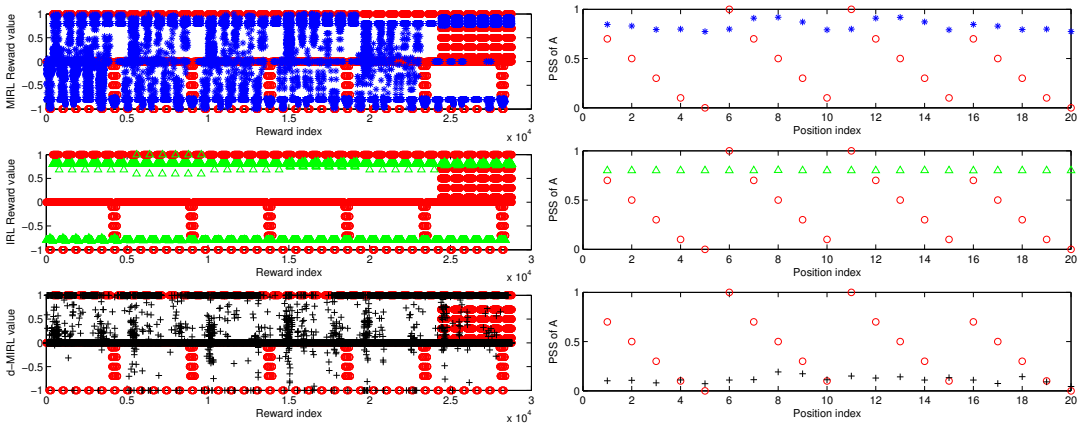
$$\begin{aligned}\text{ARD} = \Bigg\{ \frac{1}{2NM^2} \Big[\left(r_1^{\text{rec}} - r_1\right)^T \left(r_1^{\text{rec}} - r_1\right) \\ + \left(r_2^{\text{rec}} - r_2\right)^T \left(r_2^{\text{rec}} - r_2\right) \Big] \Bigg\}^{1/2},\end{aligned} \tag{4.26}$$

where the $NM^2 \times 1$ column vector r_k^{rec} and r_k denote the recovered and original reward of player k. Obviously, the smaller the ARD is, the more accurate the result is.

If only the players' PSS distributions are of interest, a similar version of the evaluation metric, termed *Average PSS Distance* (APD) can be defined as

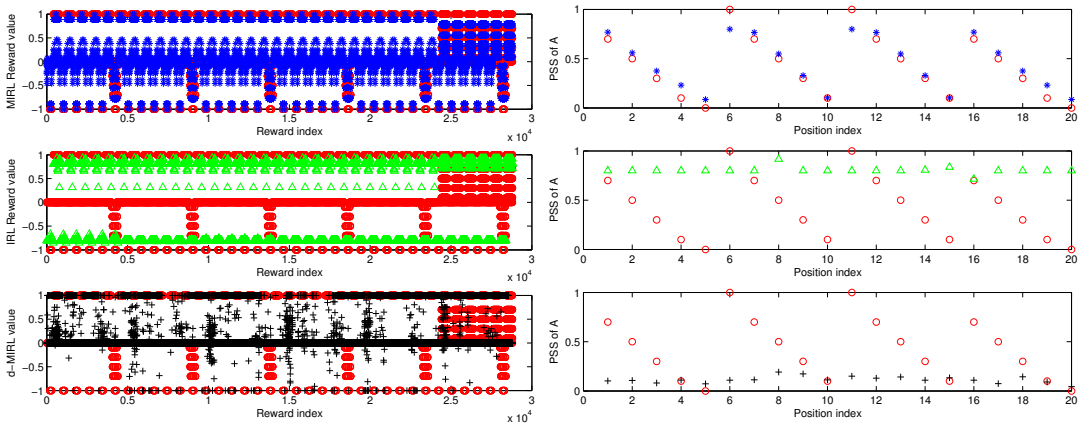
$$\text{APD} = \left\{ \frac{1}{40} \left[\sum_{i=1}^{20} \left(\theta_1^{\text{rec}}(i) - \theta_1^0(i) \right)^2 + \left(\theta_2^{\text{rec}}(i) - \theta_2^0(i) \right)^2 \right] \right\}^{1/2}, \tag{4.27}$$

where the 20×1 column vector θ_k^{rec} and θ_k^0 denote the recovered and original PSS of player k, respectively.



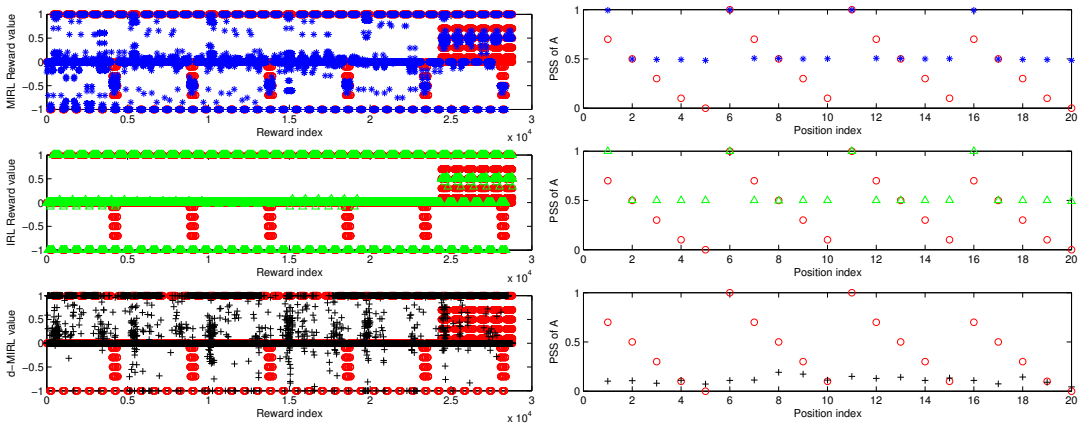
(A) Inferred Rewards (B) Inferred PSS

FIGURE 4.2: Inferred rewards and PSS: weak mean & weak covariance



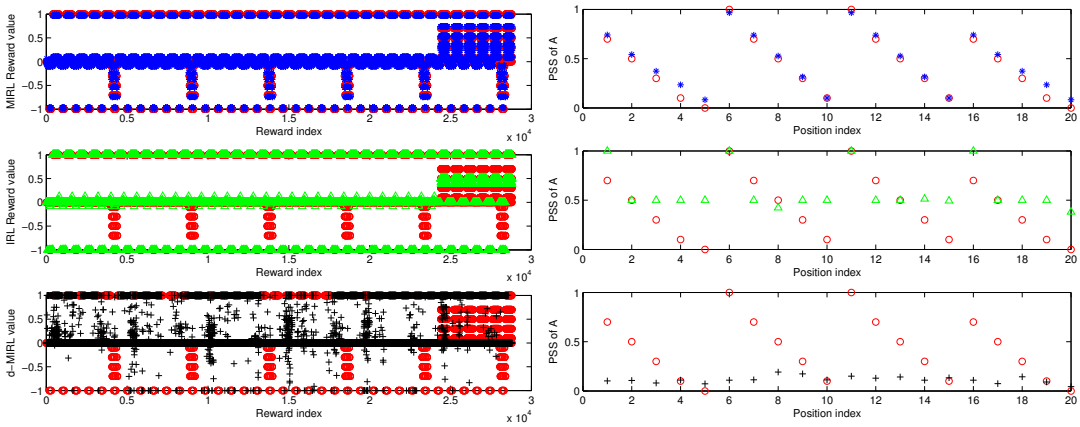
(A) Inferred Rewards (B) Inferred PSS

FIGURE 4.3: Inferred rewards and PSS: weak mean & strong covariance



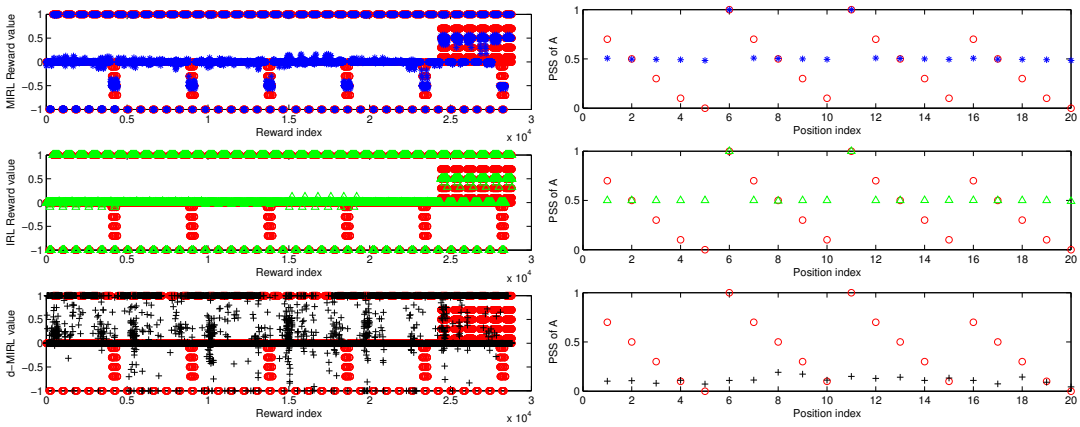
(A) Inferred Rewards (B) Inferred PSS

FIGURE 4.4: Inferred rewards and PSS: median mean & weak covariance



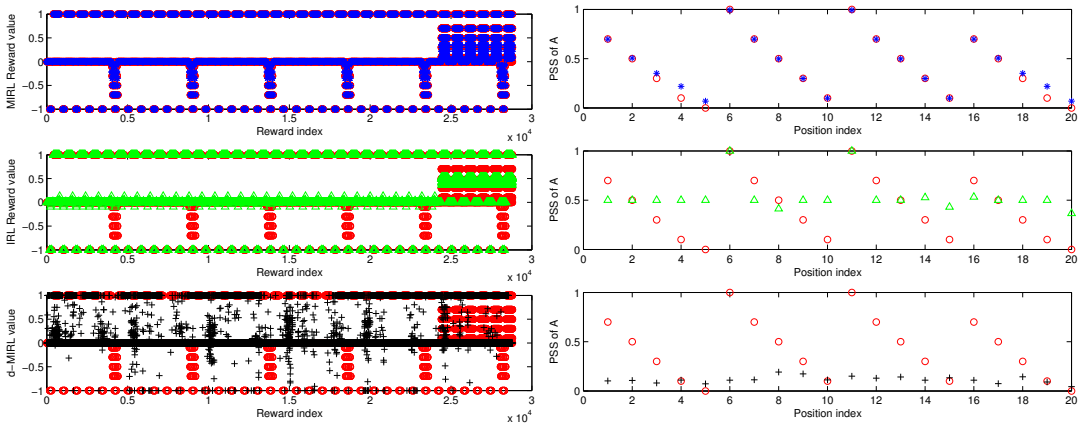
(A) Inferred Rewards (B) Inferred PSS

FIGURE 4.5: Inferred rewards and PSS: median mean & strong covariance



(A) Inferred Rewards (B) Inferred PSS

FIGURE 4.6: Inferred rewards and PSS: strong mean & weak covariance



(A) Inferred Rewards (B) Inferred PSS

FIGURE 4.7: Inferred rewards and PSS: strong mean & strong covariance

4.6.4 Results

Experiments were performed on 6 different priors formed by combining 3 different means and 2 different covariance matrices. A pertubation $\alpha = 10^{-4}$ was used in the construction of the strong covariance matrices. In all cases, the bi-policy followed by the players (the observed input to MIRL) was computed iteratively from Shapley's Theorem. Experiments on Bayesian IRL (we can also specify 6 different priors similar to those introduced in Section 4.6.2) and d-MIRL were also carried out. Note that the reward vector recovered from IRL can be extended to a MIRL reward vector by letting $r(s, a_1, j) = r(s, a_1)$ for all $j \in \mathcal{A}_2$.

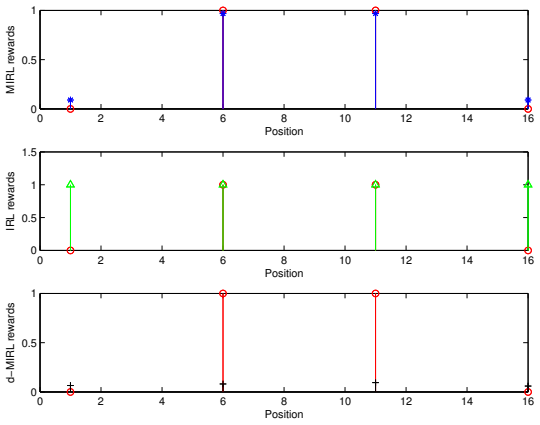
Results are shown in Figure 4.2-Figure 4.7. Take Figure 4.3 as an example. Recall that we aim to recover 28800 reward values. In each subfigure in (a), the x-axis represents the reward value index (from 1 to 22800) and the y-axis denotes the reward value. The inferred rewards of BMIRL, BIRL and d-MIRL are shown in blue stars, green triangles and black crosses, respectively, with the benchmark ground truth drawn in red circles in each subfigure. The right three subfigures in (b) show the results of A's PSSs corresponding to each case. Note that although no shots will be taken at goal positions, for

convenience, we set $PSS = 1$ for each player in their goal positions. Table 4.2 sorts each experiment with a case number, maps each case to a figure and computes the corresponding APD of the BMIRL rewards. In Case 4, we are

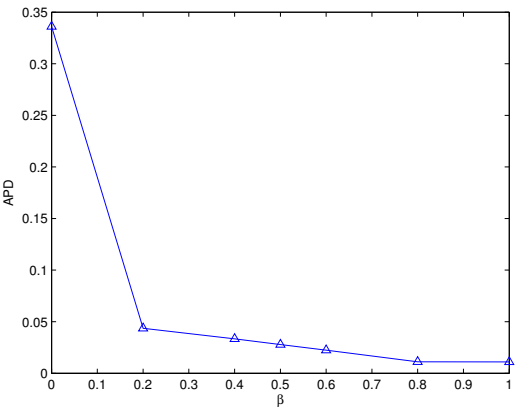
	Weak Covariance	Strong Covariance
Weak Mean	Case 1, Figure 2, 0.4535	Case 3, Figure 4, 0.0671
Median Mean	Case 3, Figure 4, 0.2169	Case 4, Figure 5, 0.0387
Strong Mean	Case 5, Figure 6, 0.2058	Case 6, Figure 7, 0.0259

TABLE 4.2: BMIRL results summary

also interested in whether the three methods can recover the actual goals for A. We calculate the average reward A receives when A is in square $1, 6, 11$ and 16. Results are shown in Figure 4.8a. Now focus on the BMIRL method. It is interesting to consider how the ball exchange rate β affects the PSS recovery result. We repeat Case 6 by changing β from 0 to 1, and calculate the APD of the inferred PSS distributions. The result is shown in Figure 4.8b.



(A) Goal recovery in Case 4



(B) APD with β changing in Case 6

4.6.5 Analysis of Results

First, we compare the results generated from the three approaches by numerically measuring the difference between the an estimated vector $\hat{x}$ and the benchmark vector x. The metric we use is *root mean squared error* (RMSE),

as the following

$$\text{RMSE} = \sqrt{\frac{\|\hat{x} - x\|_2}{\dim(x)}}$$

	WMWC	WMSC	MMWC	MMSC	SMWC	SMSC
BMIRL	0.063	0.045	0.042	0.016	0.025	0.011
BIRL	0.066	0.073	0.047	0.049	0.053	0.045
d-MIRL	0.052	0.052	0.052	0.052	0.052	0.052

TABLE 4.3: Numerical results comparison

We compute the RMSE of every recovered reward with respect to the true reward and summarize the results in Table 4.3. Note that although there are six types of BMIRL as well as BIRL due to different priors, there is only one d-MIRL reward. we can see that almost all BMIRL results are numerically closer to the ground truths comparing to the other two approaches, except when weak mean and weak convariance are selected as the prior.

Next, for the six priors we select for BMIRL, we evaluate how close every prior mean is to the recovered result using the metric of RMSE, and summarize them in Table 4.4. The purpose is to measure how much our selected priors are improved with the observed bi-policy. We can see a clear pattern: when the covariance is strong, the prior mean is shifted more to the posterior. In comparison, when the covariance is weak, the prior mean is not updated much. It is then reasonable to guess that the covariance may be more important the prior mean.

WMWC	WMSC	MMSC	MMWC	SMSC	SMWC
0.037	0.061	0.041	0.023	0.024	0.015

TABLE 4.4: Comparison between prior mean and posterior

From Figure 4.8a we see that BMIRL successfully learns the goals for A, while the other two methods fail to do so. Finally, Figure 4.8b shows that the

smaller the β is, the less accurate the recovered PSS will be. The reason is that players are inclined to dribble the ball rather than shoot it toward their opponents' goal when β is small, and consequently, observing the strategy of dribbling will not generate constraints that substantially alter the mode of the priors on shooting rewards. For example, when $\beta = 0.2$, the probability of successfully dribbling the ball to the destination for each player is, at worst, $(1-\beta)^4 = 0.407$, which means that a shot will never be taken in positions where the agent's PSS is 0.3 or 0.1.

4.7 Monte Carlo Simulation using Recovered Rewards

In the previous section, distance metrics in reward and PSS space are used to evaluate the quality of learned rewards. In this section we measure the reward quality in terms of the quality of the forward solution that would be based on the rewards. IRL is often set in the context of apprenticeship learning, in which learned rewards form the basis for anticipating or mimicking the response of agents to unknown situations. In MIRL, the analogous notion is to use learned rewards as the basis for game play in different environmental settings. In this section, we will simulate a series of games, by letting different agents use different rewards generated from the three methods discussed above and play against each other. Being rational, all agents will employ a minimax policy based off of which is the rewards they learned. Specifically, define the following agents:

- *A*, which uses true rewards;
- *B*, which uses BMIRL rewards;
- *C*, which uses BIRL rewards;

- *D*, which uses d-MIRL rewards.

A full set of agent-to-agent competition then includes the following scenarios:

- B against A;
- B against C;
- B against D.

All those games are simulated in three different environment settings, where the ball exchange rates β are 0, 0.4 and 1, respectively. Note that the symmetry of PSS values means that the two agents are equally skillful and are supposed to be equal in match if both of them follow reasonable policies generated from learned rewards.

The simulation results are presented in Table 4.5-Table 4.7. In each table, the first column is the different sets of BMIRL rewards that B employs to develop her minimax policy, where *WM*, *MM*, *SM*, *WC* and *SC* stand for *weak mean*, *median mean*, *strong mean*, *weak covariance matrix* and *strong covariance matrix*, respectively. The remaining columns are the *win or lose* (W/L) outcomes of 10000 rounds of games between B and other agents in cases where β being 0, 0.4 and 1. For example, in Table 4.5, 24.69/25.10 means B beats A with probability 24.69% and loses with probability 25.10%. It indicates that the remaining 50.21% rounds end in a tie. A tie occurs when neither player scores a point. For a more clear comparison, we only count those game episodes ending in win-lose outcomes. Each column except for the first presents B's winning percentage. Note that in Table 4.6, since there are also 6 sets of BIRL rewards, comparisons are between corresponding sets, e.g., SM-SC BMIRL vs SM-SC BIRL.

Let us coin the term *Application Metric* (AM) to refer to B's probability of winning in the soccer example. Table 4.5 shows that A outperforms or ties B

Base Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/24.80	0/62.53	0/50.30
WM & SC	24.69/25.10	25.10/25.30	50.66/49.34
MM & WC	15.28/25.34	14.36/24.69	28.44/49.43
MM & SC	24.73/25.03	24.12/25.18	49.84/50.16
SM & WC	14.85/24.52	14.94/25.50	49.31/50.69
SM & SC	24.77/25.32	24.55/25.43	49.84/50.16

TABLE 4.5: B vs A games simulation results

Base Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/0	13.50/0	0/0
WM & SC	23.36/0	24.64/0	50.29/0
MM & WC	13.55/0	15.64/0	26.82/0
MM & SC	22.73/6.74	25.45/14.80	49.55/27.58
SM & WC	15.82/0	14.27/0	49.87/0
SM & SC	23.36/0	24.64/0	50.13/0

TABLE 4.6: B vs C games simulation results

in general. This result is reasonable because A uses true rewards. In addition, we compare AM with the previous numerical metric ARD in Figure 4.9. As expected, a larger ARD results in a smaller probability of winning. What is notable is the sudden crash in probability of winning experienced when ARD becomes sufficiently large. Equivalently, the probability of B's winning drops sharply when both the mean and covariance are weak. The implication is that inferring the structure of the unknowns, is much more crucial than inferring their true values. As for the other two methods, Table 4.6-Table 4.7 show that B generally outperforms C or D.

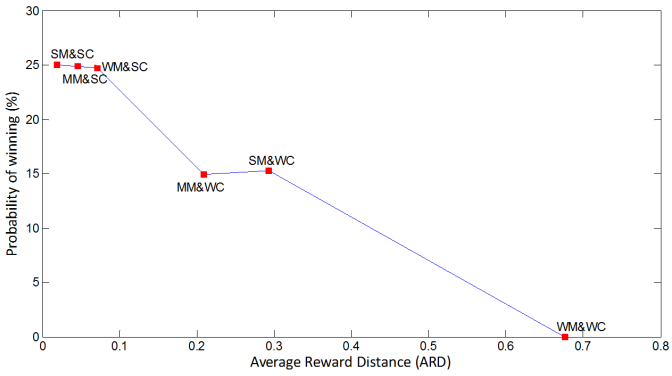


FIGURE 4.9: Two evaluation metrics comparison

Base Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/0	0/0	0/0
WM & SC	25.52/0	26.36/0	49.98/0
MM & WC	12.52/0	16.75/0	50.26/0
MM & SC	24.60/0	27.30/0	49.20/0
SM & WC	12.24/0	13.26/0	49.46/0
SM & SC	25.22/0	26.48/0	49.90/0

TABLE 4.7: B vs D games simulation results

4.8 Additional Experiments

Thus far we have demonstrated the performance of our BMIRL algorithm through a numerical experiment. There remain, however, two important questions to address. First, how does our BMIRL approach compare to supervised learning based policy learning approaches? Second, can we still expect good performance if the game is played on a larger size grid, say, $5 * 5$?

This section is dedicated to addressing these two questions through two more experiments in the context of the soccer game. The first experiment is to use multivariate linear regression to learn a linear relationship between predictors (state and the ball exchange rate) and the response (bi-strategies) and then to infer the response in a new environment. Note that normalization is needed before applying the regression. The second experiment is to redesign the game on a $5 * 5$ grid, as shown in Figure 4.10, where A and B's starting positions are 19 and 7, and their goals are 1 and 25, respectively. The PSS distributions are also re-assigned. Other settings and rules of this new game remain as they are in the old one.

Performance evaluations in these two experiments are conducted through Monte-Carlo simulation as in Section 4.7. Specifically, in the first experiment, we define agent B_p as using policy-learning method and simulate the scenario of B against B_p. In the second experiment, we investigate B_{5*5} against A_{5*5}, where B_{5*5} and A_{5*5} denote agents using BMIRL rewards and true rewards in the new game, respectively.

The results of the first experiment, presented in Table 4.8, show that BMIRL generally outperforms the policy-learning method when a strong covariance matrix is applied in the prior, and generates comparable results with those of the policy-learning method in other cases with the exception of the worst prior condition. In the second experiment, we offer more combinations of mean and covariance as prior information is very critical in the performance of BMIRL. Specifically, we provide one more *median covariance matrix*, denoted as *MC*, subject to our beliefs that: (1) when A has the ball and takes a shot, the PSS depends only on A' s position in the field; and (2) the reward for A for any non-*shoot* action is generally strongly correlated. As shown in Table 4.9, results are similar to those from the experiments reported in Table 4.5 and confirm the associated conclusions.

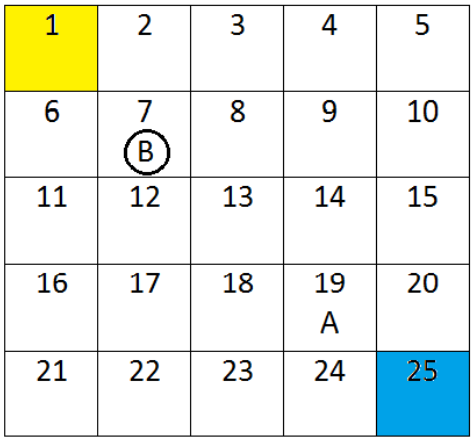


FIGURE 4.10: Soccer game: 5*5 board

Base Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/14.91	0/21.30	0/36.40
WM & SC	22.54/21.16	19.19/18.11	47.15/36.45
MM & WC	20.82/23.38	19.70/17.90	40.65/36.85
MM & SC	28.89/24.46	27.98/16.86	49.48/40.08
SM & WC	19.79/23.61	19.52/17.88	50.15/35.65
SM & SC	29.04/23.56	30.94/20.76	50.26/35.44

TABLE 4.8: B vs B_p games simulation results

Base Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	20.20/20.60	5.07/4.93	25.42/49.78
WM & MC	20.90/21.20	4.44/4.46	24.46/50.34
WM & SC	20.12/21.19	24.89/25.80	43.60/50.24
MM & WC	19.41/18.79	4.17/4.23	24.19/49.11
MM & MC	20.94/20.86	5.32/5.28	25.56/49.94
MM & SC	21.02/20.60	24.27/24.62	43.60/49.98
SM & WC	20.02/20.88	5.23/5.27	25.06/51.34
SM & MC	20.03/20.07	4.23/4.37	26.14/51.06
SM & SC	24.81/25.78	25.32/24.72	49.84/50.16

TABLE 4.9: B_{5*5} vs A_{5*5} games simulation results

4.9 Conclusions

This chapter introduces the MIRL problem in the setting of zero-sum stochastic games and presents a solution based on Bayesian inference. Although it seems that MIRL is a natural extension of IRL, it in fact presents more challenges. Even in simple static games two important distinctions between inverse learning for optimization and inverse learning for games emerge. While the model in this chapter assumes that the complete bi-policy of two players is observed, it is more likely that only actions of the individual players are observed. In an optimization setting, since deterministic policies are assumed, strategies can be inferred exactly from finitely many observations of actions. In the case of games, strategies are often mixed, and so strategies cannot be inferred exactly from finitely many observations of the actions taken in each state. Therefore, we cannot model a player's strategy as an observation as it can be done in IRL. In the setting of games, strategies must be treated as latent variables that are not observed directly, but bridge the gap between reward functions and observable actions.

Though ideally structured, the numerical examples considered in this section serve to demonstrate the ill-specified nature of the MIRL problem. Neither BIRL nor d-MIRL perform satisfactorily on the numerical examples. The

rationale underlying this phenomenon is that there always exist multiple feasible solutions that are consistent with the observations. It is extremely difficult to select a reward function that is closest to the ground truth without a certain amount of domain knowledge. Our proposed BMIRL approach makes use of domain knowledge expressed as priors on the reward function. That distinction, new to the literature of MIRL methods, is why our Bayesian method is superior to the d-MIRL method in the numerical examples. Fortunately, in many real problems domain knowledge would be available to observers.

A principal motivation for the study of MIRL in game settings is that the approach offers insight into how agents will behave if the game environment, rules, or dynamics change. Such insight may be useful in game design and management, such as balance adjustment. Effective supervised methods exist for learning policies from observed actions, but policies learned in this fashion do not project into new game environments. The reason is that the optimal policy often changes with environment and hence learning from an old policy may not help to infer a new policy. To see this, consider the abstract soccer game. In Section 4.7, three additional agents B, C and D come up with their own minimax policies by using rewards learned from three different methods, and compete with A in three different environmental settings: the ball exchange rate $\beta = 0, 0.4$ and 1. Recall that rewards were learned when $\beta = 0.6$. The similarity of two policies, say p_1 and p_2, can be measured using the *Frobenius distance* F, defined as: $F_{p_1,p_2} = \sqrt{\text{tr}\left((p_1 - p_2)(p_1 - p_2)'\right)}$. Table 4.10 shows the similarity of player B's policies as a function of β. The conclusion to be drawn is that as the environment changes, so does the policy.

	$\beta = 0.4$	$\beta = 1$	$\beta = 0$
$F_{\beta,0.6}$	5.71	8.53	20.49

TABLE 4.10: Policy difference w.r.t. β

Chapter 5

General-sum MIRL

5.1 Introduction

In this chapter, we consider five special classes of two-person general-sum MIRL problems, *uCS-MIRL*, *advE-MIRL*, *cooE-MIRL*, *uCE-MIRL*, and *uNE-MIRL*, each distinguished by its solution concept. The first problem, uCS-MIRL, is a *cooperative game* in which the agents employ *cooperative strategies* (CSs) that aim to maximize the sum of their value functions, or the *total game value*. The second and third problems consider circumstances that two player constitute two very special and unique NEs: advE is in general a *win-or-lose* equilibrium, but not necessarily for a zero-sum game; cooE is such an equilibrium that players maximize their own payoffs by “coordinating" with others. In the fourth problem, uCE-MIRL, the agents are assumed to follow strategies that constitute a *utilitarian correlated equilibrium* (uCE), which achieves the maximum total game value among all CEs. In the last problem, uNE-MIRL, are assumed to follow strategies that constitute a NE that maximizes total game value. These five MIRL variants are motivated from real applications and hence worth studies. On the one hand, uCS, uCE and uNE are such equilibriums where agents try to achieve a *socially efficient* outcome, with or without certain constraints, that maximizes the sum of their value functions, which is a *Pareto optimum*, meaning that it is not possible to make one player better off without also making the other player worse off (Barr, 2012). They

are particular of interest in welfare economics, in which policy makers try to design rules of games to achieve Pareto optimum in social welfare. On the other hand, though advE/cooE-MIRL is less usual due to the possible non-existence of advE/cooE, they are still useful in practice. For example, consider an example where two power suppliers compete with each other in the local market. Though it is a competitive game, the outcome is less likely dominate-or-exit. Hence it might be more reasonable to formulate the problem as an advE-MIRL than zero-sum MIRL. As for cooE, one classic academic example, the Stag Hunt (Skyrms, 2004), highlights its value of investigation: there are two hunters, each can chose to hunt hare or stag, with symmetric payoffs. If they both hunt stag(hare), they both will get a payoff of 2(1); and if their targets are different, the one who hunts stag will fail to get anything and the other will get a payoff of 1. In this game, (stag, stag) is a cooE.

We propose novel approaches to address these five problems under the assumption that the game observer knows the policies and solution concepts for the players. For uCS/advE/cooE-MIRL, we first develop a characteristic set of solutions ensuring that the observed bi-policy is a corresponding strategy/equilibrium and then apply a Bayesian inverse learning method. For uCE-MIRL, we develop a linear programming problem, subject to constraints that define necessary and sufficient conditions for the observed policies to be CE. For the objective function, we propose novel heuristics to choose a solution that not only minimizes the total game value difference between the observed bi-policy and its *local* uCS, but also maximizes the scale of the solution. We apply a similar treatment to the problem of uNE-MIRL.

The remainder of this chapter is structured as follows: Section 5.2 introduces notations, terminologies and definitions that will be used throughout this chapter, as well as some basic game theory equilibrium concepts through some examples. Section 5.3 summarizes several conventional MIRL

algorithms. Section 5.4 provides the main technical work, developing different approaches for different problems to learn rewards. Section 5.5 and Section 5.6 demonstrate our algorithms through several benchmark experiments. Section 5.7 offers concluding remarks.

5.2 Preliminaries

This section serves two purposes: 1. introduce concepts/notations of MRL that will be used throughout this chapter, and; 2. explain some game theory concepts through examples and manifest their properties mathematically in the context of two-person general-sum. To make it simple for presentation, we restrict our attention to the two-player general-sum case.

5.2.1 General-sum Stochastic Game

A two-player general-sum discounted stochastic game is a tuple $\{\mathcal{S}, \mathcal{A}_i, R_i, P, \gamma\}$, where $\mathcal{S}$ is the common state space for all players, $\mathcal{A}_i$ and R_i are the action space and reward for player i, respectively. P is the probabilistic function controlling state transitions, conditioned on the past state and joint actions. $\gamma \in [0, 1)$ is a reward discount factor. In this chapter, we assume that both players share the same action space. The state and action spaces are both finite, i.e., $|\mathcal{S}| = N$ and $|\mathcal{A}_i| = M$. A stochastic game is a sequence of single-stage games, or *subgames*, induced in every state $s \in \mathcal{S}$, such that both players need to determine an *individual strategy* $\pi_i(s)$ or negotiate a *bi-strategy* $\pi(s)$ that guides their actions in every subgame. The collection of all bi-strategies is a *bi-policy* π. Note that an individual strategy can be a mixed strategy, which is a probability distribution over all available actions. We define a *pure* bi-strategy $a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$ as a bi-strategy where both players select deterministic actions. Each player's reward values are assumed dependent on state and possibly, bi-strategies, but are independent of each other.

5.2.2 MRL

Let $\tilde{r}_i^{\pi}(s)$ be the *expected reward value* received by agent i at state s under bi-policy π, specifically,

$$\begin{aligned}\tilde{r}_i^{\pi}(s) &= \sum_a \pi_1(a_1|s)\,\pi_2(a_2|s)\,R_i(s,a) \\ &= [\pi_1(s)]^T R_i(s)\,\pi_2(s), \forall s \in \mathcal{S},\end{aligned} \tag{5.1}$$

where a is a pure bi-strategy, $\pi_i(s)$ is a $M \times 1$ vector denoting the probability distribution over actions in state s. $R_i(s)$ is a $M \times M$ matrix, each entry of which denotes a pure bi-strategy dependent reward value. Structuring all $R_i(s,a)$ into a column vector as r_i, we can simplify and represent (5.1) in a matrix notation as

$$\tilde{r}_i^{\pi} = B_{\pi} r_i. \tag{5.2}$$

The linear transformation operator B_{π} is a $N \times NM^2$ matrix constructed from π, whose kth row is:

$$\left[\Phi_{1,1}^{\pi}(k), \Phi_{1,2}^{\pi}(k), \cdots, \Phi_{M,M}^{\pi}(k)\right],$$

where

$$\Phi_{i,j}^{\pi}(k) = \left[\underbrace{0, \cdots, 0}_{k-1}, \phi_{i,j}^{\pi}(k), \underbrace{0, \cdots, 0}_{N-k}\right],$$

and

$$\phi_{i,j}^{\pi}(k) = \pi^1(i|k)\,\pi^2(j|k).$$

Player i's *value function*, starting at state s and under π, is defined as

$$V_i^{\pi}(s) = \sum_{t=0}^{\infty} \gamma^t E\left(\tilde{r}_i^{\pi}(s_t)\,|s_0 = s\right), \tag{5.3}$$

and its Q-function, upon s and a, is

$$\begin{aligned} Q_i^{\pi}(s,a) &= r_i(s,a) + \gamma \sum_{s'} p(s'|s,a) V_i^{\pi}(s') \\ &= r_i(s,a) + \gamma P_{s,a} V_i^{\pi}. \end{aligned} \tag{5.4}$$

A major difference between RL/IRL and MRL/MIRL, is the definition of the *value function*. In MRL/MIRL,

$$V_i^{\pi}(s) \in \text{solution concept}_i\left(Q_1^{\pi}(s), Q_2^{\pi}(s)\right), \forall s \in \mathcal{S} \tag{5.5}$$

Theoretically and empirically, players are free to employ any solution concept.

Let G_{π} denote a transition matrix under bi-policy π. Specifically, G_{π} is the $N \times N$ matrix with elements

$$g_{\pi}(s'|s) = \sum_{a} \pi_1(a_1|s)\, \pi_2(a_2|s)\, p(s'|s,a). \tag{5.6}$$

Then

$$V_i^{\pi}(s) = \tilde{r}_i^{\pi}(s) + \gamma \sum_{s'} g_{\pi}(s'|s)\, V_i^{\pi}(s'). \tag{5.7}$$

In addition, $V_i^{\pi}(s)$ can also be expressed in terms of the Q-function as

$$V_i^{\pi}(s) = \left[\pi_1(s)\right]^T Q_i^{\pi}(s)\, \pi_2(s), \tag{5.8}$$

where $Q_i^{\pi}(s)$ is a $M \times M$ matrix. We can rewrite (5.7) in matrix notation as

$$V_i^{\pi} = \tilde{r}_i^{\pi} + \gamma G_{\pi} V_i^{\pi}. \tag{5.9}$$

Thus

$$V_i^{\pi} = (I - \gamma G_{\pi})^{-1} B_{\pi} r_i, \tag{5.10}$$

where $(I - \gamma G_\pi)$ is always invertible for $\gamma \in [0, 1)$ since G_π is a transition matrix. Restructuring $Q_i^\pi(s, a)$ into a column vector, denoting $\vec{Q}_i^\pi$, we can rewrite equation (5.4) in matrix notation, over all states and joint actions, as

$$\vec{Q}_i^\pi = r_i + \gamma P V_i^\pi, \tag{5.11}$$

where P is a $NM^2 \times N$ matrix with $p(s'|s, a)$ as its elements. Combining (5.11) and (5.10) leads to

$$\begin{aligned} \vec{Q}_i^\pi &= r_i + \gamma P (I - \gamma G_\pi)^{-1} B_\pi r_i & (5.12) \\ &= \left(I + \gamma P (I - \gamma G_\pi)^{-1} B_\pi\right) r_i. & (5.13) \end{aligned}$$

In addition, (5.8) can be rewritten more compactly as

$$V_i^\pi = B_\pi \vec{Q}_i^\pi. \tag{5.14}$$

Lastly, we define the *total game value* of a two-player stochastic game starting at state s, under a bi-policy π, $V^\pi(s)$, as the sum of the value functions of both players, i.e., $V^\pi(s) = V_1^\pi(s) + V_2^\pi(s)$.

5.2.3 Cooperative Strategy

Both NE and CE introduced previously are equilibriums of *competitive games*. In a cooperative game, an agreement over a joint strategy of players can be called a *cooperative strategy* (CS). A *Characteristic Function* v defines the type of cooperation between players (Ferguson, 2008), and for a two-player single-stage game (state s), can be defined as

$$v(s, a) = \text{Val}\left(R_1(s, a), R_2(s, a)\right), a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2. \tag{5.15}$$

$\text{Val}(\cdot)$ is self-defined, based on the type of cooperation.

5.3 Conventional MIRL Approaches

Before diving into our new algorithms, we introduces several approaches to a MIRL problem. The first one is a *decentralized* MIRL (d-MIRL) algorithm developed by Reddy *et al.*, where all agents are assumed to follow a Nash equilibrium at every single game. The key idea is to find reward that maximize the difference between the Q value of the observed policy and those of pure strategies, which is analogous to the classical approach to single-agent IRL given in (Ng and Russell, 2000). Though in their original algorithm version reward is assumed dependent only on state, we can extend it to treat action dependency as well. Using our notations, the d-MIRL approach to a two-person general-sum MIRL problem, take player 1 as an example, is to solve the following linear program:

$$\begin{aligned} \text{maximize:} \quad & \sum_{s=1}^{N} \min{}_{a_1} \left(\tilde{r}_1^{\pi}(s) - \tilde{r}_1^{\pi|a_1}(s) \right) \\ & + \gamma \left(G_{\pi}(s) - G_{\pi|a_1}(s) \right) \left(I - \gamma G_{\pi} \right)^{-1} B_{\pi} r_1 \\ & - \lambda \left\| r_1 \right\|_1 \\ \text{subject to:} \quad & \left(B_{\pi|a_1} - B_{\pi} \right) D_{\pi} r_1 \leq 0 \end{aligned}$$

where λ is an adjustable penalty coefficient for having too many non-zero values in the reward vector.

The key idea of the second approach is to model a two-person general-sum MIRL as an IRL problem. This approach requires us to select one player (e.g. player 1) and treat the other as part of the passive environment. We extend the Bayesian IRL (BIRL) approach developed in (Qiao and Beling, 2011), which is only applicable to state-dependent reward recovery, to involve action-dependence cases. Note that the reward can be recovered is $R_1(s, a_1)$ instead of $R_1(s, a_1, a_2)$, as player 2 is not considered adaptive. That's

to say, $R_1(s, a_1, a_2) = R_1(s, a_1)$ for all $a_2 \in \mathcal{A}_2$. Using our notation, the algorithm to recover player 1's reward is:

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}(r_1 - \mu_{r_1})^T \Sigma_{r_1}^{-1}(r_1 - \mu_{r_1}) \\ \text{subject to:} \quad & \left(F_{a_1}^{\pi_1} - C_{a_1}\right) r_1 \geqslant 0, \end{aligned} \tag{5.16}$$

for all $a_1 \in \mathcal{A}_1$, where

$$F_{a_1}^{\pi_1} = \left[\gamma\left(G_\pi - G_{\pi_1|a_1}\right)(I - \gamma G_\pi)^{-1} + I\right] C_{\pi_1},$$

and C_{π_1} is a $N \times NM$ sparse matrix constructed from π_1, whose ith row is,

$$\left[\underbrace{0, \cdots, \pi^1(i,1), \cdots, 0}_{N}, \underbrace{\cdots}_{(M-2)N}, \underbrace{0, \cdots, \pi^1(i,M), \cdots, 0}_{N}\right],$$

and C_{a_1} is conceptually similar to C_{π_1}, except for being constructed from a pure strategy a_1 for all states.

In fact, the BIRL approach, strictly speaking, is not an algorithm but just a treatment of MIRL problems. It is worth attention, however, as people might wonder if MIRL can be "covered" by IRL. Obviously, the topic of MIRL will lose importance if the answer is yes.

The third approach is not applicable to a general MIRL problem but a restricted family: zero-sum. That algorithm to recover one player' reward vector (assuming the other player's reward is additive inverse) is

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}(r - \mu_r)^T \Sigma_r^{-1}(r - \mu_r) \\ \text{subject to:} \quad & \left(B_{\pi|a_1} - B_\pi\right) D_\pi r \leq 0 \\ & \left(B_{\pi|a_2} - B_\pi\right) D_\pi r \geq 0, \end{aligned} \tag{5.17}$$

for all $a_1 \in \mathcal{A}_1$ and $a_2 \in \mathcal{A}_2$. More details can be found in (Lin, Beling, and Cogill, 2017).

These three approaches will be revisited as benchmarks in later sections.

5.4 MIRL Model Development

This section proposes five two-player general-sum MIRL problems and corresponding approaches to them. We first informally define a MIRL problem: a two-player general-sum MIRL problem is such a problem that given $\{\mathcal{S}, \mathcal{A}_i, P, \gamma, \mathcal{O}\}$, find both players' rewards r_i that can explain the observed behaviors. Here $\mathcal{O}$ is an observation of the game play that is used to estimate the bi-policy (and P as well when P is not explicitly known). However, in this chapter, we assume that π is already available to us in place of $\mathcal{O}$.

The MRL literature suggests that an agreement over a specific solution concept may be needed to solve a MRL problem. Similarly, in our approaches to MIRL, one basic assumption is required: both players agree on a specific strategy/equilibrium to play and this information is available to us. Armed with some basic knowledge of game theory introduced in Section 2.2.2 and Section 2.2.3, we focus our attention to the following five interesting strategies/equilibriums and build models from them.

1. ***utilitarian* Cooperative Strategy** (uCS). In (5.15), we only consider $\text{Val}(\cdot) = \sum(\cdot)$. A single-stage game in state s and taking action a is a *utilitarian* cooperative strategy (uCS) if and only if

$$\sum_i R_i(s, a) \geq \sum_i R_i(s, a'), a' \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \setminus a. \tag{5.18}$$

2. **Adversarial Equilibrium** (advE). It is a variant of NE (Littman, 2001; Hu and Wellman, 1998). In addition to the property that NE has, an advE has another feature that no player is hurt by any change of others. That's to say, in a two-player single-stage game (state s), $\pi(s)$ is an advE

if and only if, in addition to (2.1),

$$R_i\left(s, \pi_i\left(s\right), \pi_{-i}\left(s\right)\right) \leq R_i\left(s, \pi_i\left(s\right), \pi'_{-i}\left(s\right)\right), \pi'_{-i}\left(s\right) \in \Pi_{-i} \setminus \pi_{-i}\left(s\right), \tag{5.19}$$

3. **Coordination Equilibrium** (cooE). Similar to advE, *coordination equilibrium* (cooE) is also a variant of NE (Littman, 2001; Hu and Wellman, 1998), in the sense that all players' maximum expected payoffs are achieved given all of them employ a cooE. Specifically, $\pi\left(s\right)$ is a cooE if and only if, in addition to (2.1),

$$R_i\left(s, \pi\left(s\right)\right) \geq R_i\left(s, \pi'\left(s\right)\right). \tag{5.20}$$

4. ***utilitarian* Correlated Equilibrium** (uCE). We borrow the concept of *utilitarian* correlated equilibrium (uCE) from (Greenwald and Hall, 2003) and state that in a two-player single-stage game (state s), $\pi\left(s\right)$ is a uCE if and only if,

$$\Sigma_i R_i\left(s, \pi\left(s\right)\right) \geq \Sigma_i R_i\left(s, \check{\pi}\left(s\right)\right), \pi'\left(s\right) \in \Pi_{CE} \setminus \pi\left(s\right). \tag{5.21}$$

5. ***utilitarian* Nash Equilibrium** (uNE). Similar to uCE, in a two-player single-stage game (state s), a NE $\pi\left(s\right)$ is a *utilitarian* Nash equilibrium (uNE) if and only if

$$\Sigma_i R_i\left(s, \pi\left(s\right)\right) \geq \Sigma_i R_i\left(s, \pi'\left(s\right)\right), \pi'\left(s\right) \in \Pi_{NE} \setminus \pi\left(s\right). \tag{5.22}$$

Among the above five equilibriums, it is easy to show that uCS always exists and unique in a cooperative game. In a noncooperative game, advE and cooE are shown to be unique, though they are essentially NEs (Hu and Wellman, 1998; Littman, 2001). However, neither of them is guaranteed to exist (Hu

and Wellman, 1998; Littman, 2001). In comparison, it is easy to conclude that uNE and uCE always exist and unique in any noncooperative game.

Readers who are new to game theory may get confused with cooE and uCS. Intuitively, cooE is a special Nash equilibrium, which means that agents are essentially selfish. However, they are *forced* to cooperate in order to maximize their own benefits. In contrast, when following a CS, agents cooperate with each other *actively* and even prepare to sacrifice their own benefits if necessary. Section 5.5 will help illustrate their difference.

5.4.1 Extension to stochastic games

Filar and Vrieze (Filar and Vrieze, 1996) show how the Q function links a stochastic game and a single stage game: treat the Q functions at each state as payoffs for single stage games, and the stochastic game is *in an equilibrium* if and only if the overall multi-stage strategies are in equilibrium. We now extend our definitions of the five strategies/equilibriums from a single game to a two-player stochastic game, as follows,

Definition 5.1. *A bi-policy π is a uCS/advE/cooE/uNE/uCE of a two-player stochastic game $\mathcal{G}$ if only if $\pi(s)$ is a uCS/advE/CooE/uNE/uCE of its sub-game $\mathcal{G}(s)$, for all $s \in \mathcal{S}$.*

Correspondingly, we define that a uCS-MIRL/advE-MIRL/CooE-MIRL/uNE-MIRL/uCE-MIRL problem is such a MIRL problem that players employ a uCS/advE/CooE/uNE/uCE.

5.4.2 uCS-MIRL

One main result characterizing the set of solutions to a two-player uCS-MIRL problem is the following:

Theorem 5.2. *Given a two-player stochastic game* $\{\mathcal{S}, \mathcal{A}_i, r_i, P, \gamma\}$*, the observed bi-policy* π *is a uCS if and only if*

$$\left(B_\pi - B_a\right) D_\pi \left(r_1 + r_2\right) \geq 0, a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \tag{5.23}$$

where $D_\pi = I + \gamma P \left(I - \gamma G_\pi\right)^{-1} B_\pi$*.* B_a *is obtained from such a bi-policy that players employ the bi-strategy* a *in all states.*

Proof. According to the definition of uCS, π is a uCS if and only if, for any state s and pure bi-strategy $a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$, we have

$$\begin{aligned}
&\pi\left(s\right) \in \arg\max_{a \in \mathcal{A}} \sum_i Q_i^\pi\left(s, a\right) \\
\Leftrightarrow &\sum_i Q_i^\pi\left(s, \pi\left(s\right)\right) \geq \sum_i Q_i^\pi\left(s, a\right) \\
\Leftrightarrow &r_1\left(s, \pi\left(s\right)\right) + r_2\left(s, \pi\left(s\right)\right) + \gamma P_{s,\pi(s)}\left(V_1^\pi + V_2^\pi\right) \\
&\geq r_1\left(s, a\right) + r_2\left(s, a\right) + \gamma P_{s,a}\left(V_1^\pi + V_2^\pi\right) \\
\Leftrightarrow &B_\pi\left(r_1 + r_2\right) + \gamma B_\pi P \left(I - \gamma G_\pi\right)^{-1} B_\pi\left(r_1 + r_2\right) \\
&\geq B_a\left(r_1 + r_2\right) + \gamma B_a P \left(I - \gamma G_\pi\right)^{-1} B_\pi\left(r_1 + r_2\right) \\
\Leftrightarrow &\left(B_\pi - B_a\right)\left(I + \gamma P \left(I - \gamma G_\pi\right)^{-1} B_\pi\right)\left(r_1 + r_2\right) \geq 0 \\
\Leftrightarrow &\left(B_\pi - B_a\right) D_\pi \left(r_1 + r_2\right) \geq 0
\end{aligned} \tag{5.24}$$

□

Since any solution that is consistent with (5.23) ensures a unique uCS, we can borrow the idea introduced in (Lin, Beling, and Cogill, 2017) and propose a Bayesian approach. The general idea is to maximize the posterior probability of the inferred rewards $p\left(r_1, r_2 | \pi\right)$,

$$p\left(r_1, r_2 | \pi\right) \propto f\left(r_1, r_2\right) p\left(\pi | r_1, r_2\right), \tag{5.25}$$

where $p(\pi|r_1, r_2)$ is the likelihood of observing π given r_1 and r_2 and $f(r_1, r_2)$ is a joint prior of r_1 and r_2 that we need to specify. Recall our "reward independence" assumption, which is

$$f(r_1, r_2) = f(r_1) f(r_2), \tag{5.26}$$

we can specify the prior over r_1 and r_2 independently. Particularly, we prefer a Gaussian prior for both, $r_i \sim \mathcal{N}(\mu_{r_i}, \Sigma_{r_i})$, where μ_{r_i} is the mean of r_i and Σ_{r_i} is the covariance. The PDF of r_i is

$$f(r_i) = \frac{1}{(2\pi)^{N/2} |\Sigma_{r_i}|^{1/2}} \exp\left(-\frac{1}{2}(r_i - \mu_{r_i})^T \Sigma_{r_i}^{-1}(r_i - \mu_{r_i})\right), i = 1, 2. \tag{5.27}$$

To model the likelihood function $p(\pi|r_1, r_2)$, we assume that the bi-policy which the two agents follow is a unique uCS given r_1, r_2. The likelihood is then a probability mass function given by

$$p(\pi|r_1, r_2) = \begin{cases} 1, & \text{if } \pi \text{ is uCS for } r_1, r_2 \\ 0, & \text{otherwise.} \end{cases} \tag{5.28}$$

Putting it together, we formulate the optimization problem for uCS-MIRL as,

$$\begin{aligned} \text{maximize:} \quad & f(r_1, r_2) \\ \text{subject to:} \quad & p(\pi|r_1, r_2) = 1. \end{aligned} \tag{5.29}$$

or, equivalently,

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2} \sum_i (r_i - \mu_{r_i})^T \Sigma_{r_i}^{-1} (r_i - \mu_{r_i}) \\ \text{subject to:} \quad & \left(B_\pi - B_{\pi|a}\right) D_\pi (r_1 + r_2) \geq 0, a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \end{aligned} \tag{5.30}$$

5.4.3 advE-MIRL

The main result characterizing the set of solutions to a two-player advE-MIRL problem is the following:

Theorem 5.3. *Given a two-player stochastic game* $\{\mathcal{S}, \mathcal{A}_i, r_i, P, \gamma\}$*, the observed bi-policy* π *is an advE if and only if*

$$\begin{aligned}
\left(B_{\pi|a_1} - B_\pi\right) D_\pi r_1 \leq 0, \forall a_1 \in \mathcal{A}_1 \\
\left(B_{\pi|a_2} - B_\pi\right) D_\pi r_2 \leq 0, \forall a_2 \in \mathcal{A}_2 \\
\left(B_{\pi|a_1} - B_\pi\right) D_\pi r_2 \geq 0, \forall a_1 \in \mathcal{A}_1 \\
\left(B_{\pi|a_2} - B_\pi\right) D_\pi r_1 \geq 0, \forall a_2 \in \mathcal{A}_2,
\end{aligned} \tag{5.31}$$

where $B_{\pi|a_1}$ *is obtained from such a bi-policy that player 2 employs her original policy while player 1 always chooses action* a_1 *in any state (game).*

Proof. Eqs (5.31) contains four inequalities. In this proof, we will first show that the first and second inequalities constitute a sufficient and necessary condition for π being a NE. Recall that a bi-policy π is a minimax equilibria for a two-player zero-sum game if and only if (Lin, Beling, and Cogill, 2017)

$$\begin{aligned}
\left[Q^\pi(s)\right]^T \pi_1(s) \geq V^\pi(s) 1_M \\
Q^\pi(s) \pi_2(s) \leq V^\pi(s) 1_M,
\end{aligned} \tag{5.32}$$

Similarly, π is a NE if and only if

$$\begin{aligned}
\left[Q_2^\pi(s)\right]^T \pi_1(s) \leq V_2^\pi(s) 1_M \\
Q_1^\pi(s) \pi_2(s) \leq V_1^\pi(s) 1_M.
\end{aligned} \tag{5.33}$$

Combining (5.14) and (5.33) leads to

$$\begin{aligned}
B_{\pi|a_2} \vec{Q}_2^\pi \leq B_\pi \vec{Q}_2^\pi, \forall a_2 \in \mathcal{A}_2 \\
B_{\pi|a_1} \vec{Q}_1^\pi \leq B_\pi \vec{Q}_1^\pi, \forall a_1 \in \mathcal{A}_1,
\end{aligned} \tag{5.34}$$

Substituting (5.12) into (5.34) and rearrange the two sides of the inequalities yields

$$\begin{aligned}\left(B_{\pi|a_1} - B_\pi\right) D_\pi r_1 \leq 0, \forall a_1 \in \mathcal{A}_1 \\ \left(B_{\pi|a_2} - B_\pi\right) D_\pi r_2 \leq 0, \forall a_2 \in \mathcal{A}_2,\end{aligned} \tag{5.35}$$

We now turn to the additional feature that an advE has. Recall (5.19), it is easy to derive that an advE for a two-player general-sum game if and only if, in addition to (5.33)

$$\begin{aligned}\left[Q_1^\pi(s)\right]^T \pi_1(s) \geq V_1^\pi(s) 1_M \\ Q_2^\pi(s) \pi_2(s) \geq V_2^\pi(s) 1_M.\end{aligned} \tag{5.36}$$

Following similar steps as we derive (5.34)-(5.36) is eventually reduced to

$$\begin{aligned}\left(B_{\pi|a_1} - B_\pi\right) D_\pi r_2 \geq 0, \forall a_1 \in \mathcal{A}_1 \\ \left(B_{\pi|a_2} - B_\pi\right) D_\pi r_1 \geq 0, \forall a_2 \in \mathcal{A}_2.\end{aligned} \tag{5.37}$$

□

Since it has been proved that in a one-stage game, if an advE exists, it must be unique (Littman, 2001), an advE for a stochastic game, if exists, is also unique. Therefore, we can still use Bayesian approach to solve advE-MIRL problems. The prior (5.27) is also valid here. But the likelihood would be

$$p(\pi|r_1, r_2) = \begin{cases} 1, & \text{if } \pi \text{ is an AdvE for } r_1, r_2 \\ 0, & \text{otherwise.} \end{cases} \tag{5.38}$$

And the optimization problem for advE-MIRL is

$$\begin{aligned} \text{minimize:} \quad & \frac{1}{2}\sum_i \left(r_i - \mu_{r_i}\right)^T \Sigma_{r_i}^{-1} \left(r_i - \mu_{r_i}\right) \\ \text{subject to:} \quad & \left(B_{\pi|a_1} - B_\pi\right) D_\pi r_1 \leq 0, \forall a_1 \in \mathcal{A}_1 \\ & \left(B_{\pi|a_2} - B_\pi\right) D_\pi r_2 \leq 0, \forall a_2 \in \mathcal{A}_2 \\ & \left(B_{\pi|a_1} - B_\pi\right) D_\pi r_2 \geq 0, \forall a_1 \in \mathcal{A}_1 \\ & \left(B_{\pi|a_2} - B_\pi\right) D_\pi r_1 \geq 0, \forall a_2 \in \mathcal{A}_2. \end{aligned} \tag{5.39}$$

In fact, there is a direct link between the minimax equilibrium of a competitive zero-sum game and an advE for a special zero-sum case, as the following proposition,

Proposition 5.4. *The minimax equilibrium of a single competitive zero-sum game is an advE, and vice versa.*

Proof. Let $r_1 = r = -r_2$, eqs. (5.31) reduce to

$$\begin{aligned} & \left(B_{\pi|a_1} - B_\pi\right) D_\pi r \leq 0, \forall a_1 \in \mathcal{A}_1 \\ & \left(B_{\pi|a_2} - B_\pi\right) D_\pi r \geq 0, \forall a_2 \in \mathcal{A}_2, \end{aligned} \tag{5.40}$$

which is exactly eqs. (23), the sufficient and necessary condition for π being a minimax equilibrium for a zero-sum game, in (Lin, Beling, and Cogill, 2017). □

From Theorem 5.4 we can see that advE is a more general concept for general-sum games whereas the minimax equilibrium corresponds specifically for zero-sum games.

5.4.4 cooE-MIRL

The main result characterizing the set of solutions to a two-player CooE-MIRL problem is the following:

Theorem 5.5. *Given a two-player stochastic game* $\{\mathcal{S}, \mathcal{A}_i, r_i, P, \gamma\}$*, the observed bi-policy* π *is an CooE if and only if*

$$\begin{aligned}
&\left(B_{\pi|a_1} - B_\pi\right) D_\pi r_1 \leq 0, \forall a_1 \in \mathcal{A}_1 \\
&\left(B_{\pi|a_2} - B_\pi\right) D_\pi r_2 \leq 0, \forall a_2 \in \mathcal{A}_2 \\
&\left(B_\pi - B_a\right) D_\pi r_1 \geq 0, \forall a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \\
&\left(B_\pi - B_a\right) D_\pi r_2 \geq 0, \forall a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2.
\end{aligned} \tag{5.41}$$

In (5.41), the first two inequalities, which guarantee π is a NE, has been proved in Section 5.4.3. The latter two inequalities warrant the unique property of CooE, the proof of which is sketched below.

Proof. According to the definition of CooE, π is a CooE if and only if, for any state s and pure bi-strategy $a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$,

$$\begin{aligned}
&\pi(s) \in \arg\max_{a \in \mathcal{A}} Q_i^\pi(s, a) \\
\Leftrightarrow &Q_i^\pi(s, \pi(s)) \geq Q_i^\pi(s, a) \\
\Leftrightarrow &r_i(s, \pi(s)) + \gamma P_{s,\pi(s)} V_i^\pi \geq r_i(s, a) + \gamma P_{s,a} V_i^\pi \\
\Leftrightarrow &B_\pi r_i + \gamma B_\pi P \left(I - \gamma G_\pi\right)^{-1} B_\pi r_i \\
&\geq B_a r_i + \gamma B_a P \left(I - \gamma G_\pi\right)^{-1} B_\pi r_i \\
\Leftrightarrow &\left(B_\pi - B_a\right) \left(I + \gamma P \left(I - \gamma G_\pi\right)^{-1} B_\pi\right) r_i \geq 0 \\
\Leftrightarrow &\left(B_\pi - B_a\right) D_\pi r_i \geq 0
\end{aligned} \tag{5.42}$$

□

We can also develop a similar optimization problem for cooE-MIRL. It is easy to show that an cooE for a stochastic game, if exists, is unique, for the reason for advE. As a result, the Bayesian approach is also valid here, with

the same prior (5.27) but a different likelihood as follows

$$p\left(\pi|r_1,r_2\right)=\begin{cases}1, & \text{if } \pi \text{ is an CooE for } r_1,r_2\\ 0, & \text{otherwise.}\end{cases} \tag{5.43}$$

Hence the optimization problem for cooE-MIRL is

$$\begin{aligned}\text{minimize:}\quad & \frac{1}{2}\sum_i\left(r_i-\mu_{r_i}\right)^T\Sigma_{r_i}^{-1}\left(r_i-\mu_{r_i}\right)\\ \text{subject to:}\quad & \left(B_{\pi|a_1}-B_\pi\right)D_\pi r_1\leq 0,\forall a_1\in\mathcal{A}_1\\ & \left(B_{\pi|a_2}-B_\pi\right)D_\pi r_2\leq 0,\forall a_2\in\mathcal{A}_2\\ & \left(B_\pi-B_a\right)D_\pi r_1\geq 0,\forall a\in\mathcal{A}=\mathcal{A}_1\times\mathcal{A}_2\\ & \left(B_\pi-B_a\right)D_\pi r_2\geq 0,\forall a\in\mathcal{A}=\mathcal{A}_1\times\mathcal{A}_2.\end{aligned} \tag{5.44}$$

5.4.5 uCE-MIRL

The result which characterizes the set of solutions to a two-player CE-MIRL problem is as follows:

Theorem 5.6. *Given a two-player stochastic game $\{\mathcal{S},\mathcal{A}_i,r_i,P,\gamma\}$, the observed bi-policy π is a CE if and only if*

$$\vec{\pi}^T H\left(s,a_i\right)^T\left[H\left(s,a_i\right)-H\left(s,\breve{a}_i\right)\right]D_\pi r_i\geq 0, i=1,2,\forall a_i\in\mathcal{A}_i,\breve{a}_i\in\mathcal{A}_i\setminus a_i, \tag{5.45}$$

where $\vec{\pi}$ is restructured from π to be a column vector of length NM^2, and $H(s,a_i)$ is a linear transformation operator as described in the proof below.

Proof. By definition of CE, for a two-player general-sum stochastic game $\mathcal{G}$, a bi-policy π is a CE if and only if

$$\begin{aligned}
&\sum_{a_2} \pi\left(a_1, a_2|s\right) Q_1^{\pi}\left(s, a_1, a_2\right) \geq \sum_{a_2} \pi\left(a_1, a_2|s\right) Q_1^{\pi}\left(s, \check{a}_1, a_2\right), \forall a_1 \in \mathcal{A}_1, \check{a}_1 \in \mathcal{A}_1 \setminus a_1 \\
&\sum_{a_1} \pi\left(a_1, a_2|s\right) Q_2^{\pi}\left(s, a_1, a_2\right) \geq \sum_{a_1} \pi\left(a_1, a_2|s\right) Q_2^{\pi}\left(s, a_1, \check{a}_2\right), \forall a_2 \in \mathcal{A}_2, \check{a}_2 \in \mathcal{A}_2 \setminus a_2
\end{aligned} \tag{5.46}$$

for all $s \in \mathcal{S}$. Rearranging (5.46) yields

$$\begin{aligned}
\pi\left(a_1, : |s\right)\left(\left[Q_1^{\pi}\left(s, a_1, :\right)\right]^T - \left[Q_1^{\pi}\left(s, \check{a}_1, :\right)\right]^T\right) \geq 0 \\
\left[\pi\left(:, a_2|s\right)\right]^T\left(Q_2^{\pi}\left(s, :, a_2\right) - Q_2^{\pi}\left(s, :, \check{a}_2\right)\right) \geq 0,
\end{aligned} \tag{5.47}$$

where $\pi\left(a_1, : |s\right)$ is a row vectors of $1 \times M$, spanning over all $a_2 \in \mathcal{A}_2$, and $\pi\left(:, a_2|s\right)$ is a column vectors of $M \times 1$, spanning over all $a_1 \in \mathcal{A}_1$. Recall

$$Q_i^{\pi}\left(s, a\right) = R_i\left(s, a\right) + \gamma \sum_{s'} p\left(s'|s, a\right) V_i^{\pi}\left(s'\right). \tag{5.48}$$

So

$$\begin{aligned}
\left[Q_1^{\pi}\left(s, a_1, :\right)\right]^T = \left[R_1\left(s, a_1, :\right)\right]^T + \gamma p\left(: |s, a_1, :\right) V_1^{\pi} \\
Q_2^{\pi}\left(s, :, a_2\right) = R_2\left(s, :, a_2\right) + \gamma p\left(: |s, :, a_2\right) V_2^{\pi}
\end{aligned} \tag{5.49}$$

Substituting (5.49) into (5.47) leads to

$$\begin{aligned}
&\pi\left(a_1, : |s\right)\left\{\left[R_1\left(s, a_1, :\right)\right]^T - \left[r_1\left(s, \check{a}_1, :\right)\right]^T + \gamma\left[p\left(: |s, a_1, :\right) - p\left(: |s, \check{a}_1, :\right)\right] V_1^{\pi}\right\} \geq 0 \\
&\left[\pi\left(:, a_2|s\right)\right]^T\left\{R_2\left(s, :, a_2\right) - r_2\left(s, :, \check{a}_2\right) + \gamma\left[p\left(: |s, :, a_2\right) - p\left(: |s, :, \check{a}_2\right)\right] V_2^{\pi}\right\} \geq 0.
\end{aligned} \tag{5.50}$$

The above inequality can be further simplified. First, let $\left[R_1\left(s, a_1, :\right)\right]^T = H\left(s, a_1\right) r_1$ and $R_2\left(s, :, a_2\right) = H\left(s, a_2\right) r_2$, where $H(s, a_i)$ is a sparse $M \times NM^2$ matrix. It is also easy to see $p\left(: |s, a_1, :\right) = H\left(s, a_1\right) P$ and $p\left(: |s, :, a_2\right) = H\left(s, a_2\right) P$. In addition, we can also have $\pi\left(a_1, : |s\right) = \left[H\left(s, a_1\right) \vec{\pi}\right]^T = \vec{\pi}^T H\left(s, a_1\right)^T$, and $\pi\left(:, a_2|s\right) = H\left(s, a_2\right) \vec{\pi}$. Substituting (5.10) into (5.50) and rearranging it,

we can get

$$\vec{\pi}^T H(s,a_i)^T \left[H(s,a_i) - H(s,\check{a}_i)\right] \left(I + \gamma P \left(I - \gamma G_\pi\right)^{-1} B_\pi\right) r_i \geq 0, i = 1,2 \tag{5.51}$$

Recall

$$D_\pi = I + \gamma P \left(I - \gamma G_\pi\right)^{-1} B_\pi, \tag{5.52}$$

we can express (5.51) compactly as

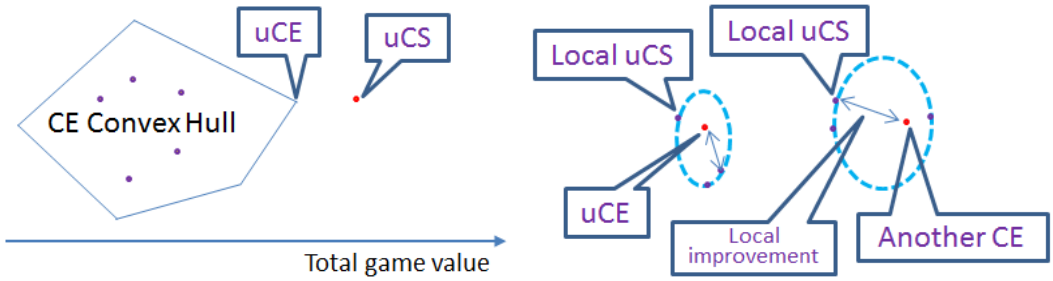
$$\vec{\pi}^T H(s,a_i)^T \left[H(s,a_i) - H(s,\check{a}_i)\right] D_\pi r_i \geq 0, i = 1,2, \forall a_i \in \mathcal{A}_i, \check{a}_i \in \mathcal{A}_i \setminus a_i, \tag{5.53}$$

□

Clearly, any sensible point that is consistent with (5.53) constitutes a CE for the stochastic game. Many points in the convex hull of CE, however, are less "meaningful" because only the uCE is of interest. Hence we desire to find some way to choose between solutions satisfying (5.53). A first idea is to maximize $\sum_s V^\pi(s)$. That is not enough though, because reaching a uCE is in practice difficult. Instead, arriving at a uCS is much easier. This fact gives us another idea. Before going into details, we need to introduce a new concept, namely, *local uCS*,

Definition 5.7. *A local uCS, corresponding to a bi-policy π and starting state s, in a two-player general-sum stochastic game, is such a bi-policy that the two players act fully cooperatively at current state s but employ π afterwards.*

It is obvious that for a two-player general sum stochastic game, among all its CEs, the uCE is "closest" to its uCS in terms of the total game value, shown in Figure 5.1 (A). In a uCE-MIRL problem, however, all CEs except uCE are unobservable. Therefore, we need to find a way to infer a set of $r_1 \& r_2$ such that the observed π is most likely the uCE of the game.



(A) (B)

FIGURE 5.1: (A) describes the relationship between uCE, uCS and other CEs. (B) explains local uCS.

By definition, a local uCS "improves" $V^{\pi}(s)$ by employing a uCS strategy only at current state s, resulting in a *local improvement* (see Figure 5.1 (B)). Adding up all those local improvements over all states gives us a way of measuring how "close" the bi-policy π is to a uCS, in terms of the total game value. Since a uCE is "closer" to a uCS than any other CE, the less its total local improvement is, the more likely a CE π is a uCE. Thus, given a CE π, a desired pair of $r_1 \& r_2$ satisfies

$$\begin{aligned} \text{minimize:} \quad & \sum_{s} y(s) - V^{\pi}(s) \\ \text{subject to:} \quad & Q_1^{\pi}(s,a) + Q_2^{\pi}(s,a) \leq y(s) \\ & V^{\pi}(s) \leq y(s) \end{aligned}$$

for all $a \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$. Putting all the above together, we propose the following linear programming problem to find the desired $r_1 \& r_2$,

$$\begin{aligned} \text{maximize:} \quad & \sum_{s} V^{\pi}(s) - \lambda \left(y(s) - V^{\pi}(s) \right) \\ \text{subject to:} \quad & Q_1^{\pi}(s,a) + Q_2^{\pi}(s,a) \leq y(s) \\ & V^{\pi}(s) \leq y(s) \\ & \text{Constraint (5.53)} \end{aligned} \tag{5.54}$$

where λ is a regularized coefficient. Expressing V_i^{π} and $Q_i^{\pi}(s,a)$ as functions

of r_i and reformulating those inequalities more compactly in matrix notation leads to

$$\begin{aligned}
\text{maximize:} \quad & \mathbf{1}_{1\times N} \times \left[(1+\lambda)\left(I-\gamma G_\pi\right)^{-1} B_\pi \left(r_1+r_2\right) - \lambda y\right] \\
& + \text{other problem-specific regularized terms} \\
\text{subject to:} \quad & \pi^T H\left(s,a_i\right)^T \left[H\left(s,a_i\right) - H\left(s,\check{a}_i\right)\right] D_\pi r_i \geq 0, i=1,2, \forall a_i \in \mathcal{A}_i, \check{a}_i \in \mathcal{A}_i \setminus a_i \\
& D_\pi \left(r_1+r_2\right) \leq y \cdot \mathbf{1}_{M\times M} \\
& \left(I-\gamma G_\pi\right)^{-1} B_\pi \left(r_1+r_2\right) \leq y.
\end{aligned} \tag{5.55}$$

We now discuss the “other problem-specific regularized terms" shown in the above problem. One challenging issue for MIRL is that there often exists many solutions equally sensible so that it is more likely than IRL to recover rewards which are far from actual ones. For example, in (Lin, Beling, and Cogill, 2017) the authors emphasize the importance of the structure of rewards. Therefore, some prior knowledge or assumption of the game, as well as the structure of the unknown rewards, is very helpful. For example, it is often assumed that, all other things being equal, an unknown reward vector is sparse (Ng and Russell, 2000). One easy way to incorporate this assumption is to add this penalty term to the objective function to regularize non-sparsity. There might be other problem-specific knowledge/assumption available and taking advantage of it will help infer higher-quality rewards.

5.4.6 uNE-MIRL

Recall that the sufficient and necessary condition for an observed bi-policy π being a NE for a two-player general-sum stochastic game is given by

$$\begin{aligned}
\left(B_{\pi|a_1} - B_\pi\right) D_\pi r_1 \leq 0, \forall a_1 \in \mathcal{A}_1 \\
\left(B_{\pi|a_2} - B_\pi\right) D_\pi r_2 \leq 0, \forall a_2 \in \mathcal{A}_2
\end{aligned} \tag{5.56}$$

Since NE is a subset of CE, we can borrow the idea proposed in Section 5.4.5 and solve a uNE-MIRL problem by solving the following LP problem

$$\begin{aligned}
\text{maximize:} \quad & \mathbf{1}_{1\times N} \times \left[(1+\lambda)\left(I-\gamma G_\pi\right)^{-1} B_\pi\left(r_1+r_2\right)-\lambda y\right] \\
& + \text{other problem-specific regularized terms} \\
\text{subject to:} \quad & \left(B_{\pi|a_1}-B_\pi\right) D_\pi r_1 \leq 0, \forall a_1 \in \mathcal{A}_1 \\
& \left(B_{\pi|a_2}-B_\pi\right) D_\pi r_2 \leq 0, \forall a_2 \in \mathcal{A}_2 \\
& D_\pi\left(r_1+r_2\right) \leq y \cdot \mathbf{1}_{M\times M} \\
& \left(I-\gamma G_\pi\right)^{-1} B_\pi\left(r_1+r_2\right) \leq y.
\end{aligned} \tag{5.57}$$

5.5 Numerical Examples I: GridWorld

This section describes the behaviour of our algorithms (except advE-MIRL) using two grid games (GGs), shown in Figure 5.2, namely GG1 for the left and GG2 for the right. These games have been used extensively in many theory-oriented MRL works (Hu and Wellman, 1998; Littman, 2001; Greenwald and Hall, 2003). In both GGs, there are two agents, A and B, and two goals (aka. homes). The two agents act simultaneously and can move only one step in any of the four compass directions. When adjacent to a wall, choosing a direction into a wall results in a no-op, where the agent remains in the current position. If both agents attempt to move into the same cell, a collision occurs and they are pushed back to their original positions immediately, except for cells in the bottom row. Either agent will be rewarded once reaching its goal under some condition. However, since the reward is discounted with time, the earlier to reach the goal, the better. GG1 and GG2 are similar in basic game rules but different in board setup in two aspects. First, in GG1, the two players' goals are separate while their goals coincide in GG2. Second, in GG2, there are two barriers and if any agent attempts to move downward through the barrier from the top, then with $1/2$ probability

this move fails and results in a no-op.

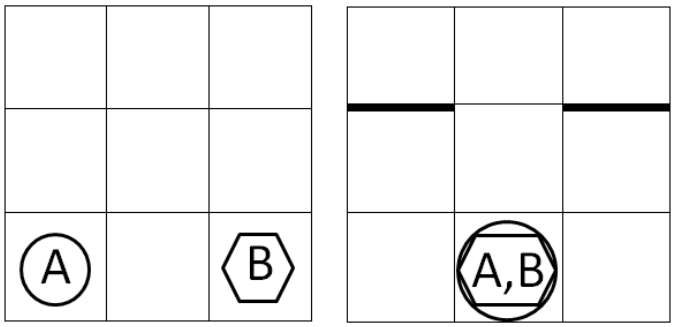


FIGURE 5.2: Grid games. The circle indicates A's goal and the hexagon indicates B's goal.

We let agents A and B play the *go-back-home* games together according to either uCS, uCE, uNE or cooE. Our task is to recover their rewards given the equilibrium, the bi-policy, and the state transition dynamics. The *basic* rewarding rule is: either player receives reward 1 (discounted with time) once reaching home and the game stops immediately, and 0 otherwise. When employing cooE, however, neither player receives reward unless they reach home *simultaneously*.

Our experiments are conducted as follows. First, we apply the cooE-MRL algorithm by Hu and Wellman (Hu and Wellman, 2003), and the uCE-MRL algorithm proposed by Green and Hall (Greenwald and Hall, 2003) to obtain cooE and uCE bi-policies, respectively. Then we develop similar Q-learning based iterative algorithms for uCS-MRL and uNE-MRL. The general procedure, namely multi-Q-learning algorithm, is the same for all these four MRL algorithms and described in 3. It is worth emphasizing that the multi-Q-learning algorithm can be applied to many variants of Q-learning problems as long as the equilibria exists and is unique (Hu and Wellman, 1998; Littman, 2001; Greenwald and Hall, 2003). It is easy to show that uCS, uCE, uNE and cooE all meet this requirement.

The second step is to apply our uCS-MIRL, cooE-MIRL uCE-MIRL and uNE-MIRL algorithms accordingly, incorporating our basic knowledge and

some reasonable assumptions into our Gaussian priors for uCS and cooE. For example, one assumption is that both players' reward vectors are sparse, only depending on reaching home or not. In addition, one agent's position might have a small affect on the other agent's reward or possibly no affect.

For each experiment, we compare recovered rewards of both players r_A^{rec} and r_B^{rec}, with the true values r_A and r_B numerically. We use a *normalized root mean squared error* (NRMSE) metric, where we first normalize a recovered reward vector r^{rec} on $[0, 1]$, as follows:

$$r^{\text{nrec}} = \frac{r^{\text{rec}} - \min(r^{\text{rec}})}{\max(r^{\text{rec}}) - \min(r^{\text{rec}})},$$

and then compute

$$\text{NRMSE} = \sqrt{\frac{\left\| r_A^{\text{nrec}} - r_A \right\|_2}{\dim(r_A)}} + \sqrt{\frac{\left\| r_B^{\text{nrec}} - r_B \right\|_2}{\dim(r_B)}}$$

In addition, in order to compare our MIRL algorithm with IRL. We use IRL algorithms to solve the uCS-, cooE- and uNE-MIRL problems, namely uCS-, cooE- and uNE-IRL respectively. Specifically, we focus on B, and try to infer its reward. Obviously, inferred IRL reward is a function of the state and B's own action. The IRL algorithm we use is BIRL, proposed in (Qiao and Beling, 2011). Note that the reward vector recovered from IRL can be extended to a MIRL reward vector by letting $R(s, a_1, a_2) = R(s, a_2)$ for all $a_1 \in \mathcal{A}_1$. Let a third player $\hat{B}$ learn this reward and figure out its own IRL policy $\pi_{\hat{B}}$. Finally, for uCS, let $\hat{B}$ play with A, with $\hat{B}$ employing $\pi_{\hat{B}}$ and A employing the corresponding π_A, compute their total game value over all states and compare with true total game values. Note that we cannot model a uCE-MIRL problem as an IRL problem because uCE permits dependencies among agents' policies (there is often a trusted mediator sending private information to game players).

Numerical results are shown in Table 5.1 and Table 5.2. Some plots are also presented for readers to have a better insight of our MIRL algorithms recovered results. In Figure 5.3, the top 2 subplots are for grid game #1 and bottom 2 are for grid game #2. The two subplots in (A) and (C) describe the true reward (in red circles) and uCS-MIRL recovered reward (in blue stars). The two subplots in (B) and (D) demonstrate the recovered reward in terms of total game value compared to true reward. In each of these subplots, red circles, blue stars and green squares represent the total game value generated from true reward, uCS-MIRL recovered reward and IRL recovered reward, respectively. Figure 5.4 describes the cooE-MIRL results. The left two plots are for grid game #1 and right two are for grid game #2. In each set, the top subplot shows player A's recovered reward (in blue stars) and the bottom one shows that of player B. We can see that both uCS-MIRL and cooE-MIRL recovered rewards are not numerically close the true values, with an obvious scale difference. That's why for a fair comparison purpose, we normalize them on $[0, 1]$ first.

From all the above results, we can easily conclude that our MIRL algorithms generate satisfactory results and performs much better than IRL algorithms for all the four problems.

Algorithm 3 General Multi-Q-learning algorithm

Require: f: uCS, cooE, uCE or uNE; α: learning rate

procedure MULTI-Q(f, T, r_1, r_2, α)
  Initialize: $s, a, Q_1, Q_2, t = 0$
  while $t < T$ **do**
    agents choose bi-strategy a in state s
    observe rewards and next state s'

    for $i = 1 \rightarrow N$ **do**
      $V_i(s') = f_i(Q_1(s'), Q_2(s'))$
      $Q_i(s,a) = (1-\alpha) Q_i(s,a) + \alpha[(1-\gamma) r_i(s,a) + \gamma V_i(s')]$

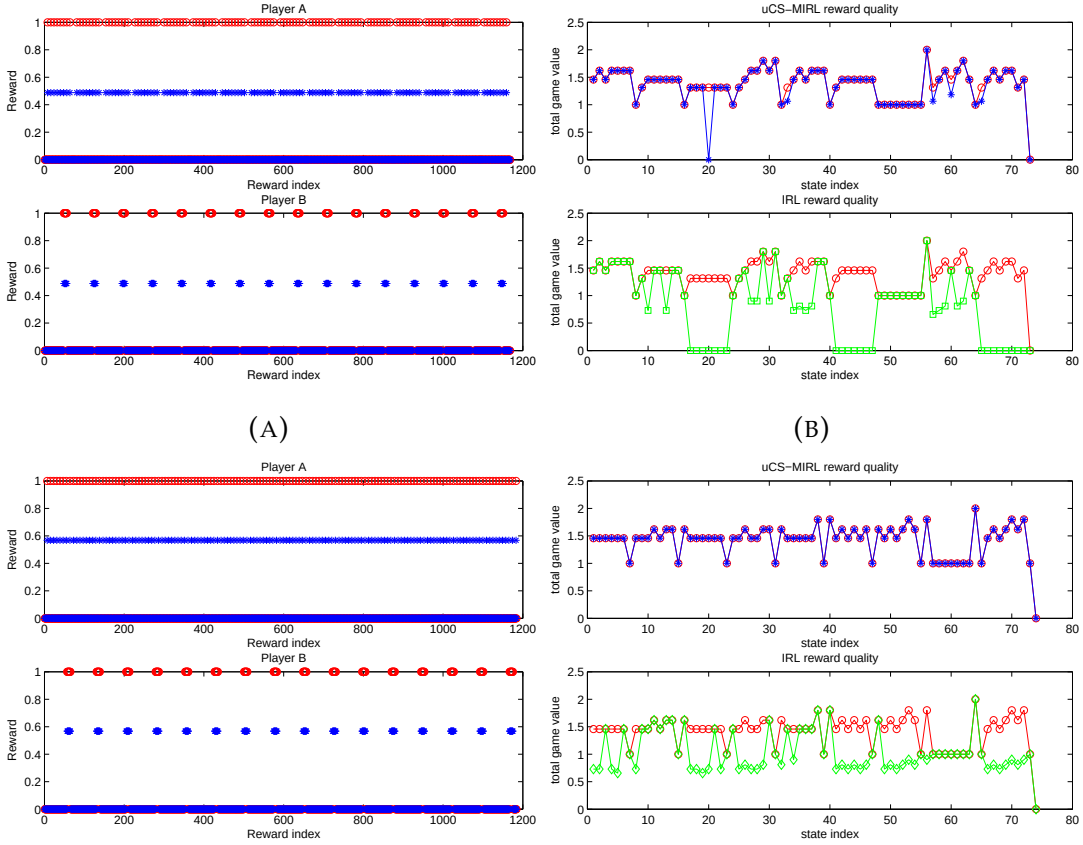
    agents choose bi-strategy $\vec{a'}$
    $s = s', a = a'$
    decay α
    $t = t + 1$

	Grid Game #1	Grid Game #2
uCS-MIRL	1.50×10^{-3}	2.27×10^{-4}
uCS-IRL	0.122	0.121
cooE-MIRL	0.026	0.026
cooE-IRL	0.409	0.319
uCE-MIRL	1.30×10^{-3}	1.39×10^{-10}
uCE-IRL	0.287	0.311
uNE-MIRL	0	0
uNE-IRL	0.271	0.283

TABLE 5.1: NAED results for reward values comparison

	Grid Game #1	Grid Game #2
uCS-MIRL	0.099	2.50×10^{-4}
uCS-IRL	0.223	0.179

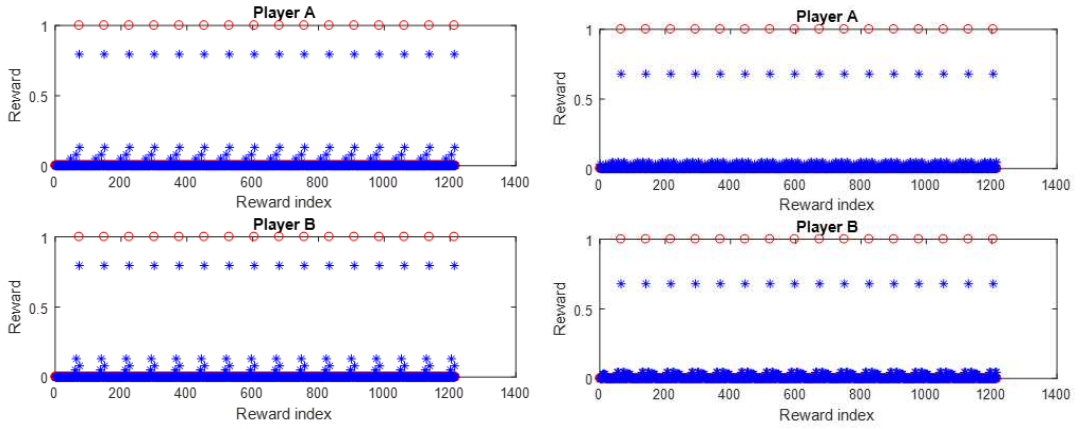
TABLE 5.2: total game value comparison for uCS



(A) (B)

(C) (D)

FIGURE 5.3: The uCS-MIRL results



(A) (B)

FIGURE 5.4: cooE-MIRL experiment result

5.6 Numerical Examples II: Abstract Soccer Game

This section dedicates to demonstrate our advE-MIRL algorithm. Two-player soccer games in many versions are popular among MRL researchers for algorithm demonstration & comparison purposes (Littman, 1994; Greenwald and Hall, 2003; Lin, Beling, and Cogill, 2017). In (Lin, Beling, and Cogill, 2017), where the most complicated version is created, authors propose a zero-sum MIRL algorithm and demonstrate its good performance. However, their algorithm works on the basis of a *zero-sum* assumption, which is too strong. In this section, we relax this *zero-sum* assumption and just assume that the two players are foes, which enables us to rely on a weaker assumption that they employ an advE.

The soccer game (see Figure 5.5) is depicted as follows. Players A and B compete with each other, aiming to score by either bringing or kicking the ball (represented by a circle) into their opponents' goals (A's goal are 6 and 11, and B's goal are 10 and 15). Both players can move simultaneously either in four compass directions, ending in a neighbouring cell or stay unmoved. A ball exchange may occur with some probability in case of a collision in the same cell. A *kick* action is also available to players. Each one has a perception of how likely she is making a scoring shot, or the *probability of a successful shot* (PSS), if kicking the ball at a given position. For simplicity, one's PSS is assumed not affected by its opponent's position. The position based PSS distribution is shown in Table 5.3.

	PSS = 0.7	PSS = 0.5	PSS = 0.3	PSS = 0.1	PSS = 0
A	1, 7, 12, 16	2, 8, 13, 17	3, 9, 14, 18	4, 10, 15, 19	5, 20
	PSS = 0.7	PSS = 0.5	PSS = 0.3	PSS = 0.1	PSS = 0
B	5, 9, 14, 20	4, 8, 13, 19	3, 7, 12, 18	2, 6, 11, 17	1, 16

TABLE 5.3: Original PSS distribution of each player

It is worth clarifying a confusion point: a player's PSS at a particular spot is her *perceived* likelihood of a scoring short, rather than the *actual* probability

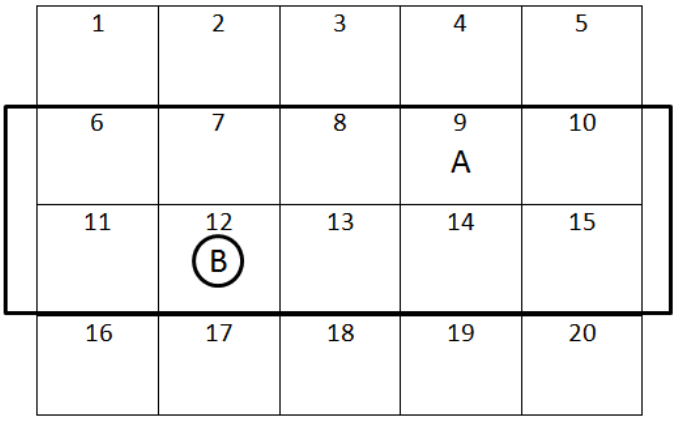


FIGURE 5.5: Soccer game: initial board

of a successful shot. So statistically calculating the "successful shot" rate from observation data does not help reflect the player's own belief of her shooting skills, which is, the player's reward.

With this setting, we let the two players play against each other, both employing a minimax equilibrium. In other words, this is a zero-sum game. But this information is *not* available to us. Instead, we are given: 1. the bi-policy of the two players over all states, and; 2. the state transition dynamics (including the ball exchange rate $\beta = 0.6$). In fact, this information can be statistically calculated or estimated with sufficient observations. We simply skip this data pre-processing stage as it is not the emphasis of this chapter. We then assume that the two players follow an advE and try to infer their rewards on this basis.

5.6.1 Prior Specification

As indicated in Section 5.4.3, one of the key specifications of the advE-MIRL model is the prior, which encodes our beliefs of the unknown rewards. We use two Gaussian priors for the unknowns of A and B with three types of means and two types of covariance matrices, as follows:

- *Weak Mean*: for A, assign 0.5 point in every state where A has possession of the ball and -0.5 point in every state where A loses possession of the

ball. Same setting for B.

- *Median Mean*: for A, assign 1 point whenever it has the ball and is in one of the corner squares 1, 6, 11 and 16, and -1 point to A whenever B has the ball and is in its hypothesized goal area 5, 10, 15 and 20. When A has the ball and takes a shot, it is assigned 0.5 wherever it is; when B has the ball and takes a shot, A is assigned -0.5 wherever B is. Otherwise, no points will be assigned. Same setting for B.

- *Strong Mean*: Similar to Median Mean except for a more accurate perception of where the goals are for A and B.

- *Weak Covariance Matrix*: an identity matrix for both A and B, which implies no knowledge or guess about the relationship between any two reward values is available.

- *Strong Covariance Matrix*: a more complex covariance matrix constructed from our following hypothesis of the reward structure, same for both A and B.

    1. when one player has the ball and takes a shot, its PSS depends only on its' current position in the field, and;
    2. at any state, one's reward for any non-*shoot* action is one's own position dependent.

To make it clear, our prior specifications do *not* imply a zero-sum relationship between A and B's reward. As the singularity issue may occur when using strong covariance matrices, we add a small numerical perturbation to the diagonal.

5.6.2 Monte Carlo Simulation using Recovered Rewards

By solving an advE-MIRL problem (5.39), we recover A and B's reward vectors, over all states and all actions. For both of them, there are 6 advE-MIRL reward vectors recovered corresponding to the 6 pairs of means and covariance matrices. Since we have seen that evaluating the quality of recovered reward by simply measuring its numerical difference from true value may lead to misleading conclusion, we adopt a Monte Carlo simulation method implied in Section 5.5, by taking the following two steps:

1. Create agents
    - *C*, which uses advE-MIRL reward
    - *D*, which uses true reward
    - *E*, which uses zero-sum MIRL reward
    - *F*, which uses dMIRL reward
    - *G*, which uses BIRL reward
2. Design competitive games
    - C against D;
    - C against E;
    - C against F;
    - C against G;

Note that agent E, F and G use rewards recovered from three conventional MIRL approaches covered in Section 5.3. Here we let C plays the role of A and others take the place of B (due to symmetry, two parties can switch roles as well). All those games are simulated in three different environmental settings, where the the ball exchange rates β are 0, 0.4 and 1. 5000 round games are simulated per case.

The simulation results are presented in Table 5.4–Table 5.7, where *WM*, *MM*, *SM*, *WC* and *SC* stand for *weak mean*, *median mean*, *strong mean*, *weak covariance matrix* and *strong covariance matrix*, respectively. To interpret the result, take the 2nd row of Table 5.5 as an example: C uses WM and SC as prior and recovers A's advE-MIRL reward, while E also use the same prior and learns a zero-sum MIRL reward vector of B. They come up with their own minimax policies according to their learned rewards and environmental settings and play against each other. 32.28/36.40 means C beats E with probability 32.28%, loses with probability 36.40%, and end in a tie with probability 31.32%, when the ball exchange rate is 1. Note that 0/0 shown in these tables means the both parties learn very bad rewards such that no one is able to score a single point even if its opponent is also poorly skilled.

advE-MIRL Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/32.44	0/58.00	0/49.98
WM & SC	20.40/25.46	20.50/38.24	42.88/50.16
MM & WC	4.60/30.12	9.36/44.00	10.44/49.88
MM & SC	24.86/24.94	25.10/24.80	49.97/50.02
SM & WC	14.90/30.52	6.80/42.50	15.42/50.08
SM & SC	25.26/24.80	25.00/24.80	50.14/49.86

TABLE 5.4: C vs D

advE-MIRL Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/2.40	0/0	0/0
WM & SC	22.76/28.94	32.28/36.40	43.14/50.14
MM & WC	0/0	9.20/5.60	4.12/16.86
MM & SC	24.86/25.12	25.04/24.96	49.54/50.44
SM & WC	11.24/10.60	8.80/9.18	16.10/24.46
SM & SC	25.28/25.06	24.94/25.12	50.13/49.86

TABLE 5.5: C vs E

From all the above results, as a whole, we can see that our advE-MIRL algorithm

- performs, if not better, comparatively with zero-sum MIRL algorithm, though the latter requires a stronger assumption.

advE-MIRL Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/0	0/0	0/0
WM & SC	27.10/0	25.42/0	50.04/0
MM & WC	6.04/0	8.64/0	18.02/0
MM & SC	25.28/0	26.06/0	49.86/0
SM & WC	13.98/0	9.00/0	49.26/0
SM & SC	24.90/0	26.08/0	49.90/0

TABLE 5.6: C vs F

advE-MIRL Rewards	W/L% ($\beta = 0.4$)	W/L% ($\beta = 1$)	W/L% ($\beta = 0$)
WM & WC	0/0	0/0	0/0
WM & SC	25.10/0	24.84/0	50.12/0
MM & WC	5.52/0	8.76/0	16.20/0
MM & SC	28.50/10.12	25.12/12.00	49.26/20.46
SM & WC	14.20/0	8.64/0	44.25/0
SM & SC	25.80/0	25.28/0	50.12/0

TABLE 5.7: C vs G

- performs notably better than d-MIRL and BIRL algorithms, particularly when using a strong covariance in prior.

5.7 Conclusions

We present novel and computationally tractable algorithms to five special variants of MIRL problems, as well as demonstrations with several benchmark grid-world examples. advE-MIRL requires weaker assumptions whereas achieves similar performance compared to zero-sum MIRL, and works much better than d-MIRL and BIRL. uCS and cooE generate good results if scales can be tweaked. uCE and uNE perform remarkably well in two benchmark grid-world examples, not only qualitatively good but also numerically close to true values. There are three reasons why the results are so good. First, these two small GGs are well-defined in the sense that there is no chance of moving in another direction by accident once a certain direction is selected (no noise in action). Second, the bi-policy π we use is exactly the equilibrium

of interest because it is generated from a corresponding MRL-Q-learning algorithm. Third, we have incorporated strong prior information about the game, and a good solution can be achieved by tuning the regularized coefficients.

Chapter 6

Conclusions

This chapter summarizes the dissertation, concludes our research findings, state the significance of our work, discusses the limitations and outlines directions for future research.

In Chapter 4, we propose an MIRL algorithm for two-person zero-sum stochastic games. This problem is of particular intest because there are many real world applications where our algorithm could be potentially applicable. Another reason that we select this problem to investigate first is that it is relatively simple, in the sense that 1. there are only two agents involved and their relationship is “clear" - fully competitive, and; 2. since their reward is zero-sum, we only need to infer one agent’s reward. To address this problem, we first assume that the two agents have been following minimax equilibriums at every sequential game. Second, we characterize a set of feasible solutions in which each solution is consistent with the observations. And most importantly, the minimax equilibrium is unique. We set up a MAP estimation. Specifically, we select a Gaussian prior, with the mean of the unknown reward vector and its covariance matrix, and aim to maximize the posterior of the unknowns given policy. Luckily, the likelihood here is simply 1. One obvious advantage of our algorithm is that it is convex and thus tractable. We also discuss the scalability of our algorithm, particularly when the covariance matrix is large and non-sparse. In the experiment, we demonstrate that our zero-sum MIRL algorithm is able to generate high quality solution, that

is to say, the policy generated from our solution behaves as well as the true optimal policy, particular when a strong covariance matrix is selected. We also show that in case of environment changes, our algorithm works better than other three treatments: 1. infer new policies directly from given policy in the old environment; 2. model the problem as an IRL, in particular, focus on one agent and treat the other as part of the passive environment, and; 3. use the decentralized-MIRL method proposed in (Reddy et al., 2012).

In Chapter 5, we study a more difficult MIRL problem where two or more agents are involved and their rewards are not zero-sum. We clearly emphasize two challenges that general-sum MIRL problems have: 1. there could be many "relationships" between multiple agents, in other words, they not necessarily follow one particular type of equilibriums, and; 2. even if we know what type of equilibriums they employ, the equilibrium of that particular type may not be unique (actually, it is often not unique). Thus we focus on five particular equilibriums and propose corresponding MIRL algorithms. To use these five algorithms, it is required: 1. each agent's full policies over all states is given or can be estimated through observations; 2. the equilibrium they follow is known. For uCS/advE/cooE-MIRL, we first develop a characteristic set of solutions ensuring that the observed bi-policy is a corresponding strategy/equilibrium and then apply a Bayesian inverse learning method. For uCE-MIRL, we develop a linear programming problem, subject to constraints that define necessary and sufficient conditions for the observed policies to be CE. For the objective function, we propose novel heuristics to choose a solution that not only minimizes the total game value difference between the observed bi-policy and its *local* uCS, but also maximizes the scale of the solution. Similar ideas have been borrowed to tackle uNE-MIRL. Experiments have shown remarkable performance of each algorithm. As we do in Chapter 4, comparisons are conducted against a Bayesian IRL algorithm.

We propose in total six algorithms to six different MIRL problems, one for

zero-sum and the other five for five general-sum problems, and demonstrate them using various benchmark grid world experiments. We can come to the following conclusions: 1. our algorithms generally work well; 2. the more informative prior to select, the better our algorithms' performance; 3. For a Gaussian prior, selecting a reasonable covariance is more important than picking a good prior mean of the unknown rewards, and; 4. Monte-carlo simulation metric is another important metric to measure the quality of our results.

In addition to theoretical contributions Chapter 1, this thesis has clearly answered two fundamental questions regarding to MIRL:

- Why is MIRL problem worth investigation?
- For some MIRL problem, is it necessary to come up with a MIRL algorithm? Or in other words, are current algorithms/treatments, such as policy learning and IRL, enough for solving a MIRL problem?

However, this work is by no means a complete treatment of MIRL problems, particularly in terms of real applications, as it has several technical limitations:

- Human beings are very unlikely to always employ a same type of equilibrium that we cover in this dissertation, in every sequential game. But in this dissertation, we do not take action noise into account.
- It is very difficult to obtain accurate estimates of state transition matrices, as well as policies, through limited observations of state-multi-action trajectories. But how to quantify the impact of the inaccuracy of these two inputs, or algorithm stability, is not yet considered.
- The assumption of knowing the type of employed equilibrium is not always valid. We thus need to come up with a equilibrium selection scheme based on their policies.

As a result, future MIRL research efforts can be put into how to break through the above limitations. In addition, we often see, in reality, the games in which two groups play against each other while within each group, members employ another cooperative or non-cooperative equilibria. An immediate idea comes to our mind is to set up a hierarchical MIRL problem and solve it in an iterative way. To the best of our knowledge, no such problems have been address. Our work can serve a starting point.

Bibliography

Abbeel, P., D. Dolgov, and A. Y. Ng (2008). "Apprenticeship learning for motion planning with application to parking lot navigation". In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'08*.

Abbeel, P. and A. Y. Ng (2004). "Apprenticeship learning via inverse reinforcement learning". In: *Proc. Intl. Conf. Mach. Learning (ICML'04)*, pp. 1–8.

Abbot, T., D. Kane, and P. Valiant (2004). "On Algorithms for Nash Equilibria". URL: http://web.mit.edu/tabbott/Public/final.pdf.

Abdallah, S. and V. Lesser (2008). "A Multiagent Reinforcement Learning Algorithm with Non-linear Dynamics". In: *J. Artif. Intell. Res.* 33, pp. 521–549.

Amadae, S. M. (2016). *Prisoners of Reason: Game Theory and Neoliberal Political Economy*. Cambridge University Press.

Aumann, R. (1974). "Subjectivity and Correlation in Randomized Strategies". In: *Journal of Mathematical Economics* 1, pp. 67–96.

Bajo, J., P. Mathieu, and M. J. Escalona (2017). "Multi-agent technologies in economics". In: *Intelligent Systems in Accounting, Finance and Management* 24 (2-3).

Baker, C. L., R. Saxe, and J. B. Tenenbaum (2009). "Action Understanding as Inverse Planning". In: *Cognition* 113.3, pp. 329–349.

Barr, N. (2012). *Economics of the Welfare State*. 5th ed. Oxford University Press.

Bazzan, A. L. C. (2009). "Opportunities for Multiagent Systems and Multiagent Reinforcement Learning in TrafïňĄc Control". In: *Autonom. Agents Multi-Agent Syst.* 18, pp. 342–375.

Bellman, R. E. (1957). *Dynamic Programming*. Princeton, NJ: Princeton University Press.

Borum, R. (2009). *Anticipating Your OpponentâĂŹs Action*. URL: http://combatsportpsychology.blogspot.com/2009/02/anticipating-your-opponents-action.html.

Casella, G. and R. L. Berger (2001). *Statistical Inference*. Duxbury Press.

Choi, J. and K. Kim (2009). "Inverse Reinforcement Learning in Partially Observable Environments". In: *Proceedings of the 21st International Joint Conference on Artifical Intelligence, IJCAI'09*. Pasadena, California, USA, pp. 1028–1033.

— (2011). "MAP Inference for Bayesian Inverse Reinforcement Learning". In: *Proc. Adv. Neural Info. Proc. Syst. (NIPS'01)*, pp. 1989–1997.

Daskalakis, C., P. W. Goldberg, and C. H. Papadimitriou (2009). "The Complexity of Computing a Nash Equilibrium". In: *SIAM Journal on Computing* 39.1, pp. 195–259.

Dimitrakakis, C. and C. A. Rothkopf (2011). "Bayesian Multitask Inverse Reinforcement Learning". In: *Proc. Euro. Workshops Reinforcement Learning (EWRL'11)*, pp. 273–284.

dInverno, M. et al. (2004). "The dMARS Architecture: A Specification of the Distributed Multi-Agent Reasoning System". In: *Journal of Autonomous Agents and Multi-Agent Systems*, pp. 5–53.

Duan, Y., B. Xia Cui, and X. Xu (2012). "A multi-agent reinforcement learning approach to robot soccer". In: *Artificial Intelligence Review* 38.3, pp. 193–211.

Earls, A. (2015). *From Germany to the World: Industry 4.0*. URL: https://www.smartindustry.com/blog/smart-industry-connect/from-germany-to-the-world-industry-4-0/.

Engel, Y., S. Mannor, and R. Meir (2005). "Reinforcement Learning with Gaussian Processes". In: *Proc. Intl. Conf. Mach. learning (ICML'05)*. Bonn, Germany, pp. 201–208. ISBN: 1-59593-180-5.

Ferguson, T. S. (2008). *Game Theory*. UCLA.

Filar, J. and K. Vrieze (1996). *Competitive Markov Decision Processes*. 1st. New York, NY: Springer-Verlag.

Fudenberg, D. and J. Tirole (1991). *Game Theory*. MIT Press.

Gabel, T. and M. Riedmiller (2007). "On a Successful Application of Multi-Agent Reinforcement Learning to Operations Research Benchmarks". In: *2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*.

Ghavamzadeh, M., S. Mahadevan, and R. Makar (2006). "Hierarchical Multi-agent Reinforcement Learning". In: *Autonom. Agents Multi-Agent Syst.* 13, pp. 197–229.

Giles, C. L. and KC Jim (2002). "Learning Communication for Multi-agent Systems". In: *Workshop on Radical Agent Concepts (WRAC)*.

Greene, T. (2017). *AI isnâĂŹt just learning to play video games, itâĂŹs helping us build them*. URL: https://thenextweb.com/artificial-intelligence/2017/10/11/video-games-are-about-to-get-real-ai-not-that-fake-cpu-opponent-crap/.

Greenwald, A. and K. Hall (2003). "Correlated Q-learning". In: *Proceedings of the 20th International Conference on Machine Learning, ICML'03*, pp. 242–249.

Hadfield-Menell, D. et al. (2016). "Cooperative Inverse Reinforcement Learning". In: *Proceedings of the 30th Neural Information Processing Systems (NIPS'16)*.

Hart, S. and D. Schmeidler (1989). "Existence of Correlated Equilibria". In: *Mathematics of Operations Research* 14.1, pp. 18–25.

Hodges, J. (2016). *Intimidation Tactics: Capture Your Opponents MIND!* URL: http://www.sportsmind.com.au/index.php/article/tennis/entry/intimidation-tactics-capture-your-opponents-mind.

Hu, J. and M. P. Wellman (1998). "Multiagent Reinforcement Learning: Theoretical Framework and an Algorithm". In: *Proc. Intl. Conf. on Mach. Learning (ICML'98)*, pp. 242–250.

— (2003). "Nash Q-Learning for General-Sum Stochastic Games". In: *The Journal of Machine Learning Research* 4, pp. 1039–1069.

Hurwicz, L. and S. Reiter (2006). *Designing Economic Mechanisms*. Cambridge University Press.

J. Albert M. Clickman, T. Swartz and R. Koning (2016). *Handbook of Statistical Methods and Analyses in Sports*. Chapman and Hall/CRC.

Kirchner, E. A. et al. (2016). "An Intelligent Man-Machine InterfaceâĂŤMulti-Robot Control Adapted for Task Engagement Based on Single-Trial Detectability of P300". In: *Frontiers in Human Neuoscience* 10 (291).

Kopacek, P. (1999). "Intelligent Manufacturing:Present State and Future Trends". In: *Journal of Intelligent and Robotic Systems* 26 (3–4), 217âĂŞ229.

Krishnamurthy, D. and E. Todorov (2010). "Inverse Optimal Control with Linearly-Solvable MDPs". In: *Proc. Intl. Conf. Mach. Learning (ICML'10)*, pp. 335–342.

Kuo, M. A. (2017). *China Manufacturing 2025: Impact on Europe*. URL: https://thediplomat.com/2017/09/china-manufacturing-2025-impact-on-europe/.

Lemke, C. E. and J. T. Howson (1964). "Equilibrium Points of Bimatrix Games". In: *SIAM Journal on Applied Mathematics* 12 (3), 413âĂŞ423.

Levine, S., Z. Popović, and V. Koltun (2011). "Nonlinear Inverse Reinforcement Learning with Gaussian Processes". In: *Proc. Adv. in Neural Info. Proc. (NIPS'11)*, pp. 19–27.

Lin, X., P. A. Beling, and R. Cogill (2017). "Multi-agent Inverse Reinforcement Learning for Two-Person Zero-sum Games". In: *IEEE Transactions on Computational Intelligence and AI in Games*. early access. DOI: 10.1109/TCIAIG.2017.2679115.

Littman, M. L. (1994). "Markov games as a framework for multi-agent reinforcement learning". In: *Proc. Intl. Conf. Mach. Learning (ICML'94)*, pp. 157–163.

— (2001). "Friend-or-Foe Q-learning in General-Sum Games". In: *Proceedings of the 18th International Conference on Machine Learning, ICML'01*, pp. 322–328.

Liu, J. and J. Wu (2002). "Multi-Agent Robotic Systems". In: *Industrial Robot: An International Journal* 29.6.

Lou, H. (2017). *AI in Video Games: Toward a More Intelligent Game*. URL: http://sitn.hms.harvard.edu/flash/2017/ai-video-games-toward-intelligent-game/.

Lynch, Gary S. (2009). *Single Point of Failure: The 10 Essential Laws of Supply Chain Risk Management*. Wiley.

Maddison, C. J. et al. (2015). "Move Evaluation in Go Using Deep Convolutional Neural Networks". In: URL: arXiv:1412.6564v2.

Maes, P. (1995). "Artificial Life Meets Entertainment: Lifelike Autonomous Agents". In: *Communications of the ACM* 38.11, pp. 108–114.

Marshall, A.W. and I. Olkin (1988). "Families of multivariate distributions". In: *Journal of the American Statistical Association* 83.403, pp. 834–841.

Maskin, E. and J. A. Tirole (2001). "Markov Perfect Equilibrium: I. Observable Actions". In: *Journal of Economic Theory* 100.2, pp. 191–219.

McGuigan, M. (2017). *Monitoring Training and Performance in Athletes*. Human Kinetics.

Michini, B. and J. P. How (2012). "Bayesian Nonparametric Inverse Reinforcement Learning". In: *Proc. Euro. Conf. Mach. Learning, Principles, Practice of Knowledge Discov. in Databases (ECML/PKDD'12)*. Vol. 2, pp. 148–163.

Millingto, I. (2009). *Artificial Intelligence for Games*. 2nd ed. CRC Press.

Mombaur, K., A. Truong, and J. Laumond (2010). "From human to humanoid locomotionâĂŤan inverse optimal control approach". In: *Autonomous Robots* 28 (3), 369âĂŞ–383.

Mozur, P. (2017). *GoogleâĂŹs AlphaGo Defeats Chinese Go Master in Win for AI*. The New York Times. URL: https://www.nytimes.com/2017/05/23/business/google-deepmind-alphago-go-champion-defeat.html.

Nash, J. (1951). "Non-Cooperative Games". In: *The Annals of Mathematics* 54.2, pp. 286–295.

Natarajan, S. et al. (2010). "Multi-Agent Inverse Reinforcement Learning". In: *Proc. Intl. Conf. Mach. Learning App. (ICMLA'10)*, pp. 395–400.

Neu, G. and C. Szepesvári (2007). "Apprenticeship learning using inverse reinforcement learning and gradient methods". In: *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence, UAI'07*, pp. 295–302.

Neumann, J. and O. Morgenstern (1944). *Theory of Games and Economic Behavior*. Princeton, NJ: Princeton University Press.

Ng, A. Y. and S. Russell (2000). "Algorithms for Inverse Reinforcement Learning". In: *Proc. Intl. Conf. Mach. Learning (ICML'00)*, pp. 663–670.

Ould-Khessal, N. (2005). "Design and implementation of a robot soccer team based on omni-directional wheels". In: *Proceedings of the 2nd Canadian Conference on Computer and Robot Vision*.

Owen, G. (1968). *Game Theory*. 1st. Philadelphia, PA: W. B. Saunders Company.

Ozdaglar, A. (2010). *Game Theory with Engineering Applications*. MIT.

Parka, S. and V. Sugumaran (2005). "Designing multi-agent systems: a framework and application". In: *Expert Systems with Applications* 28 (2), pp. 259–271.

Patek, S. D., P. A. Beling, and Y. Zhao (2007). "Natural Solutions for a Class of Symmetric Games". In: *AAAI Spring Symp. Game Theoretic Decision Theoretic Agents*, pp. 47–53.

Pitt, J. and A. Mamdani (2000). "Communication Protocols in Multi-agent Systems: A Development Method and Reference Architecture". In: *Issues in Agent Communication* 1916, pp. 160–177.

Pollock, J. L. (2006). *Thinking about Acting: Logical Foundations for Rational Decision Making*. 1st ed. Oxford University Press.

Qiao, Q. and P. A. Beling (2011). "Inverse Reinforcement Learning with Gaussian Process". In: *Proc. American Control Conf. (ACC'11)*, pp. 113 –118.

— (2013). "Recognition of agents based on observation of their sequential behavior". In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases, ECMLPKDD'13*, pp. 33–48.

Ramachandran, D. and E. Amir (2007). "Bayesian Inverse Reinforcement Learning". In: *Proc. Intl. Joint Conf. Artif. Intell. (IJCAI'07)*, pp. 2586–2591.

Rapoport, A. and A. M. Chammah (1966). "The Game of Chicken". In: *American Behavioral Scientist* 10 (3), pp. 10–28.

Ratliff, N., J. A. Bagnell, and M. Zinkevich (2006). "Maximum Margin Planning". In: *Proceedings of the 23rd International Conference on Machine Learning, ICML'06*, pp. 729–736.

Reddy, T. S. et al. (2012). "Inverse Reinforcement Learning for Decentralized Non-Cooperative Multiagent Systems". In: *Proc. IEEE Intl. Conf. Syst., Man, Cybern. (SMC'12)*.

Rudelson, M. and R. Vershynin (2014). "Invertibility of Random Matrices: Unitary and Orthogonal Perturbations". In: *J. American Math. Soci.* 27, pp. 293–338.

Runarsson, T. R. and S. M. Lucas (2014). "Preference Learning for Move Prediction and Evaluation Function Approximation in *Othello*". In: *IEEE Transactions on Computational Intelligence and AI in Games* 6.3, pp. 300–313.

Russell, S. (1998). "Learning agents for uncertain environments (Extended Abstract)". In: *Proc. Ann. Conf. on Comp. Learning Theory (COLT'98)*, pp. 101–103.

Shapley, L. S. (1953). "Stochastic Games". In: *Proc. Nat. Academy Sci., Math.* 39, pp. 1095–1100.

Shoham, Y., R. Powers, and T. Grenager (2003). *Multi-agent Reinforcement Learning: A Critical Survey*. Techinical Report. Stanford University.

Skyrms, B. (2004). *The Stag Hunt and the Evolution of Social Structure*. Cambridge University Press, Cambridge, UK.

Stone, P. and M. Veloso (2000). "Multiagent Systems: A Survey from a Machine Learning Perspective". In: *Autonomous Robotics* 8.3, pp. 345–383.

Sutton, R. S. and A. G. Barto (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.

Syed, U., B. Michael, and S. Robert E. (2008). "Apprenticeship learning using linear programming". In: *Proc. Intl. Conf. on Machine learning (ICML'08)*. Helsinki, Finland, pp. 1032–1039. ISBN: 978-1-60558-205-4.

Syed, U. and R. E. Schapire (2007). "A game-theoretic approach to apprenticeship learning". In: *Proceedings of the 20th Advances in Neural Information Processing Systems, NIPS'07*. Vol. 20, pp. 1–8.

Todd, A, P Beling, and W Scherer (2016). "Crossed and locked quotes in a multi-market simulation". In: *PloS one* 11 (3).

Valtazanos, A. and S. Ramamoorthy (2011). "Intent inference and strategic escape in multi-robot games with physical limitations and uncertainty".

In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'11*.

Wakefield, J. (2016). *Foxconn replaces 60,000 factory workers with robots*. BBC. URL: http://www.bbc.com/news/technology-36376966.

Waugh, K., B. Ziebart, and J. Bagnell (2011). "Computational Rationalization: The Inverse Equilibrium Problem". In: *Proc. Intl. Conf. Mach. Learning (ICML'11)*, pp. 1169–1176.

Yang, Steve Y et al. (2015). "Gaussian process-based algorithmic trading strategy identification". In: *Quantitative Finance* 15.10, pp. 1683–1703.

Zhang, Y. and D. Brown (2014). "Simulation Optimization of Police Patrol Districting Plans Using Response Surfaces". In: *SIMULATION: Transactions of The Society for Modeling and Simulation International* 90 (6), pp. 687–705.

Zhao, Y., S. Patek, and P. Beling (2008). "Decentralized Bayesian Search using Approximate Dynamic Programming Methods". In: 38.4, pp. 970–975.

Ziebart, B. D. et al. (2008). "Maximum Entropy Inverse Reinforcement Learning". In: *Proc. Nat. Conf. Artif. Intell (AAAI'08)*. Vol. 3, pp. 1433–1438.