# Decision Markets Implementations - for Human Forecasters and Multi-agent Learning Systems

A thesis presented in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

at Massey University, Albany,

New Zealand.

Wenlong Wang

2023

# Contents

# List of Tables

# List of Figures

# Abstract

Mechanisms of collective decision-making are an increasingly important topic, given that relevant data and information are often distributed. Collective decision-making processes involve eliciting information from multiple agents, aggregating the information, and mapping the aggregated information to a decision. An obstacle to these processes is that information is often proprietary, held by self-interested agents, and sometimes even too sensitive to share. Decision markets are mechanisms for eliciting and aggregating such information into predictions for decision-making. A design for decision markets put forward by Chen, Kash, Ruberry, *et al.* uses prediction markets to elicit and aggregate predictions that are conditional to the available actions, and then uses a stochastic decision rule to determine, based on the aggregated forecasts, which action to select. The design is incentive-compatible and uses a decision scoring rule to evaluate and incentivise the self-interested agents for their forecasts.

The first part of this thesis (Chapter 2) describes a framework for security-based decision markets that allows agents to make predictions by trading assets. Security-based decision markets are designed to be user-friendly for participants familiar with trading in stock markets. For prediction markets, such a framework is well studied. For decision markets, my results show there are important differences between scoring rule based and securities-based implementation.

The second and third parts of this thesis (Chapters 3 and 4) investigate decision markets as mechanisms of collective decision-making for multi-agent learning problems, thus building a bridge between economic mechanisms and artificial intelligence. Chapter 3 provides a decision market based algorithm that allows a principal to train multiple autonomous agents with independent and identically distributed (iid) information to solve a contextual bandit problem. Simulation results demonstrate that the proposed multi-agent systems can achieve performance equivalent to a centralised counterpart without requiring direct access to the agents' iid information, which is necessary for the centralised counterpart.

Chapter 4 describes a set of mechanisms that allow avoiding stochastic decision rules to select actions based on aggregated forecasts. This is important because committing to a stochastic (i.e.,

randomising) decision rule means that sometimes suboptimal decisions have to be taken. The mechanisms outlined in this chapter require agents to collectively predict a proxy instead of conditional outcomes. Simulations show that the performance is as good as a Bayesian model with access to all distributed information.

# Acknowledgements

I would like to express my sincere gratitude to my advisor, Prof. Thomas Pfeiffer. Without his patient guidance and unwavering support, this thesis would not be possible. You have been a mentor on my academic path and a friend in my life. I would also like to thank my co-supervisor, Dr. Tong Liu, for your insightful input. I am grateful to Dr. Pansye Elkashef and Dr. Michael Gordon for the delightful journey with the research team, for the unforgettable trip to Paris, and for your friendship and support.

I would like to thank my father for your unconditional support and encouragement. I would also like to thank my mother, who is no longer with us, for your infinite love and care. I would not have been who I am without you. You will always be with me deep in my heart. Thank you, my grandparents; you are my faithful supporters who always believe in me. I would also like to thank the rest of the family.

To my close friends, all of you have been my strongest shield against all the stress, loneliness, and unhappiness from my work and life. Therefore, you indirectly contributed to my thesis. Thank you.

Thank you to all the faculty at the New Zealand Institute for Advanced Studies and the Department of Computer Science. I would also like to thank all the administrative staff for their help and assistance. Thanks to Vesna Davidovic-Alexander for her letters that helped me secure my visa. I wish to convey my gratitude to Dr. Jiamou Liu, Dr. Bo Liu and Prof. Sarah Rajtmajer for being my PhD examiners and their valuable suggestion. I am grateful to Prof. Mark Waterland for his significant contributions as the convener.

Finally, but most importantly, to my love, Jinling Li: You have been a patient listener, an optimistic encourager, and a caring girlfriend. Thank you for all your love and support.

# Chapter 1

# Introduction

When facing a decision and aiming to achieve the desired outcome with the highest possible probability, a decision maker wants to collect all the relevant information to assist the decision-making. However, the decision maker, whom I refer to as the principal in this thesis, often does not possess all the relevant information and cannot obtain it without a cost because the information is often distributed and held by self-interested agents. Existing studies often describe such costs as incentives and use so-called scoring rules to map a probabilistic forecast and the materialisation of the corresponding future outcome to a score. When a future event is not conditional on a decision (for example whether it will rain tomorrow) this problem is well investigated. That is, a prediction market can elicit the information with incentives and aggregate the distributed information into a reliable forecast [2]–[4]. Evaluating a prediction for a future that is conditional on a decision is mechanistically more problematic.

## 1.1 Decision Markets

The mechanisms to resolve this problem are called decision markets [5]. The key problem for providing scores (or rewards) for conditional forecasts is that the counterfactual future outcomes are not observable, and therefore scoring rules cannot evaluate the corresponding forecasts. Consider, for instance, a doctor who wants to decide between two treatments for a patient. The doctor asks a group of specialists for advice on two treatments and decides to treat the patient with the first alternative. The outcome of the second treatment becomes counterfactual, and the advice for the second treatment cannot be evaluated with scoring rules. A simple solution is to evaluate the forecasts for the executed action and nullify the forecasts for the other actions. However, this solution will incentivise

manipulative behaviours [6], where forecasters make inaccurate predictions to obtain a larger reward. This problem can be mitigated with a stochastic decision rule, which maps the forecasts to a probability distribution with full support over all actions [1]. This solution ensures that regardless of which action will be selected, forecasts for each action are assigned a score which maximises expectation if the forecasts are accurate. However, a disadvantage of this solution is that the principal sometimes has to select an action which is predicted to be sub-optimal because the action is sampled from a fully supported distribution. The principal can mitigate the disadvantage by assigning a small probability to the estimated sub-optimal actions, but this will result in a higher worst-case loss for the principal because the worst-case loss grows with the inverse of the smallest action probability. Manipulations and the related advantages and disadvantages are reoccurring themes throughout the thesis, and will be discussed in much greater detail in the individual chapters.

## 1.2 Novel Contributions by Chapter

In real-world prediction markets, scoring rules are often implemented with securities trading as this is favourable for participants familiar with securities trading interfaces as used for instance in stock exchange markets. Prediction markets that employ strictly proper scoring rules can be implemented in a securities-trading manner by market scoring rules or cost functions [2], [7]. For decision markets, however, an investigation into implementations with securities trading has so far been missing. Chapter 2 is an extended version of a paper published under the title "Securities Based Decision Markets", which fills this gap [8]. The paper proposes a novel framework to implement decision markets with stochastic decision rules that allow for securities trading. Compared to prediction markets, where scoring rules and asset trading are largely equivalent, securities-based decision markets are shown to provide additional flexibility for the principal to alleviate the worst-case loss even with small probabilities for the estimated sub-optimal actions by specially designed contracts.

Decision-making is also heavily studied in the artificial intelligence community. Repeated decision-making problems with incomplete information are studied under the framework of multi-arm bandit problems [9]. The name comes from an analogy that an agent playing a slot machine with multiple levers. Each lever (arm or action) is associated with an unknown reward distribution. The goal of the agent is to accumulate as much reward as possible. Therefore, the agent needs to estimate a reward distribution for each lever according to its experience and balance between making an informed decision on which lever to pull next. The most frequently studied question in multi-armed bandit

problems is how to balance between exploration for a better-estimated reward distribution and exploiting the existing knowledge. Sometimes, the reward distributions are non-stationary, and a piece of side information (signal) is provided as a proxy for the reward distribution. Such a multi-armed bandit problem extends to a contextual bandit problem. Consider, for example, that Netflix wants to recommend a movie to a customer. Since different customers have different preferences, the reward distribution of each movie recommendation for each customer is different. Therefore, the user profiles serve as side information to the unknown reward distribution determined by whether the user will watch the recommended movie. The reward helps the recommendation system to update its computational model and become more and more accurate in movie recommendations. Such a process is referred to as bandit learning. Contextual information like data from user profiles is often proprietary, distributed and sometimes sensitive in the real world. Building a user profile database is time and finance consuming and accessing a proprietary user profile database, such as Netflix's user profile database, can be even more expensive, if not impossible. Therefore, a framework that allows eliciting forecasts for decision-making from multiple self-interested sources is advantageous. For instance, a startup shopping website could enquire with Google and Amazon for a joint recommendation about its own goods based on their user profiles and recommendation systems, and pays a reward depending on whether the customer purchases the recommended goods.

Chapter 3 is a preprinted paper titled "Decision Market Based Learning For Multi-agent Contextual Bandit Problems" which outlines and investigates a novel system that uses decision markets in multi-agent bandit learning [10]. The system can gather predictions from agents who possess information inaccessible to the system concerning various alternative actions. The rewards allocated to these agents are both incentive-compatible and structured in a manner that facilitates the agents' ability to learn from the rewards, enabling them to effectively map contextual information to accurate forecasts. Incentive compatibility ensures that the reward can be seamlessly translated into monetary payoff, fostering a collaborative synergy among rational human agents and artificial models in the creation of a human-machine hybrid system. This feature is connected to Chapter 2, wherein we offer a user-friendly interface for human forecasters. I investigate how efficiently such a system can be trained to make collective decision, compared to a centralised counterpart that can access all the private information. Simulations of the training process show that the multi-agent system can achieve a similar efficiency to the centralised counterpart.

Researchers recently proposed to study, evaluate and express many classic economic problems, such as optimal control and decision-making, in multi-armed bandit learning or reinforcement learning

frameworks, because these frameworks do not require complex reasoning and expectation calculation [11]. Bandit learning and reinforcement learning allow evaluating the dynamics of agent learning in repeated games although reaching a global Nash equilibrium is not always guaranteed [12]. In Chapter 3, I use the decision market based multi-agent learning system to investigate the dynamics among agents in a decision market with a deterministic decision rule which is not incentive compatible [6], [13]. I find a surprising result in a three-agent simulation where three agents make 'strategically distorted' reports, but nevertheless, the aggregated reports are accurate.

Decision markets with stochastic decision rules have limitations because the principal has to commit to a randomised decision rule, which is a barrier to practical applications [1]. It is in the principal's interest to always select the action predicted to yield the most desirable outcome. For instance, in the previous treatment example, choosing a sub-optimal treatment based on predictions, when a better alternative is predicted to exist, appears to be unethical. For elicitation of information from a single agent, a recommendation rule has been proposed and formalised that allows the principal to select the action deterministically [1], [6]. However, this mechanism lacks information aggregation. Chapter 4 proposes several novel, related mechanisms to fill this gap. I first assume the principal also has information about at least one of the actions, and uses it as a proxy which is statistically correlated with the outcome of the action. The distinction of the mechanism is that it requires the agents to predict the proxy rather than the outcomes of the actions. This mechanism relies on ideas from peer prediction mechanisms. Peer prediction mechanisms solve the problem of how to evaluate forecasts without verification. Therefore, the agents' rewards only depend on the information the principal possesses and become independent of the specific decision the principal will make. I further propose expanded mechanisms that lift the assumption that the principal possesses a suitable proxy. I use the multi-agent learning system proposed in Chapter 3 to simulate the dynamics of learning in these mechanisms, and obtain promising results. Chapter 5 concludes the thesis and discusses limitations and future studies.

## 1.3 Examples and Use Cases

In this section, we will provide a more detailed explanation of the previous examples. There is substantial common ground between decision markets, which originated in economics, and multi-armed bandit problems from computer science, and we will use the previous example as bridges to link the terminology from these two subjects.

In the previous treatment example, we have a doctor who makes a decision between a treatment A and a treatment B. The objective of the doctor is to select the treatment that maximizes the chances of recovery within a certain time. The doctor knows the patient, and thus has some information (i.e., private signals) about what treatment may work best for the patient. Note that the doctor's knowledge can be condensed into a conditional forecast for the patient's recovery conditional on the treatment. However, the doctor has limited expertise with the specific treatments, and thus consults with an expert for treatment A and an expert for treatment B. In a decision market, the first expert will revise the initial opinion of the doctor, and the second expert will further revise it. Given their expertise, the first expert will particularly revise the prospective outcomes if treatment A is selected, and the second expert will do the same for treatment B.

In a human/AI hybrid decision making process, the experts might be computational agents with access to large datasets about treatments A and B, respectively, and importantly the agents do not need to share their data, but solely need to share conditional forecasts to come to a joint decision. In the context of computer science, the problem transforms into a multi-agent contextual bandit problem. The doctor faces a two-arm bandit problem, with each of the two treatments representing one arm, and with the contextual information being distributed over the doctor and the two (possibly computational) experts. Each expert, in turn, faces a continuous contextual bandit problem, because the revised conditional probabilities are typically continuous. Once the doctor reviews the aggregated report, they will apply a decision rule to correlate the combined information with an appropriate treatment. Subsequently, the doctor will monitor the patient's condition after the treatment and can determine a reward for each specialist involved. For human specialists, the reward could be monetary, while for artificial agents, the reward is a score can be utilised for learning purposes. Therefore, decision markets can be utilized for hybrid system that can seamlessly collaborate with both humans and AI.

The structure and constraints in this example resemble Federated Bandit Learning (FBL), a topic which is receiving considerable interest. Federated Bandit Learning is a scheme in which multiple agents collaborate to solve a multi-armed bandit learning problem while maintaining their learning data locally. Similar to our own scenario, the data is distributed diversely among these multiple agents, and there are constraints in the sharing of data. A crucial distinction between our approach and FBL is that we presume agents to be self-interested rather than collaborative.

## 1.4 Final Considerations

As Chapters 2, 3 and 4 are published or prepared to be published as independent papers, each has its own abstract, introduction, related work and conclusion sections, and some sections are overlapping. Readers interested in scoring rules, prediction markets and decision markets, can find a detailed introduction and related work at the beginning of Chapter 2. Multi-agent learning and multi-armed bandit problems, and related work are introduced at the beginning of Chapter 3. In Chapter 4, readers can additionally find related work about peer prediction. Since the chapters differ in their use of notation, an overview of the notation used throughout the chapter is provided at the beginning of each chapter.

There are many relations between the topics of these chapters. Chapter 2 provides an easy-to-use securities trading interface targeting humans (with a provision for potential human/AI hybrid systems). It also provides a channel for market makers to leverage financial tools to mitigate worst-case losses in a securities-based setting. Chapter 3 describes a multi-agent learning system that covers the AI side of potential human/AI hybrid systems. It offers a novel way to perform decentralised learning without accessing the local data but still produces equivalent decision quality as a centralised model that can access all the data. It also investigates the question of deterministic vs. stochastic decision rules, and provides insights into the dynamics of learning agents in an incentive incompatible market. Chapter 4 is about incentive-compatible decision-making with aggregated information in a deterministic manner. In principle, this applies to both human and computational agents (therefore relating to Chapters 2 and 3). It uses the same learning paradigm as Chapter 3 and eliminates the need for stochastic decision rules, which are necessary in Chapter 2.

# Chapter 2

# Securities based decision markets

Chapter 2 is an extended version of the paper *'Securities based decision markets'* that was published in the '*Third International Conference on Distributed Artificial Intelligence*' in 2021. This chapter describes an implementation of decision markets where forecasts are made through the trading of securities. Such an implementation will facilitate a convenient way of using decision markets by human participants, especially traders with experience in securities trading. This chapter serves as a literature review for strictly proper scoring rules, prediction markets and decision markets, which are the prerequisite contexts for the subsequent chapters.

## Summary of Notation

| | |
|---|---|
| $\Omega$ | set of possible outcomes in a prediction market, $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$ |
| $\omega_i$ | outcome with index $i$ |
| $r_i$ | probabilistic report for outcome $\omega_i$ |
| $\vec{r}$ | probabilistic reports over all outcomes in a prediction market |
| $s_i(\vec{r})$ | scoring rule that quantifies the accuracy of $\vec{r}$ once $\omega_i$ materialises |
| $\vec{p}$ | forecasters' belief |
| $G(\vec{p}, \vec{r})$ | expected payoff of a forecaster who reports reports $\vec{r}$ and believes $\vec{p}$ |
| ${}^*\vec{r}$ | updated probabilistic reports in sequential reporting |
| $\vec{q}$ | quantity of outstanding securities in a securities based prediction market |
| $\vec{r}(\vec{q})$ | security prices when outstanding securities are $\vec{q}$; this is equivalent to probabilistic reports in a prediction market with sequential reporting |
| $c(\vec{q})$ | cost of outstanding securities $\vec{q}$ |

| | |
|---|---|
| $\mathcal{A}$ | set of available actions in a decision market, $\mathcal{A} = \{\alpha_1, \alpha_2, ..., \alpha_m\}$ |
| $a_j$ | action with index $j$ |
| $\Omega_j$ | set of outcomes for action $a_j$ in a decision market |
| $\omega_i^j$ | outcome with index $i$ of action $a_j$ |
| $\vec{\phi}(\vec{r})$ | decision rule that depends on the final forecasts $\vec{r}$ in a decision market |
| $\vec{r}_j$ | set of probabilistic reports for action $a_j$ in a decision market |
| $S_i^j(\vec{r}_j)$ | decision scoring rule that quantifies the accuracy of $\vec{r}_j$ when action $a_j$ is selected and outcome $\omega_i^j$ materialises; this decision score also depends on probability $\phi_j$ of the selected action $a_j$ |
| $G$ | expected payoff of a forecaster in a probabilistic report based decision market |
| $\hat{G}$ | expected payoff of a forecaster in a securities based decision market |

## Abstract

Decision markets are mechanisms for selecting one among a set of actions based on forecasts about their consequences. Decision markets that are based on scoring rules have been proven to offer incentive compatibility analogous to properly incentivised prediction markets. However, in contrast to prediction markets, it is unclear how to implement decision markets such that forecasting is done through the trading of securities. We here propose such a securities based implementation, and show that it offers the same expected payoff as the corresponding scoring rules based decision market. The distribution of realised payoffs, however, might differ. Our analysis expands the knowledge on forecasting based decision-making and provides novel insights for intuitive and easy-to-use decision market implementations.

## 2.1 Introduction

Prediction markets [2], [3], [14]–[22] are popular tools for aggregating distributed information into often highly accurate forecasts. Participants in prediction markets trade contracts with payoffs tied to the outcome of future events. The pricing of these contracts reflects aggregated information about the probabilities associated with the possible outcomes. A frequently used contract type is Arrow-Debreu securities that pay \$1 when a particular outcome is realised, and otherwise pay \$0. If such a security is traded at \$0.30, this can be interpreted as forecast for that outcome to occur at 30% chance. Potential caveats with the interpretation of prices in prediction markets as probabilities have been discussed in the literature [3], [18], but are not seen as critical for typical applications [2], [3], [18]. Prediction markets have been extensively investigated in lab based experiments and real world settings [3], [14], [16]–[19], [21], [23].

In many practical prediction markets applications, such as recreational markets on political events, participants trade directly with each other, and one participant's gain is the other participant's loss. Prediction markets can, however, also be designed to offer net benefits to the participants. Such incentivised prediction markets can be used by a market creator who is willing to compensate the market participants for the information obtained from the market [2], [15], [20]. Incentivised prediction markets rely on market maker algorithms to trade with the participants, and on cost functions to update prices based on past transactions. These functions are closely related to proper scoring rules such as the Brier (or quadratic) scoring rule and the logarithmic scoring rule [24], [25], which measure the accuracy of forecasts and allow rewarding a single expert based on forecast and actual outcome. The market maker in an incentivised prediction market subsidises the entire market rather than single experts; its worst-case loss is finite and its expected loss depends on how much the participants 'improve' on the information entailed by the initial market maker pricing [2].

Accurate forecasts, as obtained from prediction markets, can be of tremendous value for decision makers. Commercial companies, for instance, can benefit substantially from accurate forecasts regarding the future demand for their products. However, many decision-making problems require conditional forecasts [26]. To decide, for instance, between alternative marketing campaigns, a company needs to understand how each of the alternatives will affect sales. In other words, it needs to predict, and choose between, 'alternative futures'. To implement such forecasting based decision-making, Hanson [5] proposed so called decision markets. While it is non-trivial to properly incentivise participants to provide their information in decision markets, it has been shown that this can be achieved [1], [6],

[13], [26], [27].

Properly incentivised decision markets work in a stepwise process to select one among a number of mutually exclusive actions. First, forecasts about the expected future consequences of each action are elicited in a step analogous to incentivised prediction markets. Second, a decision rule is used to select an action based on the forecasted consequences. Once an action has been selected, and its consequences are revealed, payoffs are provided for the forecasts as elicited in the first step. Importantly, the decision rule in properly incentivised decision markets is stochastic, with each action being picked with a strictly positive probability [13], [26]. Payoffs are scaled up to ensure that the participants' expected payoffs in decision markets remain analogous to those made in properly incentivised prediction markets [13], [26], and that game-theoretical results on strategic interactions between participants in prediction markets [15] carry over.

The literature on decision markets has so far focused on implementations based on scoring rules. For prediction markets it is well established how to implement properly incentivised forecasting such that forecasts are made through the trading of securities. A similar securities based decision market implementation has however not yet been described. Such an implementation is important because participants in decision markets are likely familiar with ordinary asset trading, and it is thus convenient for them to report their forecasts through a securities trading interface. Furthermore, securities based decision markets simplify managing liabilities, because the payment for the purchased securities covers the traders' worst-case loss. We here propose such a securities based decision market setting, and compare it to existing, scoring rule based decision markets.

The remaining manuscript is organised as follows: In section 2.2 we briefly introduce scoring rules, sequentially shared scoring rules, prediction markets and scoring rule based decision markets. In section 2.3, we describe a securities based market design and compare it with the existing scoring rule based decision markets. In section 2.4 we compare our design with the the scoring rule based decision markets mechanism in terms of worst-case losses. In section 2.5, we discuss how trading in this setup allows to re-allocate worst-case losses. Finally, in section 2.6, we conclude and discuss our future work.

## 2.2 Related Work and Notation

### 2.2.1 Scoring Rules

Let us define $\Omega$ as a finite set of mutually exclusive and exhaustive outcomes $\{\omega_1, \omega_2, ..., \omega_n\}$. A probabilistic prediction for those outcomes is denoted by $\vec{r} = (r_1, r_2, ..., r_n)$ with $\sum_{x=1}^{n} r_x = 1$ and

$r_i \in [0, 1]$. A scoring function $s_i(\vec{r})$ allows to quantify the accuracy of prediction $\vec{r}$ once the outcome $\omega_i$ materialises [28].

Scoring rules allow to incentivise forecasters for predictions. Denoting the reported distribution as $\vec{r} = (r_1, r_2, \ldots, r_n)$ and the forecasters' belief as $\vec{p} = (p_1, p_2, \ldots, p_n)$, the expected payoff for a forecaster is given by

$$G(\vec{p}, \vec{r}) = \sum_{k=1}^{n} p_k s_k(\vec{r})$$

A scoring rule is defined as proper if a forecaster maximises his/her expected payoff by truthfully reporting what he/she believes.

$$G(\vec{p}, \vec{p}) \geq G(\vec{p}, \vec{r})$$

Furthermore, a scoring rule is strictly proper if $G(\vec{p}, \vec{p}) > G(\vec{p}, \vec{r})$ for all $\vec{r} \neq \vec{p}$.

### 2.2.2 Sequentially Shared Scoring Rules

Because information is often distributed across multiple agents, it is of interest to expand proper scoring to elicit forecasts from groups of forecasters. In his work on incentivised prediction markets, Hanson proposed a mechanism to sequentially elicit information from forecasters [2], [20]. The mechanism keeps a current report $\vec{r}$ and offers a contract for a new report ${}^*\vec{r}$ to be scored as $s_i({}^*\vec{r}) - s_i(\vec{r})$ if the outcome $\omega_i$ is observed. Note that $s_i({}^*\vec{r}) - s_i(\vec{r})$ is a proper scoring rule if $s_i$ is a proper scoring rule. Once a forecaster accepts the offer, the decision maker will update the current report from $\vec{r}$ to ${}^*\vec{r}$ and allow a next forecaster to further modify the new current report. Under such a sequentially shared scoring rule, forecasters are scored for how much they improve or worsen the current report. Such a mechanism uses incentives efficiently in that it avoids paying for the same information twice.

### 2.2.3 Securities based Prediction Markets

The mechanism described in section 2.2.2 involves a two-sided liability. The decision maker is liable to pay each forecaster who improves a forecast, and forecasters are liable to pay the decision maker if they worsen a forecast. It is often considered convenient to implement sequentially shared scoring rules through the trading of Arrow-Debreu securities [2], [20]. In such an implementation, forecasters purchase securities from the market maker and their payments cover their liabilities. Another reason to use securities based trading is that the majority of existing real-world prediction markets, such as recreational markets on sports or political events, are trading securities in a double auction process. Traders who are familiar with these prediction markets will prefer an interface to be expressed in

terms of trading with securities.

To incentivise traders, securities are bought and sold by a market creator. The market creator uses a market maker algorithm which keeps track of past trades and sets security prices derived from a cost function. The total amount spent on purchasing a particular quantity of securities can be calculated from this cost function. We denote the quantity of outstanding securities as $\vec{q} = (q_1, q_2, \ldots, q_n)$ for a market on $n$ mutually exclusive and exhaustive outcomes $\Omega$. Element $q_i$ represents the number of securities sold by the market creator that pay if outcome $\omega_i$ is observed. The instantaneous prices of the securities with outstanding quantities $\vec{q}$ are denoted as $\vec{r}(\vec{q})$ and play the same role as reports in scoring rule based markets.

Assume a trader wants to change the outstanding securities distribution from $\vec{q}$ to ${}^*\vec{q}$ by buying securities to change the price from $\vec{r}$ to ${}^*\vec{r}$. The cost for the trader to purchase the amount of securities ${}^*\vec{q} - \vec{q}$ can be calculated from $C({}^*\vec{q}) - C(\vec{q})$, where $C(\vec{q})$ denotes a cost function. Once the final event, i.e. $\omega_i$ is observed, the market maker will resolve the market by paying \$1 for each winning security. If the trader holds ${}^*q_i - q_i$ securities when the market is resolved, his/her payout will be $\$({}^*q_i - q_i)$. Overall the realised payoff for the trader will be

$$({}^*q_i - q_i) - (C({}^*\vec{q}) - C(\vec{q}))$$

Chen and Pennock generalised the relationship between cost functions, price functions and scoring rules, and proposed three equations that establish their equivalence [7]:

$$\begin{cases} s_i(\vec{r}) = q_i - C(\vec{q}) & \forall i \\ \sum_i r_i(\vec{q}) = 1 \\ r_i(\vec{q}) = \frac{\partial C(\vec{q})}{\partial q_i} \end{cases} \tag{2.1}$$

Furthermore, Chen and Vaughan proved that there exist a one-to-one mapping between any strictly proper scoring rule and cost function in securities based prediction markets and such a securities based market is incentive compatible [29]. Elicitation through scoring rule based and securities based prediction markets offers the same payoffs for participants when the markets start with the same initial forecasts and end with the same final forecasts.

Cost functions $C(\vec{q})$ and price functions $r_k(\vec{q})$ have the following properties:

$$
\begin{aligned}
C(\vec{q} + \beta\vec{1}) &= \beta + C(\vec{q}) \\
r_k(\vec{q} + \beta\vec{1}) &= r_k(\vec{q})
\end{aligned}
\tag{2.2}
$$

where $\beta$ is a real constant [7]. These properties imply that the same report can be made through different trades. If a trade ${}^*\vec{q} - \vec{q}$ changes market prices from $\vec{r}$ to ${}^*\vec{r}$, so does a trade ${}^*\vec{q} + \beta\vec{1} - \vec{q}$. This permits the trader to make any report by buying contracts from the market creator. Short selling is not required. The overall payoff will not be affected by the choice of $\beta$, because both costs of purchasing the contracts, and the payout from the contracts at resolution increase by the same amount.

### 2.2.4 Decision Markets

The design of decision markets expands prediction markets to use conditional forecasts for decision-making. Decision markets consist of two components. The first component is a set of conditional prediction markets, each of which elicits the forecasts for one of the actions. The second component is the decision rule that defines— based on conditional prediction markets forecasts— how the final decision will be made. An example is the MAX decision rule [6] which is to always select the action that has the highest predicted probability for a desired outcome to occur.

Decision markets with deterministic rules such as the MAX decision rule do not always properly incentivise a forecaster to truthfully report irrespective of the scoring rule it uses [6], [13]. An intuitive example to illustrate how a trader can benefit from misreporting is given in [6]. Chen, Kash, Ruberry, *et al.* described that a stochastic decision rule can myopically incentivise forecasters to truthfully report [26]. This approach is rephrased in the following in the notation used throughout this paper to allow for straight forward comparison with the securities-based implementation.

**Definition 1.** *In a decision market, the market creator has a finite set of $m$ actions $\mathcal{A} = \{\alpha_1, \alpha_2, ..., \alpha_m\}$ to choose from. For each action $\alpha_j$, there is a set of possible outcomes $\Omega_j = \{\omega_1^j, \omega_2^j, \ldots, \omega_{n_j}^j\}$, which $n_j$ is the number of possible outcomes for action $\alpha_j$. Both action set $\mathcal{A}$ and outcome sets $\Omega_j$ are collectively exhaustive and mutually exclusive. A stochastic decision rule $\vec{\phi}$ assigns a probability $\phi_k$ to each action $\alpha_k$ with $\phi_k > 0$ and $\sum_{k=1}^{m} \phi_k = 1$.*

Note that in our notation the outcome $\omega_i^j$ for action $\alpha_j$ can be unrelated to $\omega_i^k$ for action $\alpha_k$. In other words, outcomes can be specific to the actions. The sets for the outcome of two actions can be

completely disjoint. The decision rule can take the final report into account, i.e. $\vec{\phi} = \vec{\phi}(\vec{r}_1, \vec{r}_2, \ldots, \vec{r}_m)$ where $\vec{r}_1, \vec{r}_2, \ldots, \vec{r}_m$ are the final reports over the $m$ different actions. It can, for instance, approximate the MAX decision rule by assigning high probabilities to actions with desirable outcomes.

Similar to scoring rules in prediction markets, decision scoring rules can be defined to map forecasts, decisions and outcomes to a real number. For simplicity, we will denote this score as $S_i^j(\vec{r}_j)$ that the selected action is $\alpha_j$ and the observed event is $\omega_i^j$. Assume $s_i^j(\vec{r}_j)$ is a strictly proper scoring rule for conditional market $j$. A decision score for changing the current report from $\vec{r}_j$ to ${}^*\vec{r}_j$ is given by:

$$S_i^j({}^*\vec{r}_j) - S_i^j(\vec{r}_j) = \frac{1}{\phi_j}\left(s_i^j({}^*\vec{r}_j) - s_i^j(\vec{r}_j)\right) \tag{2.3}$$

The expected payoff $G$ of a forecaster in a scoring rule based decision market as defined in definition 1 is given by:

$$\begin{aligned} G &= \sum_{j=1}^{m} \phi_j \sum_{i=1}^{n_j} p_i^j \frac{1}{\phi_j}\left(s_i^j({}^*\vec{r}_j) - s_i^j(\vec{r}_j)\right) \\ &= \sum_{j=1}^{m} \sum_{i=1}^{n_j} p_i^j \left(s_i^j({}^*\vec{r}_j) - s_i^j(\vec{r}_j)\right) \end{aligned} \tag{2.4}$$

where $p_i^j$ denotes the belief of the forecaster which will be identical to ${}^*\vec{r}_j$ if the scoring rule is strictly proper. The forecaster has the same expected payoff as if he/she participated in $m$ independent and strictly proper prediction markets. Moreover, findings on strategic interaction between traders and incentives for instantaneous revelation of information from [15] apply as well.

Note that $\phi_j$ in equations (2.3) and (2.4) is the probability for the selected action in the decision rule after the final report. Equation (2.4) shows that the value of $\phi_j$ does not affect the expected payoff that risk-neutral forecasters seek to maximise. This is important because for scoring rules that depend on the final report, no participant except for the final forecaster knows the value of $\phi_j$. To provide truthful forecasts forecasters do not need the decision rule $\phi$ and its dependence on the final report as long as they can trust that the rule has full support.

An alternative method is to ask a single expert for a recommendation and use scoring rules to align the incentives of the expert with the market creator's interest [26], [30].

### 2.2.5 Empirical Work

There is a substantial body of empirical studies on prediction markets [3], [14], [16]–[19], [21], [23], and a number of empirical studies on decision markets [31]. Some of these studies have addressed

whether there are any differences in the efficiency of a scoring rule based mechanism vs. a securities based mechanism. Jian and Sami conducted laboratory experiments and found the performance of the two mechanisms is similar [32]; but they stated that further validation was required in 'field settings', which familiarity with how to report forecasts may be an important factor. While similar experimental comparisons in decision markets do not exist yet, it is worth noting that securities trading is frequently used in the experimental literature about decision markets [31].

## 2.3 Strictly Proper Securities Based Decision Markets

In section 2.2.3, we discuss the advantages of implementing forecasting through the trading of securities. We here formulate a cost function for securities based decision markets that offers the same expected payoff for participants as a scoring rule based decision market. In a prediction market, the cost function and price function can be calculated by solving the equations (2.1) [7]. However, because only decision markets with a stochastic decision rule are myopically incentive compatible, the stochastic decision rule needs to be accounted for.

### 2.3.1 Design

We adopt the cost function approach for prediction markets as described in section 2.2.3. To account for the stochastic decision rule, the securities traded in this market have payoffs that depend on the selected action.

**Definition 2.** *In addition of the notation in definition 1, we denote* $\vec{q}_j = (q_1^j, q_2^j, \ldots, q_{n_j}^j)$ *as outstanding securities for the conditional market for action* $\alpha_j$*. Element* $q_i^j$ *represents the number of securities sold by the market creator that pay if action* $\alpha_j$ *is selected and outcome* $\omega_i^j$ *is observed. The payout per security is denoted by* $v_j$*, and can depend on the selected action, but is the same for all traders and does not depend on the observed outcome. The payout for all other securities is zero. Cost function, price function and corresponding scoring rule for the conditional market on action* $\alpha_j$ *and outcome* $\omega_i^j$*, are denoted by* $C_j(\vec{q}_j)$*,* $r_i^j(\vec{q}_j)$ *and* $s_i^j(\vec{r}_j)$*, respectively, and together fulfil equation 4.*

**Theorem 1.** *Let a trader in a securities based decision market as defined in definition 2 make a trade* ${}^*\vec{q}_j - \vec{q}_j$ *to move prices from* $\vec{r}_j(\vec{q}_j)$ *to* $\vec{r}_j({}^*\vec{q}_j)$*. Then the trader will have the same expected payoff as a forecaster who makes the same forecast in a scoring rule based decision market as described in equation* (2.4) *if and only if we set* $v_j = 1/\phi_j$ *for all action* $\alpha_j$*.*

*Proof.* Let a forecaster in the a scoring rule based decision market change the reports from $\vec{r}_k$ to ${}^*\vec{r}_k$ for any action $\alpha_k$. The expected payoff of the forecaster is denoted as $G$ and is given in the equation (2.4). Let a trader in our securities based decision market change the outstanding securities distribution from $\vec{q}_k$ to ${}^*\vec{q}_k$ for each action $\alpha_k$ such that prices change from $\vec{r}_k$ to ${}^*\vec{r}_k$. Then the realised payoff the trader gains from such a trade is given by:

$$v_j \left( {}^*q_i^j - q_i^j \right) - \sum_{k=1}^{m} \left( C_k({}^*\vec{q_k}) - C_k(\vec{q_k}) \right)$$

where the selected action is $\alpha_j$ and the observed outcome is $\omega_i^j$.

The expected payoffs of the trader is denoted as $\hat{G}$ and we obtain:

$$\begin{aligned} \hat{G} =& \sum_{j=1}^{m} \phi_j \sum_{i=1}^{n_j} p_i^j \left( v_j \left( {}^*q_i^j - q_i^j \right) - \sum_{k=1}^{m} \left( C_k({}^*\vec{q_k}) - C_k(\vec{q_k}) \right) \right) \\ =& \sum_{j=1}^{m} \phi_j v_j \sum_{i=1}^{n_j} p_i^j \left( {}^*q_i^j - q_i^j \right) - \sum_{k=1}^{m} \left( C_k({}^*\vec{q_k}) - C_k(\vec{q_k}) \right) \end{aligned} \tag{2.5}$$

Substituting equation (2.1) into the equation (2.5), we obtain:

$$\begin{aligned} \hat{G} =& \sum_{j=1}^{m} \sum_{i=1}^{n_j} p_i^j \left( {}^*q_i^j - q_i^j \right) - \sum_{k=1}^{m} \left( C_k({}^*\vec{q_k}) - C_k(\vec{q_k}) \right) + \sum_{j=1}^{m} (\phi_j v_j - 1) \sum_{i=1}^{n_j} p_i^j \left( {}^*q_i^j - q_i^j \right) \\ =& G + \underbrace{\sum_{j=1}^{m} (\phi_j v_j - 1) \sum_{i=1}^{n_j} p_i^j \left( {}^*q_i^j - q_i^j \right)}_{\text{a}} \end{aligned} \tag{2.6}$$

The expected payoff $\hat{G}$ in a securities based market is equal to the expected payoff $G$ in a scoring rule based market if and only if term a in equation (2.6) is zero. One way to achieve this for arbitrary trades is to set the payoffs of the contracts $v_j$ to $1/\phi_j$. Thus $G = \hat{G}$ if $v_j = 1/\phi_j$.

An alternative with $v_j \neq 1/\phi_j$ would be to choose $v_j$ such that the vector (dot) product $\vec{a} \cdot \vec{b}$, with vector element $a_j$ being defined as $\sum_{j=1}^{m} (\phi_j v_j - 1)$ and $b_j$ being defined $\sum_{i=1}^{n_j} p_i^j \left( {}^*q_i^j - q_i^j \right)$, becomes zero. This however, would require to make $v_j$ dependent on trade-specific quantities such as the ${}^*q_i^j$ and contradicts the properties of contract payoffs as defined in definition 2.

□

### 2.3.2 Distribution of Realised Payoffs

Securities based decision markets and corresponding scoring rule based decision markets provide the identical expected payoff for participants under the same conditions. However, the actual distribution of payoffs for the participants are not necessarily the same. In this subsection, we will discuss the difference between securities based decision markets and the corresponding scoring rule based decision market in terms of realised payoffs for participants.

The realised payoffs for a forecaster who changes report $\vec{r_k}$ to ${}^*\vec{r_k}$ in a scoring rule based decision market is given by equation (2.3). In the securities based market, assume a trader makes a trade to change the price for any action $\alpha_k$ from $\vec{r_k}$ to ${}^*\vec{r_k}$ . This trade changes the market creator inventory from $\vec{q_k}$ to ${}^*\vec{q_k}$ and has a cost given by $C_k({}^*\vec{q_k}) - C_k(\vec{q_k})$. Let the market creator select the action $\alpha_j$ to execute and observe the outcome $\omega_i^j$. Using equation (2.1) we obtain the realised payoffs for the trader in the securities based decision market:

$$\begin{aligned} & \frac{1}{\phi_j}\left({}^*q_i^j - q_i^j\right) - \sum_{k=1}^{m}\left(C_k({}^*\vec{q_k}) - C_k(\vec{q_k})\right) \\ = \quad & \frac{1}{\phi_j}\left(s_i^j({}^*\vec{r_j}) - s_i^j(\vec{r_j})\right) + \underbrace{\frac{1}{\phi_j}\left(C_j({}^*\vec{q_j}) - C_j(\vec{q_j})\right) - \sum_{k=1}^{m}\left(C_k({}^*\vec{q_k}) - C_k(\vec{q_k})\right)}_{\text{a}} \end{aligned} \tag{2.7}$$

The term a in equation (2.7) shows that there is a difference in realised payoffs of participants between the securities based decision market and the scoring rule based decision market. This difference cancels out when computing the expected payoff of a participant. Although the sign of term a cannot be decided easily, term a will increase when the trader spends more in the selected conditional market and decrease when the trader spends more in the conditional markets that are not selected.

Equation (2.7) shows that the realised payoffs in our securities based decision market can be rewritten as a scoring rule with an additional term a. This term can be interpreted as a 'lottery' that costs $I$ and returns $I/\phi$ at probability $\phi$. Equation (2.7) can be generalised as:

$$\frac{1}{\phi_j}\left(s_i^j({}^*\vec{r_j}) - s_i^j(\vec{r_j})\right) + \underbrace{\frac{1}{\phi_j}I_j - \sum_{k=1}^{m} I_k}_{\text{a}} \tag{2.8}$$

Equation (2.7) is a special case of equation (2.8) where $I_k$ equals to the total cost the trader spent in conditional market $k$ for each $k$. This can be interpreted as a forecaster making a report in a scoring rule based decision market and simultaneously participates in a 'lottery' at a cost that depends on

the report. Any securities based decision market can be seen as a market where the market creator offers a scoring rule based decision market along with a 'lottery' with an expected payoff of zero. The terms $I_k$ can be chosen such that the payoffs from a scoring rule based decision market are the same as for a securities based decision market. In the following we provide an example to illustrate the differences between the payoffs in securities based and scoring rule based decision markets. In Section 2.4 and Section 2.5 we will show how the additional flexibility in the design of securities based decision markets can be used to reallocate liabilities and worst case losses between traders and market creator.

**Example: investment into a single conditional market.**

To further detail the differences between a securities based decision market and the corresponding scoring rule based decision market, we analyse an example where a trader invests in only one conditional market, the market corresponding to action $\alpha_k$. We focus on a setting where forecasters do not engage in 'short selling', and thus only hold positive positions, and the cost $C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)$ paid to the market creator is positive. Assume the market creator to select the action $\alpha_j$ and to observe the outcome $\omega_i^j$. We will compare the realised payoffs between securities based decision markets and corresponding scoring rule based decision market under two conditions.

The realised payoffs for a trader who invested into the selected market, i.e. $j = k$, can be found in table 2.1. Naturally, the forecaster in corresponding scoring rule based decision market is also assumed to report in the selected conditional market.

Table 2.1: Realised payoff difference between a decision scoring rule based decision market and the corresponding security based decision market when participants invest into or report on the selected action.

| **Market Type** | **Realised Payoff** |
|---|---|
| Scoring Rule Based | $\frac{1}{\phi_j}\left(s_i({}^*r_j) - s_i(r_j)\right)$ |
| Securities Based | $\frac{1}{\phi_j}({}^*q_i^j - q_i^j) - (C_j({}^*\vec{q_j}) - C_j(\vec{q_j}))$ |

Using equation (2.1) for the realised payoffs of securities based decision market in table 2.1 we obtain:

$$\begin{aligned} & \frac{1}{\phi_j}\left({}^*q_i^j - q_i^j\right) - \left(C_j({}^*\vec{q_j}) - C_j(\vec{q_j})\right) \\ = \; & \frac{1}{\phi_j}\left(s_i({}^*r_j) - s_i(r_j)\right) + \underbrace{\frac{1-\phi_j}{\phi_j}\left(C_j({}^*\vec{q_j}) - C_j(\vec{q_j})\right)}_{\text{positive}} \end{aligned} \tag{2.9}$$

Assuming that traders can only hold positive positions, i.e. cannot 'short' securities, equation (2.9)

shows that participants gain a larger payoff in securities based decision markets compared to scoring rule based decision markets.

Table 2.2 shows the realised payoffs of the trader and the forecaster investing into, or report, on an unselected conditional market, i.e. $k \neq j$.

Table 2.2: Payoff difference between a decision scoring rule based decision market and the corresponding security based decision market when participants invest into or report on an unselected conditional market.

| **Market Type** | **Realised Payoff** |
|---|---|
| Scoring Rule Based | 0 |
| Securities Based | $-(C_k({}^*\vec{q_k}) - C_k(\vec{q_k}))$ |

Changing the prediction for a conditional market corresponding to an action that is not selected has a zero payoff in the scoring rule based decision market, regardless of how accurate the prediction is. This is because in the scoring rule based decision market, unselected conditional markets will be declared void. However, there is a cost for changing a prediction for an unselected market in the securities based market because purchasing shares to changing a prediction is costly.

## 2.4 Worst-case Losses for Participants and Market Creator

An analysis of worst-case losses is crucial for practical implementation because it needs to be ensured that all liabilities can be properly resolved. A common way to ensure that all parties can serve their liabilities is through depositing escrows which can cover the worst-case scenario. A further purpose is to understand how liabilities are distributed between market creator and participants.

### 2.4.1 Worst-case Loss for Participants

Consider a forecaster in a scoring rule based decision market who reports ${}^*\vec{r_k}$ when the current prediction is $\vec{r_k}$ for each conditional market $k$. The worst-case loss for this report is given by:

$$\begin{aligned} & \min_{j,i} \left( S_i^j({}^*\vec{r_j}) - S_i^j(\vec{r_j}) \right) \\ = \quad & \min_{j,i} \frac{1}{\phi_j} \left( s_i^j({}^*\vec{r_j}) - s_i^j(\vec{r_j}) \right) \end{aligned} \tag{2.10}$$

From equation (2.10) we can tell that the worst-case loss for the forecaster depends on both the decision rule $\phi_j$ and the report ${}^*\vec{r_j}$ he/she made. The probability $\phi_j$ depends on the decision rule. Small probabilities in the decision rule, which may be in the interest of market creator to approximate deterministic scoring rules, increase the worst-case loss for the forecaster.

A trader in a security based decision market purchase securities from $\vec{q_k}$ to ${}^*\vec{q_k}$. Assuming again that forecasters cannot hold negative positions, the worst-case loss for the trader can be calculated as:

$$\sum_{k=1}^{m} \Big( C_k(\vec{q_k}) - C_k({}^*\vec{q_k}) \Big) \tag{2.11}$$

Equation (2.11) shows that the worst-case loss for a trader in the securities based decision market only depends on the cost the trader spent. In other words, the trader in the securities based market will not be exposed to any liabilities beyond the costs already paid for purchasing the assets. Therefore a securities based implementation has the advantage that it does not need to further track the liabilities on the side of the traders. Moreover, the worst-case loss of trader does not depend on the decision rule.

### 2.4.2 Worst-case Loss for Market Creator

The loss of a market creator mirrors the profits gained by the participants. Apart from the distribution of realised payoffs for the participants, there is therefore a difference of the worst-case loss for the market creators between a scoring rule based decision market and the corresponding securities based decision market.

Carrying over the conditions from equation (2.3), the worst-case loss for a market creator of a scoring rule based decision market is

$$\begin{aligned} & \min_{j,i} \Big( S_i^j(\vec{r_j}) - S_i^j({}^*\vec{r_j}) \Big) \\ = \quad & \min_{j,i} \frac{1}{\phi_j} \Big( s_i^j(\vec{r_j}) - s_i^j({}^*\vec{r_j}) \Big) \end{aligned} \tag{2.12}$$

As we can tell from equation (2.12), the worst-case loss for a market creator depends on three factors: initial report $\vec{r_j}$ and final report ${}^*\vec{r_j}$ for the selected conditional market and the decision rule $\phi_j$. Among three factors, the market creator has control over the initial report $\vec{r_j}$ and the value that decision rule $\phi_j$ can take, but does not have control over which action is being picked. Even though the decision rule can be arbitrary as long as forecasters are convinced that it has full support, it is the interest of the market creator to take the final forecasts of the market into account. For instance, it does not fit the interest of the market creator to assign a small probability to the action that is predicted to most likely lead to a desirable outcome. The relationship between decision rule $\phi_j$ and the final score $s_i^j({}^*\vec{r_j})$ can be complex in that it depends on how exactly the final forecast determines

the decision rule $\vec{\phi}$. There is a suggestion about computing a minimal feasible decision rule for each action according to the budget of market creator [1].

Using the conditions for equation (2.7), the worst-case loss for a market creator of a securities based decision market is:

$$\sum_{k=1}^{m}\left(C_k({}^*\vec{q_k}) - C_k(\vec{q_k})\right) - \max_{i,j}\left(\frac{1}{\phi_j}\left({}^*q_i^j - q_i^j\right)\right) \tag{2.13}$$

In equation (2.13), the term $\sum_{k=1}^{m}(C_k({}^*\vec{q_k}) - C_k(\vec{q_k})))$ is the income from securities sales for market creator, which mirrors the cost spent by participants in order to move the inventory distribution from $\vec{q_k}$ to ${}^*\vec{q_k}$ in each conditional market $k$. The second term, $\max_{i,j}(\frac{1}{\phi_j}({}^*q_i^j - q_i^j))$ in the equation is the maximal payout that can be won by participants. In order to compare the worst-case losses, we substitute the equation (2.1) into the equation (2.13) and obtain:

$$\begin{aligned}
&\sum_{k=1}^{m}\left(C_k({}^*\vec{q_k}) - C_k(\vec{q_k})\right) - \max_{i,j}\left(\frac{1}{\phi_j}\left({}^*q_i^j - q_i^j\right)\right) \\
= \quad &\underbrace{\sum_{k=1,k\neq j}^{m}\left(C_k({}^*\vec{q_k}) - C_k(\vec{q_k})\right)}_{\text{a}} - \max_{i,j}\left(\underbrace{s_i^j({}^*\vec{r_j}) - s_i^j(\vec{r_j})}_{\text{b}} - \underbrace{\frac{(1-\phi_j)}{\phi_j}\left({}^*q_i^j - q_i^j\right)}_{\text{c}}\right)
\end{aligned} \tag{2.14}$$

Term a of equation (2.14) is non-negative, and depends on the sales in all conditional markets except for the one representing the selected action. Term b is the scoring rule that corresponds to our cost function and is bounded. However, term c depends on final outstanding securities ${}^*q_i^j$ and $(1-\phi_j)/\phi_j$. While the market creator has control over $\phi_j$, the final outstanding securities is not known ex ante. Therefore no finite initial escrow can guarantee to cover the market creator's liability. This loss of a bound on the worst-case loss of a market maker differs from the the loss of a bound from low probabilities in the decision rule as described in [1]. The final budget for a market maker in a decision market does not have an upper limit because traders can buy arbitrarily large numbers of shares $q_i^j$ on the selected action, while buying fewer (or no) shares on the other actions. Note that it is in the interest for the market creator to assign a small probability $\phi_j$ to actions that are not preferred. The term $(1-\phi_j)/\phi_j$ increases rapidly as $\phi_j$ approaches zero. In summary, the advantage of a worst-case loss for the participants that does not depend on the decision rule thus comes at the disadvantage that the worst case loss for the market creator cannot be known ex ante.

### 2.4.3 Numeric Example for Worst-case Losses

Let us consider a numeric example that a market creator has two actions and each action has two outcomes. The market creator uses simple logarithmic scoring rule with $s_j({}^*r_j) = log({}^*r_j)$ for both actions. The corresponding cost function is $C_j({}^*\vec{q}_j) = log((e^{q_1} + e^{q_2})/2)$. The markets start with an initial report of $r_1^j = r_2^j = 0.5$ and $q_1^j = q_2^j = 0$ for each action $\alpha_j$. A forecaster reports as shown in the table 2.3.

Table 2.3: The reports and the corresponding securities.

| | **Report** ${}^*\vec{r}_j$ | | **Securities** ${}^*\vec{q}_j$ | |
|---|---|---|---|---|
| | $\omega_1$ | $\omega_2$ | $\omega_1$ | $\omega_2$ |
| $\alpha_1$ | 0.88 | 0.12 | 2 | 0 |
| $\alpha_2$ | 0.27 | 0.73 | 0 | 1 |

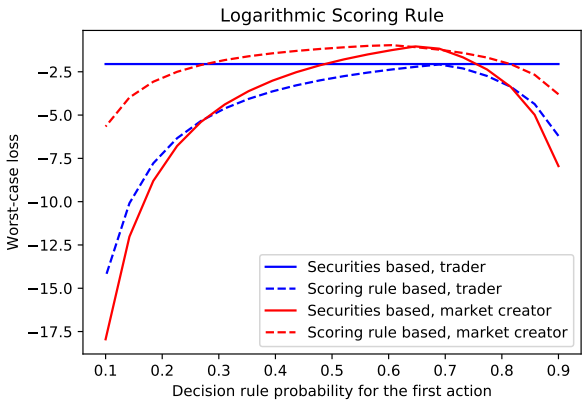


Figure 2.1: Worst-case loss comparison between the trader and the market creator in different decision markets with a varying probabilities in the decision rule.

Figure 2.1 shows the worst-case loss for the trader and the market creator under a securities based and scoring rule based mechanism in dependence of the decision rule. A trader in the securities based mechanism faces a worst-case loss that does not depend on the decision rule. In the scoring rule based market, his/her risk is higher and increases rapidly if the decision rule approaches 0 or 1. For the values used here, the market creator in the scoring rule based decision market faces less risk than the trader but the risk is still dependent on the decision rule probability. In securities based market, the market creator, however, faces larger risks. The reduction of the worst-case loss of the forecaster comes with an increased worst-case loss for the market creator. This feature of securities based decision markets is preferred by market creators who are willing to sacrifice financial efficiency for more liquidity.

## 2.5 Re-allocation of worst case losses between market creator and participants

Compared to scoring rule based decision markets, securities based decision markets offer additional flexibility to shape the distribution of realized payoffs. This flexibility arises because in a securities based decision market, each report ${}^*r$ can be realized through infinitely many trades. As outlined in equation (2.2), in prediction markets, if a trader purchases $\beta$ security for each outcome, the cost will be $\beta$. The prices, i.e. the current forecast for the probability distribution over the outcomes, remains the same. The payout from these additional securities will be exact $\beta$ regardless of outcomes, and the net realised payoff for such a trade will be zero.

In a securities based decision market this is, however, not the case. Assume a trader in a securities based decision market purchases a number of $\beta_k$ of each outcome in conditional market $k$, for any $k$. We refer to such a trade as purchasing a 'bundle' of securities. Let the market creator select action $\alpha_j$. Regardless of which outcome is observed, the realised payoff of the trader from these trades can be obtained as:

$$\frac{1}{\phi_j}\beta_j - \sum_{k=1}^{m}\beta_k \tag{2.15}$$

While the prices for each outcome in all conditional markets remains the same, the realised payoff for the trader in a decision market is affected by these trades and depends on which action is selected.

This property allows a trader to adjust the distribution of realised payoffs through purchasing bundles of securities without changing the reported probability distributions. Purchasing the same number of $\beta_k$ of each outcome in conditional market $k$ can also be viewed as the trader purchasing a 'lottery' ticket that costs $\beta_k$ and returns $\beta_k/\phi_k$ at a probability of $\phi_k$. The realised payoff of a trader in a securities based decision market can be rewritten as:

$$\frac{1}{\phi_j}({}^*q_i^j - q_i^j) - \sum_{k=1}^{m}(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)) + \frac{1}{\phi_j}\beta_j - \sum_{k=1}^{m}\beta_k \tag{2.16}$$

For each specific report there is a manifold of trading strategies that link to it each of which leading to a different distribution of payoffs.

### 2.5.1 Standardised trades

For further discussion, we introduce a standardized trade as reference point in the trading strategy space. We define a standardized trade for changing reports from $\vec{r}_k$ from ${}^*\vec{r}_k$ as the trade with the

smallest possible non-negative elements in the number of purchased shares for any $k$. The standardised trade can be obtained from any arbitrary trade $({}^*\vec{q}_k - \vec{q}_k)$ as:

$$({}^*\vec{q}_k - \vec{q}_k) - \min_i({}^*q_i^k - q_i^k) \times \vec{1}_k \qquad \forall k$$

An example is shown in table 2.4. An arbitrary and the standardised trade lead to the identical report. A standardised trade is the least costly way to make a report without shorting in a securities based decision market. As for any trade that does not involve short positions, traders will have no liabilities once they have paid the cost for purchasing securities. On the other hand, the market creator has a greater liability to resolve the outstanding securities after the action is selected and the outcome is realised.

Table 2.4: Both the arbitrary trade and the standardised trade lead to the same report but result in different realised payoffs.

| | **Arbitrary Trades** | | **Standardised Trades** | |
|---|---|---|---|---|
| | $\omega_1$ | $\omega_2$ | $\omega_1$ | $\omega_2$ |
| $\alpha_1$ | 3 | 1 | 2 | 0 |
| $\alpha_2$ | 1 | 2 | 0 | 1 |

### 2.5.2 Approximating the payoffs of scoring rule based decision markets

The flexibility to shape the distribution of realised payoffs can be used to design trades such that the payoffs in securities based decision markets match exactly those in scoring rule based markets. However, this requires the traders to accept negative positions, i.e. to short sell securities, and re-introduces two-sided liabilities.

Let $\beta_k$ in equation (2.16) to be substituted by $-\big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big)$ for all $k$. The realised payoff for such a trader can be obtained by:

$$\begin{aligned} & \frac{1}{\phi_j}({}^*q_i^j - q_i^j) - \sum_{k=1}^{m}\big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big) - \frac{1}{\phi_j}\big(C_j({}^*\vec{q}_j) - C_j(\vec{q}_j)\big) \\ & + \sum_{k=1}^{m}\big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big) \\ = \; & \frac{1}{\phi_j}\Big(({}^*q_i^j - q_i^j) - \big(C_j({}^*\vec{q}_j) - C_j(\vec{q}_j)\big)\Big) \\ = \; & \frac{1}{\phi_j}\big({}^*s_i^j(\vec{r}_j) - s_i^j(\vec{r}_j)\big) \end{aligned} \tag{2.17}$$

With such a trade, a trader makes a report through longing securities that she/he believes are

under-priced and shorting securities on over-priced outcomes to meet the cost. The net cost for such a trade is zero. An example for the numbers of securities exchanged under such a trading strategy is shown in Table 2.5.

Table 2.5: Traders can design their trades to achieve the same realised payoffs as the corresponding scoring rule based decision market. However, this requires short selling.

| | **Standardised Trades** | | **Scoring Rule** | |
|---|---|---|---|---|
| | $\omega_1$ | $\omega_2$ | $\omega_1$ | $\omega_2$ |
| $\alpha_1$ | 2 | 0 | $2 - \ln(e^2+1)/2$ | $-\ln(e^2+1)/2$ |
| $\alpha_2$ | 0 | 1 | $-\ln(e+1)/2$ | $1 - \ln(e+1)/2$ |

### 2.5.3 Market Creator Liability Free Decision Markets

We will conclude this discussion a with a design that allocates liabilities entirely to the side of traders. Let the $\beta_k = \max_i({}^*q_i^k - q_i^k)$ in the conditional market which represents the action $\alpha_k$, we obtain

$$({}^*\vec{q}_k - \vec{q}_k) - \max_i({}^*q_i^k - q_i^k) \times \vec{1}_k \qquad \forall k$$

The realised payoff from the market creator point of view can be obtained by:

$$\begin{aligned}
& \sum_{k=1}^{m} \left(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\right) - \sum_{k=1}^{m} \beta_k - \frac{1}{\phi_j}\left(({}^*q_i^j - q_i^j) - \beta_j\right) \\
= \quad & \sum_{k=1}^{m} \left(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\right) - \sum_{k=1}^{m} \max_x({}^*q_x^k - q_x^k) - \frac{1}{\phi_j}\left(({}^*q_i^j - q_i^j) - \max_x({}^*q_x^j - q_x^j)\right) \\
= \quad & \frac{1}{\phi_j} \underbrace{\left(\max_x({}^*q_x^j - q_x^j) - ({}^*q_i^j - q_i^j)\right)}_{\text{Non-negative}} - \underbrace{\sum_{k=1}^{m}\left(\max_x\left(s_x^k({}^*\vec{r}_k) - s_x^k(\vec{r}_k)\right)\right)}_{\text{Known ex ante}}
\end{aligned} \tag{2.18}$$

As demonstrated by equation (2.18), the term that depends on decision rules and quantities of outstanding securities is guaranteed to be non-negative. As a result, the liability of the market creator is guaranteed to be covered if the market creator can afford $\sum_{k=1}^{m} \max_x \left(s_x^k({}^*\vec{r}_k) - s_x^k(\vec{r}_k)\right)$ which is known ex ante.

Table 2.6: In this market, traders report probability distribution through short selling completely.

| | **Standardised Trades** | | **Liability-free** | |
|---|---|---|---|---|
| | $\omega_1$ | $\omega_2$ | $\omega_1$ | $\omega_2$ |
| $\alpha_1$ | 2 | 0 | 0 | $-2$ |
| $\alpha_2$ | 0 | 1 | $-1$ | 0 |

In this setting, the market creator essentially purchases securities from the forecasters. The market

creator covers the worst case loss by paying for the securities.

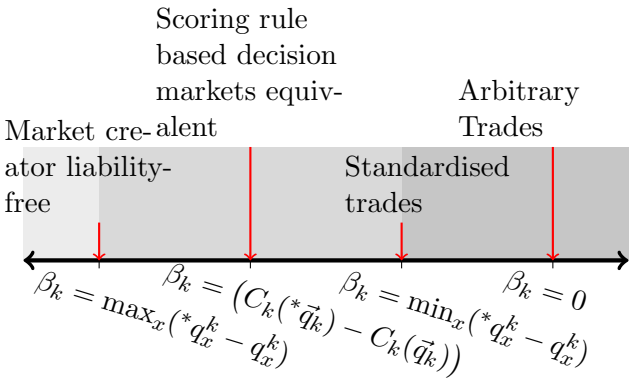


Figure 2.2: The liability is shifted with different choice of $\beta_k$.

Figure 2.2 provides a graphical summary of the trading strategies discussed above. In the dark gray area on the right side of the figure, traders cover their worst-case losses when paying for the purchased securities. The worst-case loss for the market creator can become arbitrarily high. In contrast, in the left, light gray part of the graph where traders engage solely in short selling, the market creator will cover their worst-case loss when paying for the securities purchased from the forecasters, and the traders will have liabilities to the market creator. The scoring rule based decision market sits in between where the market creator and forecasters both have liabilities.

### 2.5.4 External Insurers

In previous sections, we use the trading of bundles $\beta_k$ in a conditional market $k$ to better understand how liabilities are allocated in different mechanisms. Consider we separate traders into two kinds: regular traders report in standardised trading (thus change prices) with long position only; special traders trade 'bundles' of securities (which does not change prices) with short position only. These special traders can be considered as insurers. In this case, the liability can be separated from both regular traders and the market creator. Thus the stochastic decision rule can further approximate deterministic decision rules with a predictable worst-case loss for the market creator.

If an insurer accepts short positions at an amount equal to the costs of all regular traders' outstanding securities $\left(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\right)$ for each conditional market $k$, the realised worst-case loss of

the market creator can be obtained by:

$$
\begin{aligned}
& \sum_{k=1}^{m} \big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big) - \frac{1}{\phi_j}({}^*q_i^j - q_i^j) \\
& + \frac{1}{\phi_j}\big(C_j({}^*\vec{q}_j) - C_j(\vec{q}_j)\big) - \sum_{k=1}^{m} \big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big) \\
= \quad & \frac{1}{\phi_j}\Big(\big(C_j({}^*\vec{q}_j) - C_j(\vec{q}_j)\big) - ({}^*q_i^j - q_i^j)\Big) \\
= \quad & \frac{1}{\phi_j}\big(s_i^j(\vec{r}_j) -^* s_i^j(\vec{r}_j)\big)
\end{aligned}
$$

The realised payoff will be identical to the scoring rule that the cost function is derived from. If we change the insurers' shorting position to $\max_x \left({}^*q_x^k - q_x^k\right)$ to each conditional market $k$ and substitute equation (2.1) into the market creator's realised payoff , then we will have:

$$
\begin{aligned}
& \sum_{k=1}^{m} \big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big) - \frac{1}{\phi_j}({}^*q_i^j - q_i^j) + \frac{1}{\phi_j}\max_x \left({}^*q_x^k - q_x^k\right) - \sum_{k=1}^{m} \max_x \left({}^*q_x^k - q_x^k\right) \\
= \quad & \sum_{k=1}^{m} \big(C_k({}^*\vec{q}_k) - C_k(\vec{q}_k)\big) - \sum_{k=1}^{m} \max_x({}^*q_x^k - q_x^k) - \frac{1}{\phi_j}\Big(({}^*q_i^j - q_i^j) - \max_x({}^*q_x^j - q_x^j)\Big) \\
= \quad & \frac{1}{\phi_j} \underbrace{\Big(\max_x({}^*q_x^j - q_x^j) - ({}^*q_i^j - q_i^j)\Big)}_{\text{Non-negative}} - \underbrace{\sum_{k=1}^{m} \Big(\max_x \big(s_x^k({}^*\vec{r}_k) - s_x^k(\vec{r}_k)\big)\Big)}_{\text{Known ex ante}}
\end{aligned}
$$

The insurer essentially offers a lottery. For real-world applications it might not be realistic to assume that this is done without additional costs. However, paying a small additional fee to such an insurer is might be in the interest of the market creator to reduce the worst-case loss, which in turn will also to better approximate deterministic decision rules.

## 2.6 Conclusion and Discussion

We introduce a setting for securities based decision markets that can be conveniently deployed in practical applications. In such a setting, a trader will report a forecast through trading securities. For the securities that represent the selected action and observed outcome, a trader will receive $1/\phi_j$ payoff per share, where $\phi_j$ is the probability in the decision rule corresponding to the selected action. Other shares pay zero, including those purchased in the unselected conditional markets. We prove that under the same condition, specifically, the same action space and the same outcome space, a securities based decision market in our setting has the same expected payoff for participants as the

corresponding scoring rule based decision market.

We compare a securities based decision market under our setting and the corresponding scoring rule based decision market in terms of realised payoffs for participants. The comparison demonstrates that the difference depends on how much the participants report or trade in the selected conditional market. We notably find that the forecaster in scoring rule based decision market will have no cost for reporting in unselected conditional markets while this is not the case in the securities based decision market. We further show that with an additional 'lottery', a forecaster in scoring rule based decision market can have the identical realised payoffs as a trader in the corresponding securities based decision market under the same condition. Similarly, the realised payoffs of a trader in a securities based decision market can recover the realised payoffs of the corresponding scoring rule based market, but this requires the forecasters to 'short-sell' securities, i.e. to hold negative positions.

By being equivalent to a scoring rule based decision market with an additional zero-mean 'lottery', the securities based mechanism described here offers an additional set of parameters that allow to shape the distribution of payoffs beyond what can be achieved based on scoring rules alone. This allows to re-allocate liabilities and worst-case losses between forecasters and the market creator. As illustrated in section 2.4, in a market where forecasters only purchase positive positions (no short selling), their liabilities are covered when paying for the purchased securities. Moreover, in contrast to scoring rule based decision markets, their worst-case losses do not depend on the probabilities used in the decision rule. A securities based decision market design might thus be of advantage for a market creator who aims to attract forecasters who are concerned about limiting their worst-case losses. In Section 2.5, we show that the creator risks can be further mitigated by external insurers, which allows to closer approximate deterministic decision rules. Further empirical studies will be of value in determining how to shape trading to obtain the most accurate forecasts.

# Chapter 3

# Decision Market Based Learning for Multi-agent Contextual Bandit Problems

Chapter 3 mainly comprises the paper titled '*Decision market based learning for multi-agent contextual bandit problems*' preprinted in `https://arxiv.org/abs/2212.00271`. In Chapter 2, I discuss the use of decision markets with human traders, whereas this chapter investigates the adoption of decision markets to solve a multi-agent learning problem with computational agents. Section 3.2.1 and 3.2.2 serve as a literature review for multi-agent systems and bandit problems, respectively. Section 3.2.3 overlaps with 2.2.

## Summary of Notation

| | |
|---|---|
| $k$ | number of available actions |
| $m$ | number of agents |
| $T$ | time step variable, $T \in \{1, 2, \ldots, n\}$ |
| $A$ | action variable, $A \in \{1, 2, \ldots, k\}$ |
| $\Omega_{(A)}$ | outcome of action $A$ (Bernoulli variable) |
| $E$ | agent, $E \in \{1, 2, \ldots, m\}$ |
| $Pr_{(E)}$ | probabilistic reports from agent $E$ over all actions, $Pr_{(E)} \in [0, 1]^k$ |
| $Pr_{(E,A)}$ | probabilistic reports from agent $E$ about action $A$ |

| | |
|---|---|
| $Pr_{(0)}$ | initial prior probabilistic reports |
| $Pr_{(m)}$ | the final aggregated probabilistic reports |
| $C_{(E)}$ | contextual information for agent $E$ |
| $\phi(Pr_{(m)})$ | decision rule, i.e., a probability distribution of the available actions, depending on final report $Pr_{(m)}$ |
| $\hat{S}\left(Pr_{(E)}, \Omega_{(A)}\right)$ | scoring rule that quantifies the accuracy of $Pr_{(E)}$ once $\Omega_{(A)}$ materialises |
| $S(Pr_{(E)})$ | decision scoring rule that quantifies the accuracy of report $Pr_{(E)}$ given that the selected action is $A$ and the outcome is $\Omega_{(A)}$; it scales up the score by $1/\phi_A(Pr_{(m)})$ |
| $\Theta_{(E)}$ | learning parameters matrix of agent $E$ |
| $\mu(C_{(E)}, Pr_{(E-1)}, \Theta_{(E)})$ | parameters used to sample the log-odds reports from agent $E$ |
| $H_{(E)}$ | log-odds reports |
| $G_{(E,T)}$ | a matrix of approximated partial derivatives of the expected score of expert $E$ with respect to policy parameters |
| $B(C_{(E,T)})$ | baseline function that does not vary with action $A$ but only depends on $C_{(E,T)}$ |
| $F$ | number of pieces of experience within one 'mini-batch' used for learning |

## Abstract

Information is often stored in a distributed and proprietary form, and agents who own this information are often self-interested and require incentives to reveal it. Suitable mechanisms are required to elicit and aggregate such distributed information for decision-making. In this study, we use simulations to investigate the use of decision markets as mechanisms in a multi-agent learning system to aggregate distributed information for decision-making in a contextual bandit problem. The system utilises strictly proper decision scoring rules to assess the accuracy of probabilistic reports from agents, enabling them to learn to solve the contextual bandit problem jointly. Our simulations show that our multi-agent system with distributed information can be trained as efficiently as a centralised counterpart with a single agent that receives all information. Moreover, we use our system to investigate scenarios with deterministic decision scoring rules which are not incentive compatible. We observe the emergence of more complex dynamics with manipulative behaviour, which agrees with existing theoretical analyses.

## 3.1 Introduction

In many decision-making tasks, the relevant information is distributed over multiple parties. To optimise decision-making, multi-agent learning systems are required to obtain, aggregate and learn from such distributed information. When the agents' information is private and the objective is self-interested, rewards may be required to induce the agents to reveal their information. For efficient multi-agent learning in such a situation, the rewards must be designed so that as agents maximise their rewards in the training phase, the system's overall performance is also optimised.

Consider, for example, a recommendation system that aims to optimise advertisement targeting by using information from multiple sources (e.g., Google, Facebook and Amazon). Such information could involve the companies' different user profile data for the targeted person, which the companies have no interest to reveal. The system, therefore, needs to elicit information in a form that is agreeable to the information source (e.g. recommendations for the task at hand, rather than complete user profiles) and needs to provide fair rewards for these contributions. Such rewards can be monetary but need to be designed such that each information source can learn from the realised rewards and while maximising its rewards, the performance of the recommendation system improves as well.

In this work, we develop a multi-agent learning system that provides agents with rewards that align the agents' objectives with the system's objectives. We test the system in simulations of learning in a multi-armed Bandit problem where contextual information is distributed over multiple agents. Our approach is based on decision markets, which are an extension of prediction markets. While prediction markets are mechanisms of multi-agent forecasting, decision markets are mechanisms of multi-agent decision-making where decisions are made based on forecasts. The contextual bandit problem we study in the simulations can be seen as a one-step reinforcement learning problem.

This paper is organised as follows. In Section 3.2, we discuss relevant work from three related topics: multi-agent learning (Section 3.2.1), bandit problems (Section 3.2.2) and decision markets (Section 3.2.3). In Section 3.3, we introduce our research methodology. In Section 3.4, we present simulation results, and in Section 3.5, we discuss future work directions.

Our results show that the decision market based multi-agent bandit system can optimise decision-making without requiring individual agents to share their contextual information directly. We use our system to examine the agents' behaviour in a decision market with a stochastic decision rule which provides proper incentives for accurate reporting by the agents, and in a decision market with a deterministic decision rule which can be manipulated and exploited by the agents. We find that

under a stochastic decision rule, agents quickly learn to provide accurate reports. Under the deterministic decision rule, we observe interesting manipulative interactions that nevertheless often result in surprisingly good decision-making.

## 3.2 Related Work

### 3.2.1 Multi-agent Learning

Multi-agent learning is an essential and rapidly growing research area in computer science field. According to the nature of the interactions between the agents, multi-agent learning algorithms can be grouped into three categories [33]: purely cooperative, purely competitive, and mixed.

An example of a cooperative task is the wolf-pack game where two wolves chase prey. Because the prey is faster than the wolves, the wolves need to learn a cooperative strategy to capture the prey [34]. When either one wolf reaches the prey, the task is solved and the entire wolf-pack receives a reward. In such a cooperative task, agents usually share an identical reward function and thus learn to maximise the joint rewards of the system.

In competitive tasks, one agent's gain is the other agent's loss. In the wolf-pack game, for instance, the wolf agents and the prey agent are in a competitive game. This kind of game is well-studied in game theory. Optimal strategies can be learned with the min-max learning paradigm, which is guaranteed to find the best policy under the worst-case assumption of the opponent's move when the policy searching space is manageable [35]. Recent attention has focused on deep reinforcement learning to solve complex competitive games such as Go [36], [37].

Mixed tasks contain elements of the two extremes described above and the nature of games and agents can be very diverse. For instance, some tasks involve two teams playing against each other [38], and cooperation and competition co-exist in these tasks. Some studies focus on the dynamics of self-interested agents' interaction in such tasks [39]. In our work, the agents engage in a partial information game that is neither purely cooperative nor purely competitive. It is not purely cooperative, because the agents have individual reward functions and thus can be seen as self-interested, and they are not purely competitive because the overall reward provided to the agents increases if the agents work well together. To draw an analogy with the previous wolf-pack game, our game is essentially a wolf-crow game where a group of crows has information about the location of numerous prey that the wolf would benefit from obtaining. The wolf needs to reward the crows for their information, such that they learn to guide the wolf to the most valuable prey.

A further classification of multi-agent learning exists along the training methods dimension. A significant share of attention focuses on a paradigm called centralised training decentralised execution [34], [40]. On the other hand, some algorithms fall into the fully decentralised paradigm.

Centralised training with decentralised execution is suitable for agents that execute actions locally, based on local information. During the training phase, however, all the local information is accessible to a centralised 'critic' that can evaluate the agent's actions from a higher level. This paradigm can ameliorate two problems in multi-agent learning: the 'coordination problem' and the 'non-stationary issue' [33]. The coordination problem arises when agents have to 'match' or coordinate their actions to maximise rewards. The non-stationary issue arises when agents optimise in an environment that is non-stationary because it contains other co-optimising agents [34], [40]. This paradigm, however, is not suitable for our task, because the local information is private and is not readily shared.

Federated learning is a novel paradigm that aims to solve a centralised learning problem without compromising the privacy of users. The federated bandit problem [41], [42] provides an important framework for recommendation systems without central access to local data [43] and sometimes even without a centralised model [44].

Another approach is to let agents learn local policies independently [45], [46]. This approach is suitable when information is private, but learning can be affected by the non-stationary problem. In our design, we use a mechanism from economics to decorrelate the relationship between the reward for an independent agent and the peers' actions and therefore expect to mitigate the non-stationary problem.

### 3.2.2 Bandit Problems

Bandit problems provide a framework for studying optimal decision-making when several alternative actions are available that yield rewards from an unknown, stationary distribution. Actions can be discrete or continuous [47]. The rewards essentially quantify the quality of the selected action. Agents act repeatedly and learn by taking a history of the rewards into account for decision-making in subsequent rounds. Agents need to balance the exploration for potential better action and the exploitation according to the best knowledge so far [9]. For some problems, 'hints' or contexts exist and provide information about the reward distribution associated with an action. The reward distribution is non-stationary and hints change accordingly. The bandit problem, therefore, extends to a contextual bandit problem which is equivalent to a one-step reinforcement learning problem [48].

Contextual bandit problems are well-suited models for decision-making challenges based on current

and past information. Many practical applications, such as product recommendations and advertisement targeting, are based on bandit learning and categorised into recommendation systems. An important use case of multi-agent bandit learning is cognitive radio. In cognitive radio, users want to identify idle channels intelligently, which is often discussed along with a multi-agent environment where two users want to avoid selecting the same idle channel [49]–[51]. Recommendation systems are predominately investigated with centralised models, but with increasing regulation of data privacy, security and access right, decentralised models that can utilise the private data on individual devices are gaining relevance. In the system we are investigating, the agents learn locally from local contextual data to help optimise decision-making in a multi-agent multi-armed bandit problem.

### 3.2.3 Decision Markets

Collective decision-making with distributed information is a familiar challenge in economics. In decision markets, this challenge is addressed by eliciting and assessing forecasts about the consequences of the available actions. Specifically, a principal (decision maker) elicits forecasts from agents with access to relevant information and then selects an action according to these forecasts. After execution, the principal will compute scores for the agents' contributions.

Scoring rules provide such an assessment of forecasts by assigning a numerical score to forecasts depending on the realised outcome [25]. A proper score can guarantee the highest expected return if the evaluated forecast aligns with the actual belief of the forecaster. In other words, a rational agent maximising its expected score will under a proper scoring rule report the most accurate forecast it can make. Proper scoring rules are suited to reward single agents for their forecasts and allow them to learn to make forecasts more accurate.

While proper scoring rules provide proper incentives for single agents to make accurate forecasts, properly incentivised prediction markets are mechanisms to elicit and aggregate forecasts from multiple agents that have access to different pieces of information. The mechanisms of aggregation depend on the implementation of prediction markets. Hanson proposes a proper prediction market mechanism that allows agents to make direct probabilistic reports in a sequential order [2], and suggests market scoring rules to price assets such that sequential reporting and trading in an asset market becomes equivalent. Such a mechanism requires agents to make Bayesian updates based on the report from the previous agent. Chen and Pennock further generalise market scoring rules and their relation to scoring rules [7]. Implementations following this approach require a principal to provide liquidity for a prediction market by an automated market maker algorithm that is always available to trade.

While scoring rules assess single forecasts, and prediction markets provide a mechanism for aggregating the forecasts from multiple agents, decision markets extend these approaches to evaluate and aggregate forecasts for decision-making [5]. The challenge is to quantify the quality of forecasts that are conditional on the actions. Because the realised future depends on the selected action, it is difficult to assess forecasts about the other actions. Decision rules, which choose actions according to multiple forecasts, are the core of the solution to this challenge. A naive way is to use a deterministic decision rule that always selects the best action according to the forecasts. However, this approach may incentivise misleading forecasts since the forecasts directly determine the selected action [6], [13]. Chen, Kash, Ruberry, *et al.* propose a stochastic decision rule that breaks this relation, and thus makes the forecasts reliable [1], [26]. Wang and Pfeiffer extend the mechanism from direct probabilistic forecasts to equivalent asset trading markets [8].

## 3.3 Algorithm

### 3.3.1 Problem Setup

We study a multi-agent multi-armed contextual Bernoulli bandit problem, where one agent (referred to as the principal) decides between multiple alternative actions and receives a corresponding reward that evaluates the quality of the decision. The context, however, is distributed over multiple self-interested agents. In the system we investigate here, the principal uses a decision market to sequentially elicit probabilistic reports for the Bernoulli outcomes of the available actions from the agents (see Figure 3.1). In each time step, the principal receives an initial set of prior probability distributions for the outcomes of each action. It then selects an agent to alter this report. The agent will be scored for this altered report using a decision scoring rule. The principal then adopts this report and selects the next agent to alter it, and this process is repeated until the last agent has been selected. Once all agents have been queried, the principal uses the final report (from the last agent) and a decision rule to select an action. When the selected action is executed and the outcome is observed, the scores for all agents can be calculated, and the time step concludes.

Note that while the principal faces a contextual Bernoulli bandit problem, every other agent faces a continuous contextual bandit problem, where the agent's action is its probabilistic report to the principal (see Figure 3.1). To clearly distinguish between these two contextual bandit problems and the, we refer to the context of the Bernoulli bandit problem as the system's context, and the context in the continuous bandit problem of the individual agents as the agent's context. The agent's

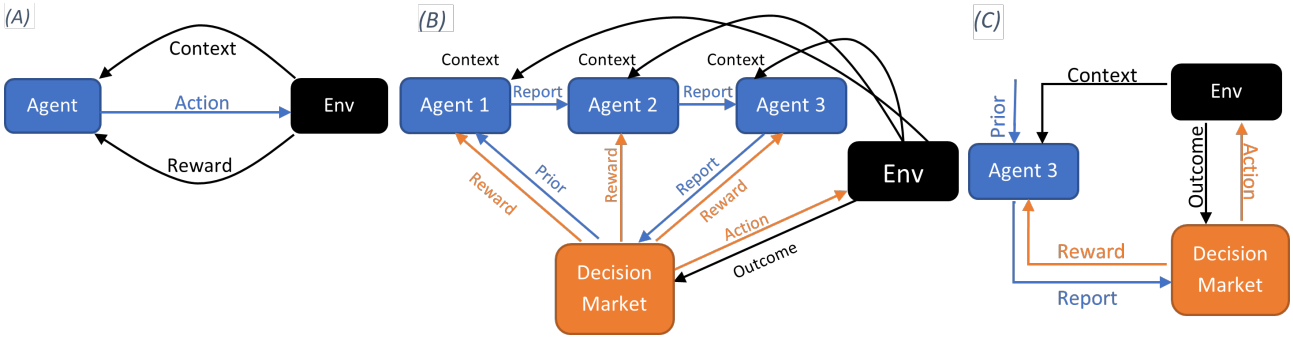


Figure 3.1: **Decision markets based multi-agent bandit system**. Panel (A) shows a diagram for a regular contextual bandit problem. An agent can choose an action after receiving contextual information. The action results in a reward from the environment. Panel (B) shows a multi-agent contextual bandit problem with a decision market, which is the main design of this paper. An action is selected by a decision market, which aggregates distributed posterior probabilities reported from agents. The decision market assigns a reward to each agent based on the quality of their reports. Panel (C) shows a contextual bandit problem with a continuum-arm space in agent 3's perspective.

context consists of the signals it receives from the system's environment, and the previous report it receives from the principal or the previous agent. The system's context consists of all signals that are received by the agents from the environment, including the priors that the principal receives from the environment. The principal in this system cannot learn. However, the agents can learn to use the context to generate reports that maximise the score they receive. We test if, in such a system, the agents can efficiently learn such that the principal's performance in the Bernoulli bandit problem improves.

In the following, we provide the notation and properties of the Bernoulli bandit problem, the agents and the context they receive, the principal's decision rule and scoring rule, and the agents' learning algorithm.

### 3.3.2 Bernoulli Bandit Problem

We denote the time step as $T \in \{1, 2, \ldots, n\}$. We assume that in each time step the principal selects one action from a finite, discrete set of action $A \in \{1,\ 2, \ldots, k\}$. The outcome $\Omega_{(A)} \in \{0, 1\}$ of action $A$ is a Bernoulli variable. We assume $\Omega_{(A)} = 1$ is the outcome desired by the principal. Selecting one action and observing the outcome will not reveal any information about the outcomes of the other actions.

We consider $m$ agents, which are denoted as $E \in \{1, 2, \ldots, m\}$. For each time step $T$, agent $E$ will privately receive a signal, which provides information about the outcome of the principal's available actions. The agent also receives a probabilistic report $Pr_{(E-1)} \in [0, 1]^k$ from the principal when

$E = 1$ or the previous agent. The agent's context denoted by $C_{(E)}$ consists of the signals and the report from the principal or the previous agent and is used by the agent to return a report $Pr_{(E)}$. The probabilistic report about any specific action A is denoted as $Pr_{(E,A)}$.

### 3.3.3 Principal, Decision Rule and Scoring Rule

At the beginning of each time step $T$, the principal starts with an initial vector of probabilities $Pr_{(0)}$. This can be seen as prior probabilities that are provided from the environment to the principal, which is compatible with a common prior assumption that is often used in related work [4]. The principal passes this initial report to the first agent, and the first agent returns an updated report denoted as $Pr_{(1)}$. The principal passes the updated vector to the next agent and repeats this procedure until the final report $Pr_{(m)}$ is received from the last agent $m$. After enquiring with all the agents, the principal uses the final report and a decision rule to select an action. We define a decision rule as a function which maps the final reports to a probability distribution over the available actions:

$$\Phi : Pr_{(m)} \rightarrow \Delta\left(\{A\}\right) \tag{3.1}$$

We denote $\Phi_A\left(Pr_{(m)}\right)$ as the probability of action $A$ to be selected. The principal will sample an action from the distribution $A \sim \Phi\left(Pr_{(m)}\right)$ and execute it. Afterwards, the principal observes the outcome $\Omega_{(A)}$ for the executed action and receives the corresponding reward. The principal will score the report of agent $E$ using a decision score function:

$$S : Pr_{(E)} \times \Phi\left(Pr_{(m)}\right) \times A \times \Omega_{(A)} \rightarrow \mathbb{R} \tag{3.2}$$

to compute the score for the report $Pr_{(E)}$ from agent $E$ given that the probabilities used to sample the action are $\Phi\left(Pr_{(m)}\right)$, the selected action is $A$ and the outcome is $\Omega_{(A)}$. For simplicity, we will omit the last three inputs and denote the decision scoring rule function as $S\left(Pr_{(E)}\right)$.

The relationship between a decision scoring rule and a scoring rule is:

$$S\left(Pr_{(E)}, \Phi\left(Pr_{(m)}\right), A, \Omega_{(A)}\right) = \frac{1}{\Phi_A\left(Pr_{(m)}\right)} \hat{S}\left(Pr_{(E)}, \Omega_{(A)}\right) \tag{3.3}$$

where $\hat{S}\left(Pr_{(E)}, \Omega_{(A)}\right)$ is any strictly proper scoring rule, such as the logarithmic scoring rule or the Brier score, and the decision rule $\Phi\left(Pr_{(m)}\right)$ has full support. In other words, strictly proper decision scoring rules calculate a proper score for a certain action and outcome by scaling up the inverse of

the action's probability from the decision rule. As a result, the expected scores from strictly proper decision rules do not depend on the actions' probabilities from the decision rule. We further define $S\left(Pr_{(E)}, \Phi\left(Pr_{(m)}\right), A, \Omega_{(A)}\right) = 0$ when $\Phi_A\left(Pr_{(m)}\right) = 0$ to include decision rules without full support, such as deterministic decision rules which we will discuss in Section 3.4.

### 3.3.4 Continuum-armed Contextual Bandit Learning

As outlined in Section 3.3.2, each agent receives a report from the previous agent (or the principal if the agent is the first agent to report) and a private signal from the environment. Based on this context the agents return an updated report which is scored. The updated report represents an agent's action, and therefore we can treat each agent as a continuum-armed contextual bandit agent. In many learning algorithms, the continuum-armed bandit problem is discretised to a finite-armed bandit problem [47], [52]. In this study, we treat this problem differently, by learning parameters which generate updated reports with the policy gradient method that maximises the expected score $\mathbb{E}\left[S\left(Pr_{(E)}\right)\right]$ of agent $E$ [53].

Formally, at time step $T$, we assume that agent $E$ keeps a matrix of parameters $\Theta_{(E)}$. Given the context $C_{(E)}$ and the prior probability $Pr_{(E-1)}$, the agent will construct a $k$ dimensional density function. We assume that the density function is a k-dimensional normal distribution $N\left(\mu\left(C_{(E)}, Pr_{(E-1)}, \Theta_{(E)}\right), \sigma^2\right)$ with the means $\mu\left(C_{(E)}, Pr_{(E-1)}, \Theta_{(E)}\right)$. After sampling a vector of log-odds $H_{(E)} \sim N\left(\mu\left(C_{(E)}, Pr_{(E-1)}, \Theta_{(E)}\right), \sigma^2\right)$ from this probability density, the agent computes the updated probabilistic report as:

$$Pr_{(E,A)} = \frac{1}{1 + \exp\left(-H_{(E,A)}\right)} \tag{3.4}$$

for action $A$.

At time step $T$ agent $E$ updates its parameters to maximise the expected score by gradient ascent

$$\Theta_{(E,T+1)} = \Theta_{(E,T)} + \alpha \frac{\partial \mathbb{E}\left[S\left(Pr_{(E,T)}\right)\right]}{\partial \Theta_{(E,T)}} \tag{3.5}$$

where $\alpha$ is the learning rate. The approximation of the gradient from an agent's experience follows the methodology described in [53], [54]. Further detail is provided in Section 3.3.5.

### 3.3.5 Simulation Setup

In the simulations, we use the classic urn problem as the model for the Bernoulli bandit problem. Specifically, the principal will face an environment which consists of k urns. Each urn represents an action of the k actions in the Bernoulli bandit problem. There are two types of urns, which are red type (1) and blue type (0), representing the possible outcomes of the action, i.e. the Bernoulli variable. The type is hidden from the principal until the principal selects the urn. The type of the other (unselected) urns, however, remains hidden.

At the beginning of each time step $T$, $k$ prior probabilities $Pr_{(0)}$ will be sampled from a normal distribution (in log-odds form), one for each urn. The Bernoulli type of urn $A$ will be sampled using the prior probability $Pr_{(0,A)}$. The prior probabilities will be given to the principal and they will be re-sampled at each time step. Urns contain multiple balls of two colours, red and blue. The composition is determined by the type of the urn and remains fixed for all time steps $T$. Simulations in section 3.4 are all conducted in an environment with two urns each of which can be Bernoulli type 0 or 1. A Bernoulli type 1 urn contains 2/3 red balls and Bernoulli type 0 contains 1/3 red balls.

A number of $J$ balls will be randomly sampled with replacement from one or multiple urns by an agent. The colour of these balls constitutes the private signal of the agent from the environment. The colour of the balls (and their origin), as well as the prior probabilities $Pr_{(0)}$ or the previous updated report $Pr_{(E-1)}$ jointly form the contextual information vector $C_{(E)}$ of an agent. For our setting with two urns to select from, two types of urns and balls, the context can be implemented as a vector with 6 elements (see equation 3.6), where $c_{r1}$ represents the number of red balls drawn from urn 1. $c_{b1}$ represents the number of blue balls drawn from urn 1, and $c_{p1}$ is the log odds transformed prior report for urn 1. Similarly, $c_{r2}$, $c_{b2}$ and $c_{p2}$ represent the number of red and blue balls as well as the log odds transformed prior report for urn 2.

The contextual information vector multiplied with the matrix of learning parameters $\Theta_{(E)}$ of the agent gives the means $\mu\left(C_{(E)}, \Theta_{(E)}\right)$ for a normal distribution $N\left(\mu\left(C_{(E)}, \Theta_{(E)}\right), \sigma^2\right)$. The log odds $H_{(E)}$ of the actual report will be sampled from this normal distribution. The computation of the means for the updated report can be written as

$$\begin{pmatrix} c_{r1} \\ c_{b1} \\ c_{p1} \\ c_{r2} \\ c_{b2} \\ c_{p2} \end{pmatrix}^{\mathsf{T}} \times \begin{pmatrix} \theta_{r1}^{(1)} & \theta_{r1}^{(2)} \\ \theta_{b1}^{(1)} & \theta_{b1}^{(2)} \\ \theta_{p1}^{(1)} & \theta_{p1}^{(2)} \\ \theta_{r2}^{(1)} & \theta_{r2}^{(2)} \\ \theta_{b2}^{(1)} & \theta_{b1}^{(2)} \\ \theta_{p2}^{(1)} & \theta_{p2}^{(2)} \end{pmatrix} = \begin{pmatrix} \mu_1 & \mu_2 \end{pmatrix} \tag{3.6}$$

The result of the multiplication $\mu_1$ is the mean log-odds transformed report for urn 1, and $\mu_2$ is the mean log-odds transformed report for urn 2. The reported log odds for urn 1 and urn 2 will be sampled from a normal distribution with this mean and a fixed variance, i.e., $H_1 \sim N\left(\mu_1, \sigma^2\right)$, and $H_2 \sim N\left(\mu_2, \sigma^2\right)$.

We implement logarithmic scoring rules in the simulation. Therefore, the decision scoring rule can be written as

$$\begin{aligned} &S(Pr_{(E)}, \Phi\left(Pr_{(m)}\right), A, \Omega_{(A)}) \\ &= \begin{cases} \frac{1}{\Phi_A\left(Pr_{(m)}\right)} \log \frac{Pr_{(E)}}{Pr_{(E-1)}}, & \text{if } \Phi_A\left(Pr_{(m)}\right) > 0 \text{ and } \Omega_{(A)} = 1 \\ \frac{1}{\Phi_A\left(Pr_{(m)}\right)} \log \frac{1-Pr_{(E)}}{1-Pr_{(E-1)}}, & \text{if } \Phi_A\left(Pr_{(m)}\right) > 0 \text{ and } \Omega_{(A)} = 0 \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{3.7}$$

This implies that an agent receives a difference in the logarithmic score between the agent's score and the previous report's score. In other words, the agent is scored for how much the accuracy of the preceding report is improved or worsened. In our simulation, we use a stochastic decision rule that favours the type one urn. In other words, we assign the highest probability to the urn that is forecasted to be most likely to be type one. In our two urns simulations, we assign a probability of 90% to selecting probability to the urn that is reported to be most likely type one and 10% to the other. From the principal perspective, this is essentially an $\epsilon$ greedy two-arm bandit problem with $\epsilon =$ 10%. A fixed exploration rate will cause a loss of performance, which is clear in our simulation results in section 3.4.1. One can use bandit learning techniques to optimise the principal, but this is not the focus of this work.

Agent E's initial parameters $\Theta_{(E)}$ is sampled from a standard normal distribution $N(0, 1)$. At time step $T$, we refer a tuple $\left(C_{(E,T)}, \mu_{(E,T)}, H_{(E,T)}, S\left(Pr_{(E,T)}\right)\right)$ that consists of useful information

for agent $E$ to learn as an experience. We use the experience replay buffer technique as in [55], [56]. The difference is, in our simulation, each experience is independently and identically distributed, and therefore we apply this technique only for the efficiency of the hardware usage. After agent $E$ receives the score, it will store the experience in its own experience replay buffer. If the existing experience number exceeds a certain threshold, the latest experience will replace of the oldest one. Afterwards, a fixed number $F$ of experience tuples will be uniformly sampled from the experience replay buffer for an update. Assume the experience tuple at time step $I$ is within that $F$ samples, the gradient for the tuple can be obtained by

$$G_{(E,I)} = C_{(E,I)} \times \left(S\left(Pr_{(E,I)}\right) - B\left(C_{(E,I)}\right)\right) \times \frac{H_{(E,I)} - \mu_{(E,I)}}{\sigma^2} \tag{3.8}$$

where $B(C_{(E,I)})$ is a baseline function that does not vary with action $A$ but only depends on the contextual vectors [54]. Finally, an average gradient of a mini-batch will be computed by

$$\overline{G_{(E,T)}} = \frac{1}{F} \sum_{I} G_{(E,I)} \tag{3.9}$$

to update the parameters of time step $T$ as shown in equation 3.5.

### 3.3.6 Performance Evaluation

The principal's objective is to select urns of type 1. To track the systems' performance we, therefore, use the sum of Bernoulli outcome variable $\sum_{T=1}^{n} \Omega_{(A,T)}$ of the executed action A. Additionally, we use the error of the final reports, defined as the mean squared residual between the final aggregated reports that the principal receives from the sequential reporting and the correctly updated Bayesian posterior $\widehat{Pr}$ of an observer with access to the entire environmental context.

$$Er = \sum_{A} \left(Pr_{(m,A)} - \widehat{Pr}\right)^2 \tag{3.10}$$

This metric is used only for evaluation purposes as the signals are not accessible by the principal during the training phase.

## 3.4 Simulation Results

We use our simulation setup to investigate three different decision market scenarios. In the first set of simulations (Section 3.4.1) we compare a multi-agent system with a centralised agent. In the multi-agent system, signals are distributed across the individual agents, while in the centralised system, there is a single agent that receives all signals. In both cases, a stochastic decision rule is used. The results show that the multi-agent contextual bandit system performs as well as the centralised system.

In the second set of simulations (Section 3.4.2), we analyse decision markets with deterministic decision rules, starting with a single agent. The results show that because such markets are not incentive-compatible, agents can learn strategies that lead to reports which differ from correct information aggregation. The agent's behaviour resembles strategies described by Othman and Sandholm [6]. However, in many cases, agents learn to provide accurate forecasts (despite receiving a lower reward), which indicates that strategically inaccurate forecasting is difficult to learn with gradient methods.

In the third set of simulations (Section 3.4.3), we are investigating decision markets with deterministic decision rules and multiple agents. We observe novel strategies leading to non-trivial interactions between agents, with strategically distorted reporting by the agent who reports first. The results show that while final reports are as accurate as for decision markets with stochastic decision rules, the distribution of rewards for agents under stochastic decision rules is fairer compared to the distribution under deterministic decision rules.

### 3.4.1 Decision Markets with Stochastic Decision Rules: Distributed vs. Centralised Systems

In this set of simulations, we compare the performance of a system with $J$ individual agents, each of which receives a single signal, with a corresponding centralised system where a single agent receives $J$ signals. The simulations follow the approach described in Section 3.3, with $J$ being set to 3, 5, 9, and 15. Each signal is a draw of a single ball (sampled with replacement) from one of the urns.

As shown in Figure 3.2, we observe that the mean square error (MSE) of the final report decreases rapidly and stabilises close to zero in both multi-agent and centralised systems. The MSE declines faster in the multi-agent system, compared to the centralised counterpart when the agent or signal number is high. This is analysed in more detail further below. Once converged, the average rewards for both systems are very similar, with the reward being defined as one when the selected urn turns

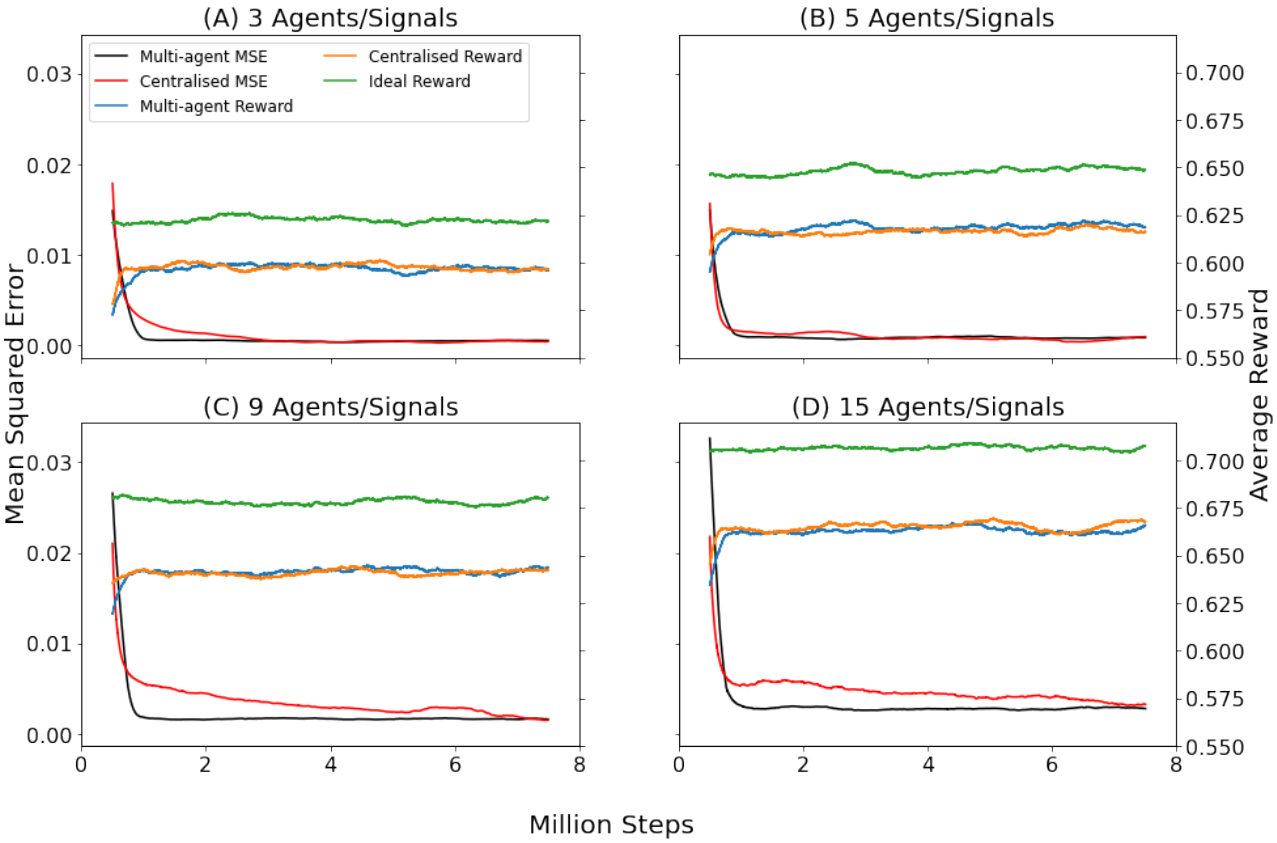


Figure 3.2: **System performance of multi-agent and centralised systems**. Panel (A)-(D) show simulations with 3,5,9 and 15 signals. In the multi-agent system, each agent receives one signal. In the centralised counterpart, a single agent receives all signals. The black line and red line are the running averages of the mean squared error of multi-agent and centralised systems, respectively. The green line is the average received reward for a principal who chooses an action according to a posterior from a correct Bayesian model that can use all available information. The blue line shows the actual reward received for the multi-agent systems. The orange line is the reward received by the centralised systems. The errors and rewards for the centralised and distributed systems are very similar. The rewards are lower compared to the Bayesian model, with the difference arising from the use of a stochastic decision rule in the multi-agent and centralised system.

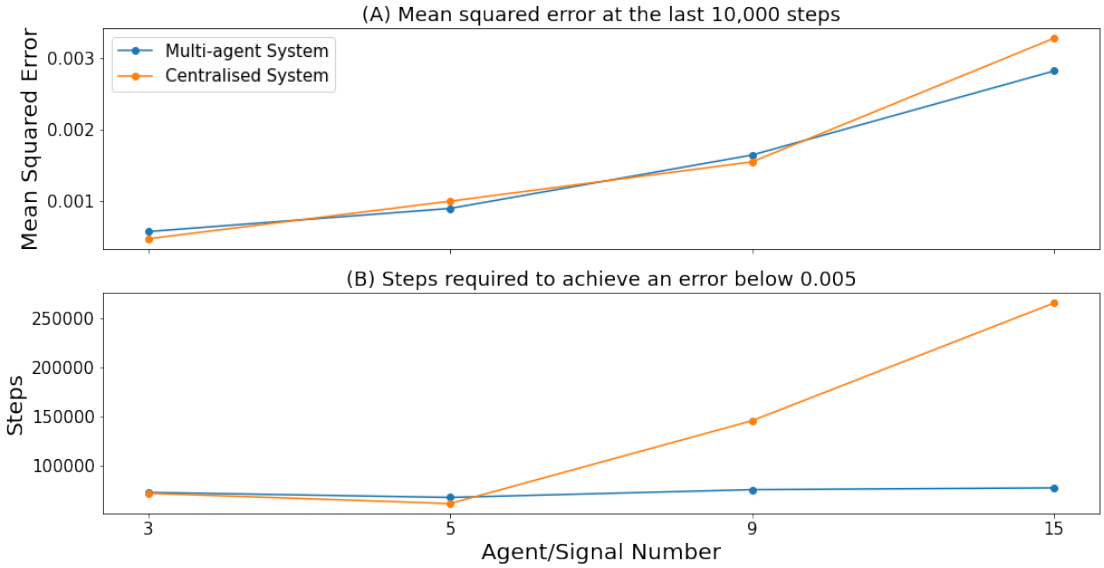


Figure 3.3: **Final accuracy and time to convergence in multi-agent and centralised systems**. Plot (A) shows the mean squared error in the last 10,000 steps. Plot (B) shows the training iterations required to reach a mean squared error below 0.005.

out to be of the preferred type (red), and zero when it is not. Note that the gap between the actual reward and the ideal reward is due to the nature of stochastic decision rules, which assigns a positive probability to select a sub-optimal action. The performance will be close to the ideal reward if we account for the disadvantage of the stochastic decision rules.

We further analyse performance by investigating the time to convergence, and the average squared residual error after convergence. We recorded the average MSE at the last 10,000 steps (see Figure 3.3A) and the number of training steps (see Figure 3.3B) required for a system to reach an acceptable performance (here set to 0.005). Figure 3.3A shows the converged performance of both systems increases similarly with an increasing number of signals. Figure 3.3B, however, indicates that the steps required for the centralised system to reach an acceptable performance increase with the number of signals it receives. In contrast, the multi-agent system does not show any relation between the number of agents and the steps required for training to reach a MSE of 0.005. In other words, the multi-agent system shows better scalability.

Figure 3.4 shows the progress of learning parameters $\Theta$ for a centralised agent (panels A and B) and a distributed agent from a multi-agent system (panels C and D). As shown in the figure, the learning parameters converge to theoretically ideal values. For a centralised agent, the parameters that relate prior probabilities with reports converge slower than in the distributed counterpart. This is because compared to the contribution of five balls to the final report, the information provided by

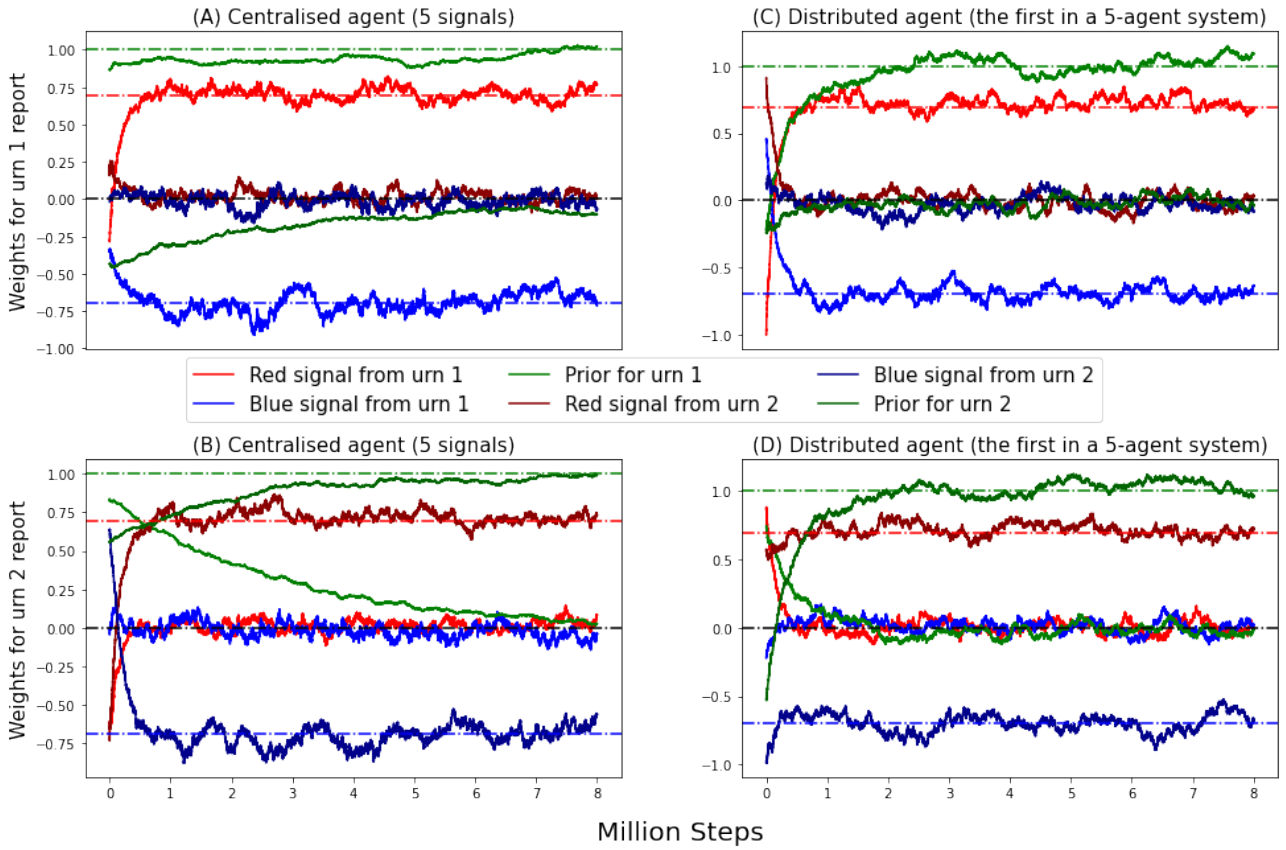


Figure 3.4: **Progress of the learning parameters for a centralised and distributed agent**. Plot (A) and plot (B) show the six parameters that determine the report for urn 1 and urn 2 respectively for a centralised agent. Plot (C) and plot (D) show the parameters for a distributed agent. These parameters are the left column in the matrix in equation 3.6. Specifically, the red line with the label 'Red signal from urn 1' in the legend is the history of $\theta_{r1}^{(1)}$ and the parameter can be interpreted as the weight of receiving a red ball from urn 1 for the posterior report of the urn 1. Similarly, 'Blue signal from urn 1', 'Red signal from urn 2' and 'Blue signal from urn 2' show the weights corresponding to the signal colour and which urn it comes from. The green lines labelled 'Prior for urn 1' and dark green lines labelled 'Prior for urn 2' show the weights for prior probabilities passed by the previous agent or the principal on the updated reports. The dash lines are the ideal value for the learning parameters.

the prior probability is less important. The time to convergence increases when the signal number increases, while in the multi-agent counterpart each agent learns every parameter at a similar pace.

Overall, from the simulation results, we find that a multi-agent bandit system that uses a decision market with stochastic decision rule can learn to make highly accurate reports, similar to the corresponding centralised system. The advantage of our multi-agent system is that the contextual information an agent receives can remain private to the agent. More specifically, the agents reveal how the contextual information they receive affects their probabilistic reports, but they do not need to reveal the contextual information itself, or the weights that were used to link the context with the report.

### 3.4.2 Decision Markets with Deterministic Decision Rules: Single Agent Simulations

Section 3.4.1 demonstrates that decision markets with stochastic decision rules can elicit information distributed over multiple agents. These agents can be computational and can use the decision market score to learn using their information to make accurate forecasts. However, decision markets with stochastic decision rules are inefficient because they entail that the principal sometimes selects an action that is forecasted not to be the best possible action. It is in the interest of the principal to use a deterministic decision rule and select the action that has the highest probability to achieve the desired outcome.

Such a decision rule, however, is not incentive-compatible and has been theoretically shown to be manipulatable by rational and myopic agents [6], [13]. From a reinforcement learning perspective, a score derived from a decision market with a deterministic decision rule cannot be expected to allow agents to learn providing reports that can be interpreted as accurate probabilistic reports. Othman and Sandholm's work discusses strategies of an agent who is the last to make a report and can benefit from strategically inaccurate reporting, and show that there are situations where subsequent agents with the same piece of information have no incentives to correct such inaccuracies.

We here investigate the strategies that are learned by a single agent with a single signal as specified in Section 3.3. Further simulations with multiple agents with independent information are investigated in Section 3.4.3.

We find that in our simulations of a single agent in a decision market with a deterministic decision rule, depending on initial parameters different strategies are learned. In most simulations, agents learn to make reports similar to those in the stochastic decision markets. The reports of the agents

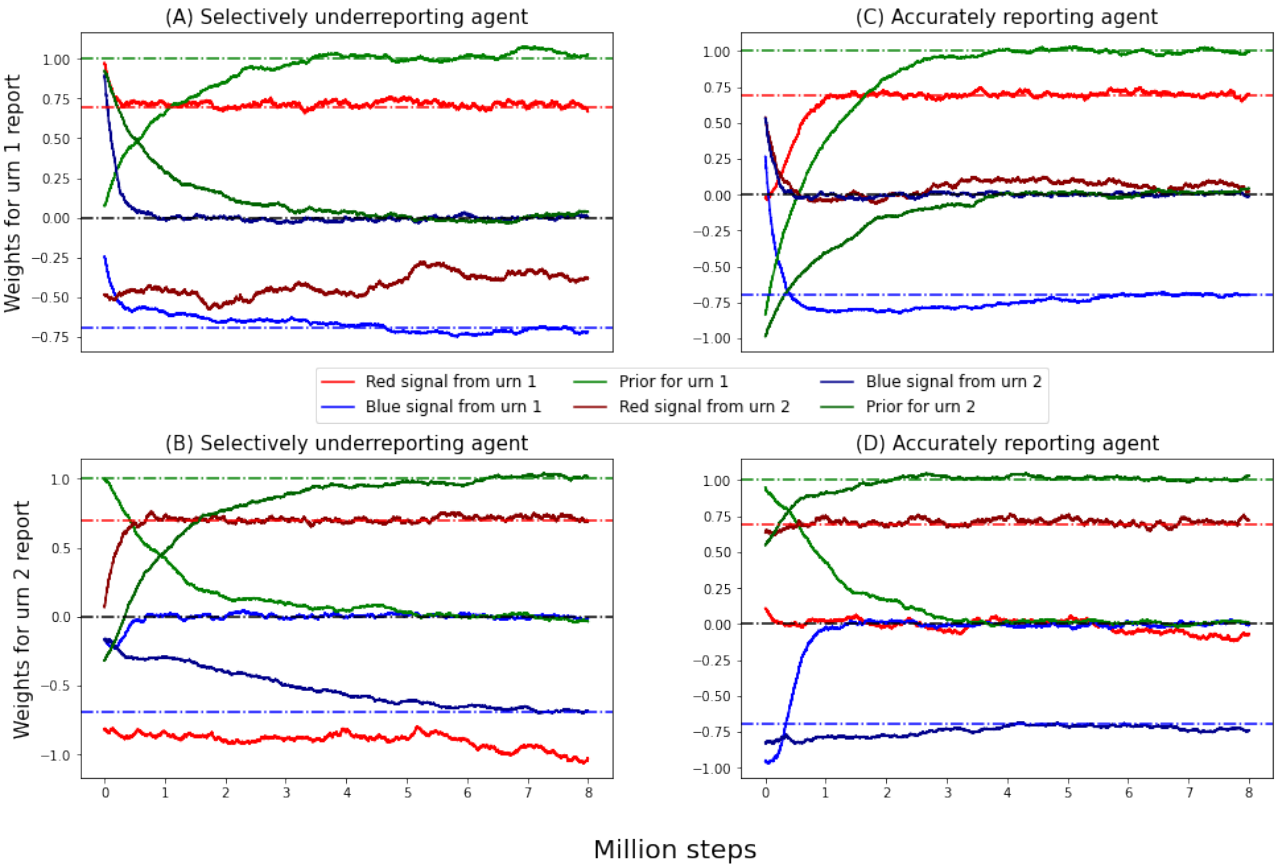


Figure 3.5: **Progress of the learning parameters for a selectively underreporting and accurately reporting agent**. Plot (A) and (B) are weights for the log odds report from a selectively under-reporting agent. Plot (C) and (D) are weights from an accurate reporting agent. The accuracy is determined by if the posterior report agrees with the posterior Bayesian inference with all the signals accessible.

represent accurately estimated probabilities for the outcomes, given the information available. Such agents could be seen as 'honest' agents. We also observe agents learn accurately report the probability for one urn, but provide a report that is lower than the accurate report for the other urn. The weights of an accurately reporting agent and a 'selectively underreporting' agent are shown in Figure 3.5.

While the weights for an accurate agent are the same as the ones learned by the agents in the decision markets with stochastic decision rule, the selectively underreporting agent differs in two weights. These two weights are zero for the accurate agents but negative for the selectively underreporting agent. One of these weights lowers the report for urn 2 when a red signal for urn 1 is received; the other weight lowers the report for urn 1 when a red signal for urn 2 is received. This contrasts with an accurate agent who would not lower a report for an urn for which no signal has been received.

A selectively underreporting agent can benefit from this strategy (see Figure 3.6) because if reporting accurately, the agent receives a larger payoff if the urn is selected from which the signal was received. A single agent can therefore maximise its payoff by reporting accurately for the urn from

which the signal was received, and submitting a report for the other urn that is sufficiently low such that the former rather than the latter urn is selected. However, we only observe agents learn to selectively underreport when obtaining a red signal. In principle, selective underreporting also maximises the payoff for an agent who receives a blue signal. However, the blue signal lowers the probability for an urn to be of the favourable type; therefore this strategy requires lowering the report for the other urn much more. If underreporting is insufficient to change the choice of the urn, it is disadvantageous, making selective underreporting difficult to learn with local, gradient-based methods when a blue signal is received.

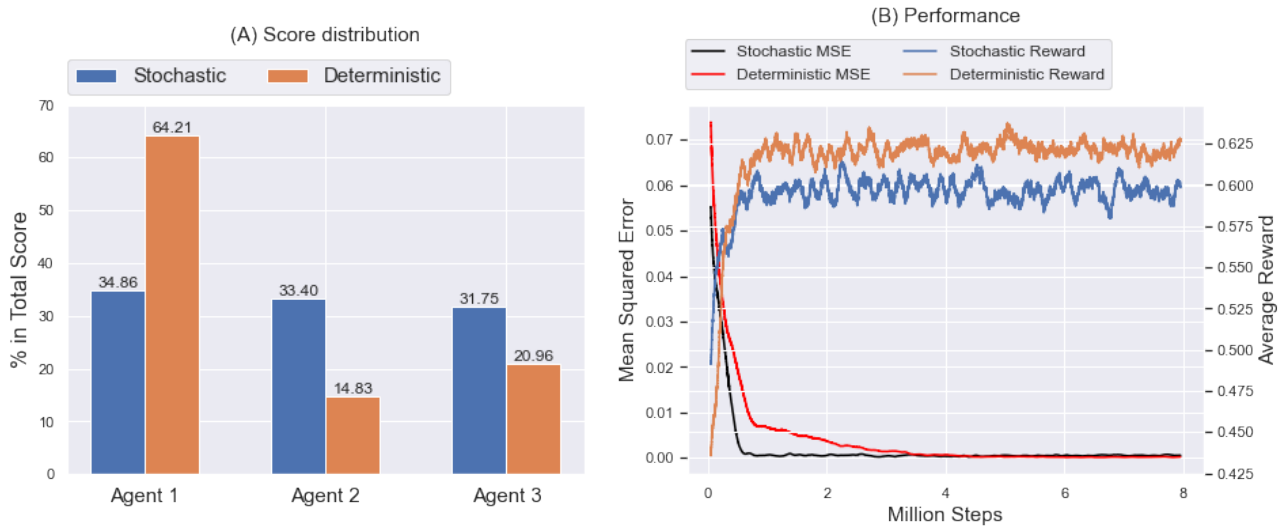


Figure 3.6: **Reward distribution and performance of decision markets with stochastic and deterministic decision rules**. Plot (A) shows the average score ratio comparison between individual agents in different reporting sequences and different decision markets. Plot (B) shows the performance comparison between a three-agent system based on a decision market with a stochastic and deterministic decision rule.

In summary, in our single-agent simulations, we find strategic manipulation similar to the strategies expected by Othman and Sandholm. An agent can benefit from manipulating the final report and misleading the principal to a sub-optimal action. However, such a strategy would be vulnerable to subsequent agents with the same piece of the information who have incentives to correct previous underreporting.

### 3.4.3 Decision Markets with Deterministic Decision Rules: Simulations with Multiple Agents

The multi-agent dynamic in a deterministic decision market has so far not been investigated. We here use our system to investigate the interaction between three agents in a decision market with a MAX

decision rule.

In Section 3.4.2, we observe agents learn to 'game' a decision market by 'honestly' reporting about the urn that offers the larger expected reward to the agents while underreporting (or 'trash-talking') the other option. This strategy maximises expected payoffs but could be exploited by subsequent agents. Othman and Sandholm discuss a strategy where the final agent strategically inflates the final report of the urn that offers a higher reward than accurately reporting to the agent under certain conditions (see example 2 in [6]). Such a strategy can also be profitable, and it cannot be exploited by subsequent agents with the same information. However, it is unclear what strategies are beneficial when multiple agents with conditionally independent signals exist. To study this situation, we use simulations with three agents. As in the previous simulations (Section 3.4.1), each agent will draw a ball from a random urn and return it after privately recording the colour. The agents make sequential reports.

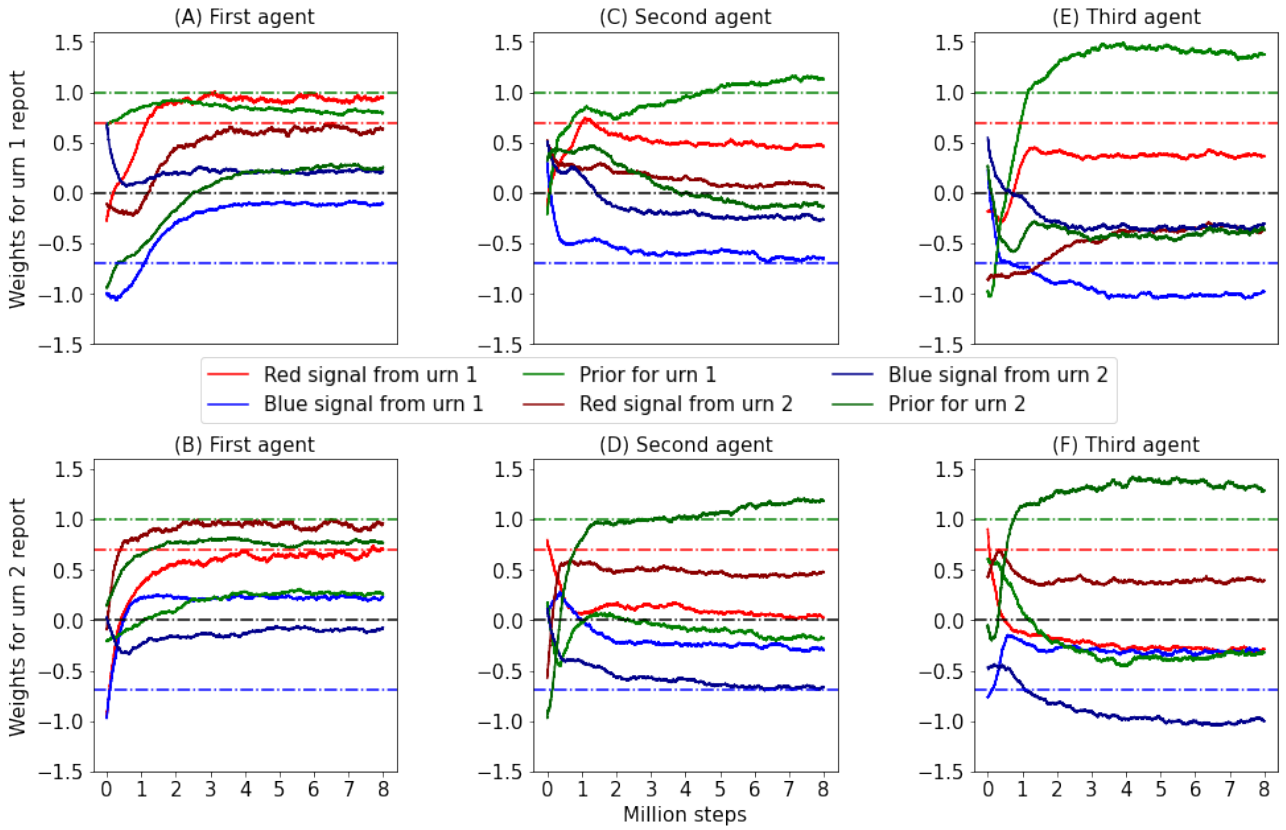


Figure 3.7: **Progress of the learning parameters for a three-agent system based on a decision market with a deterministic decision rule**. The $2 \times 3$ grid plots show the change of weights from each agent. The dashed lines show the weights required for strict Bayesian updates like reports. The weights in panels (A) and (B) show the weights of the first agent in a decision market with a deterministic decision rule; (C) and (D) of the second agent; and (E) and (F) of the final agent.

Figure 3.7 shows the agents' strategies in terms of the weights learned by the agent. The weights

for the first agent are all increased compared to a Bayesian agent accurately reporting the probabilities given the information. This means that the first agent strategically overreports. Subsequent agents essentially correct this initial overreporting such that the final reports become quite accurate (see Figure 3.8.). Intuitively, this strategy is beneficial because the first agent learns that the urn with the highest probability will be selected. By providing increased reports on all outcomes, the first agent can benefit from increasing all reports. The agent increases the report most when receiving a red signal, but it also increases the reports for the urn from which no signal has been received. When a blue signal is received, it provides the lowest report, though this report is still larger than one given by an accurate Bayesian agent (see Figure 3.8.). Interestingly, this leads to a reward distribution that substantially favours the first agent. While agents in the decision markets with stochastic decision rules receive very similar rewards, under a deterministic decision rule, the first agent receives a much higher expected score. Thus, while from the principal's perspective the decision-making performance is very similar to the strategies emerging in our simulations, the expected scores offered under a stochastic decision rule can be seen as more 'fair' compared to the scores under a deterministic decision rule.

With the simulation, we reveal interesting dynamics in a multi-agent system with a deterministic decision market. The first agent learns an overreporting strategy and takes the lion's share of the score. The subsequent agents correct the report of the first agent, which results in a surprisingly accurate final report.

## 3.5 Conclusion and Discussion

In this paper, we investigate the use of decision markets, which are economic mechanisms for decision-making based on distributed information, for contextual bandit learning in a multi-agent system. Unlike existing multi-agent systems, we assume contextual information is distributed across multiple self-interested agents who own their information and require incentives to reveal it and learn to interpret it. This scenario is relevant for real-world commercial models because contextual information such as user profiles or patient information is often proprietary and potentially too sensitive to be made available to a centralised model. Rather than revealing contextual information, training data and learned parameters, each agent solely need to reveal its predictions for the available actions. Because a decision market offers a proper score for these predictions, it allows training multiple self-interested models without accessing private contextual information. The score aligns the individual agents with the system's performance such that as the agents improve their individual scores, the

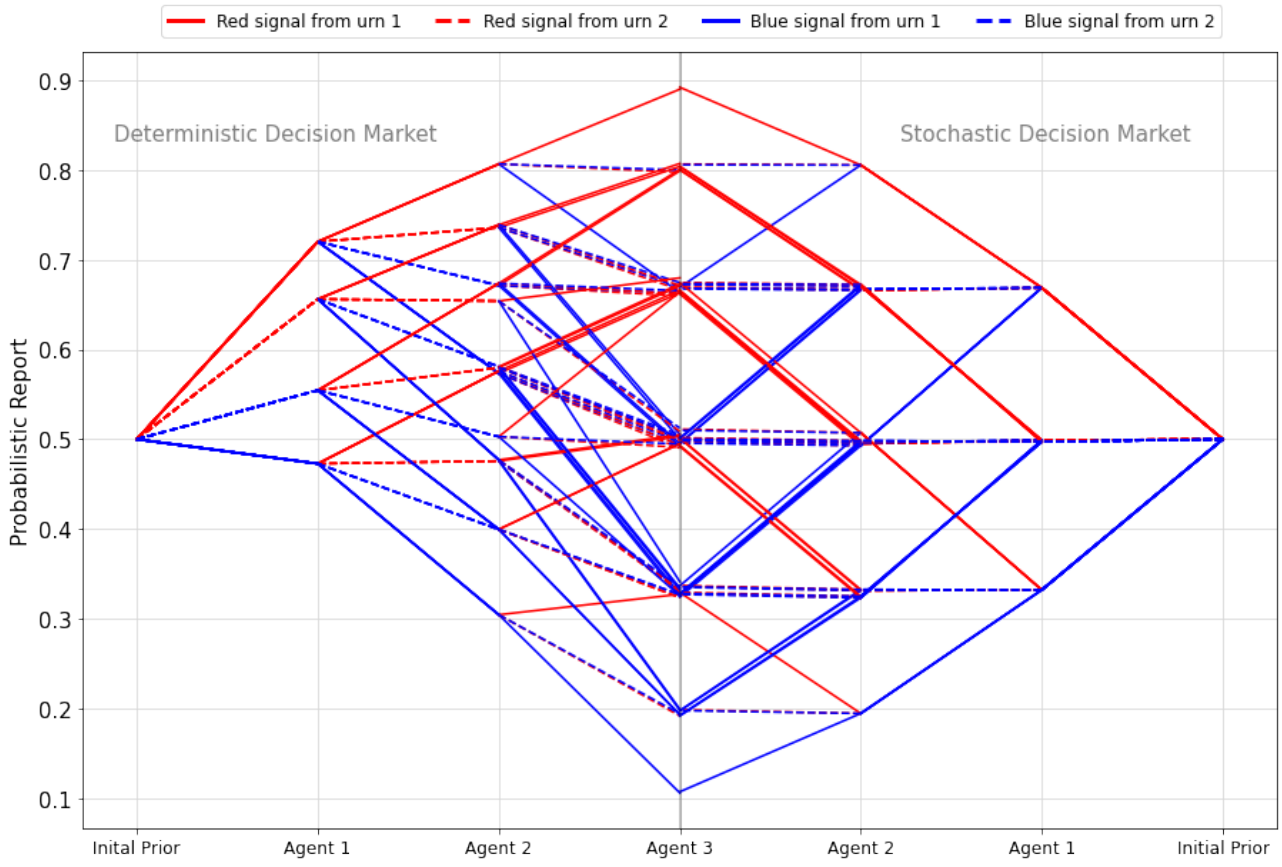


Figure 3.8: **Change of probabilistic reports in decision markets with deterministic and stochastic decision rules**. The plot has shown the change of the probabilistic report for urn 1 passed by each agent (the report for urn 2 is symmetric). A line labelled 'Red signal from urn 1' shows the change in the report after receiving a red ball from urn 1. The other labels follow the same convention. A solid line indicates the ball is from urn 1 (the reporting urn) and a dashed line means the ball is from urn 2 (the other urn).

collective decision-making efficiency improves as well.

Our simulations show that the decision market based multi-agent system can train self-interested agents to achieve an equally efficient performance as a centralised trained counterpart with accessibility to all pieces of the same contextual information. This result indicates that coordination problems and the non-stationary issue that can arise in multiagent systems do not affect the performance of our system in the simulations.

We use our system to investigate the dynamics of multi-agent interactions under different decision rules. While decision markets with stochastic decision rules allow the agents to learn to make highly accurate forecasts, the stochastic decision rule reduces the efficiency of decision-making. We, therefore, simulate how agents learn under a deterministic decision rule. In the one-agent system with a deterministic decision rule, the agent, who learns with a gradient-based algorithm, can learn to selectively diminish the probability of the action for which it does not have any information. Moreover,

the agent benefits from this underreporting strategy because it ensures that the action is selected for which the forecasts are expected to be scored higher. However, the learned strategies depend on the initial values for the learning parameters, and often the agents learn to report accurately. This highlights the limitations of gradient methods used in the simulations to find the global optimum.

In a three-agent system with a deterministic decision rule, we observe the first agent learn to overreport for both actions and thereby gain a significant first-mover advantage. The subsequent agents gradually correct the report which results in an accurate final report. The average scores for individual agents are less equitably distributed under a deterministic decision rule, compared to a stochastic decision rule. Our results suggest that our simulation-based approach to testing economic mechanisms in a multiagent learning context can identify strategies that are beneficial to the individual agents and their consequences for the overall system performance.

A future study could use global optimising techniques to find globally optimal strategies and thereby help identify Nash equilibria. For instance, in Section 3.4.2 we mentioned that selective underreporting for a blue signal is difficult to learn with local gradient-based methods. This is because underreporting has to be sufficient to change the decision. Less strong underreporting does not change the decision but reduces accuracy for the selected action and therefore reduces the score. Global optimising techniques, however, have a much higher computational complexity. Another future study direction is overcoming the limitation of stochastic decision rules, which sometimes require the principal to select the action that is predicted to be sub-optimal. This requires a mechanism that allows for a deterministic action selection while simultaneously maintaining incentive compatibility. A promising approach might be to use peer prediction methods to resolve the decision markets.

# Chapter 4

# Proxy Forecasting to Avoid Stochastic Decision Rules in Decision Markets

Chapter 4 mainly consists of the paper titled '*Proxy Forecasting to Avoid Stochastic Decision Rules in Decision Markets*' preprinted in `https://arxiv.org/abs/2303.10857` [57]. Both Chapter 2 and 3 use a decision market with a stochastic decision rule, which suffers from inefficiency due to the randomness required in decision-making. This chapter describes several related mechanisms that can lift this inefficiency and achieve collective deterministic decision-making.

## Summary of Notation

| | |
|---|---|
| $k$ | number of available actions |
| $m$ | number of agents |
| $T$ | time step variable, $T \in \{1, 2, \ldots, n\}$ |
| $A$ | action variable, $A \in \{1, 2, \ldots, k\}$ |
| $\Omega_{(A)}$ | outcome for action $A$ (Bernoulli variable) |
| $P$ | principal |
| $E$ | agent, $E \in \{1, 2, \ldots, m\}$ |
| $D_{(P,A)}$ | the principal's signal about action $A$ (Bernoulli variable; used as proxy) |
| $D_{(E,A)}$ | agent $E$'s signal about action $A$ (Bernoulli variable) |
| $Pr_{(E,A)}$ | probabilistic report of agent $E$ for principal $P$'s signal (i.e., the proxy) being $D_{(P,A)} = 1$, $Pr_{(E,A)} \in [0, 1]$ |
| $Pr_{(E)}$ | vector of reports from agent $E$ over all possible proxies, $Pr_{(E)} \in [0, 1]^k$ |

| | |
|---|---|
| $C_{(E)}$ | contextual vector consisting of agent $E$'s own signal $D_{(E)}$ and previous report $Pr_{(E-1)}$ |
| $\Theta_{(E)}$ | policy parameter matrix of agent $E$ |
| $\mu_{(E,A)}$ | parameters used to sample the log-odds report from agent $E$ about action $A$ |
| $X_{(E,A)}$ | actual log-odds sampled with $\mu_{(E,A)}$ |
| $\mu_{(P,A)}$ | the principal preferences over action $A$ as use for training the principal |
| $\phi_{(P)}$ | the probability distribution over the actions used by the softmax function for principal learning, $\phi_{(P)} \in [0,1]^k, \sum \phi_{(P)} = 1$ |
| $s(Pr_{(E,A)}, D_{(P,A)})$ | scoring rule that quantifies the accuracy of $Pr_{(E,A)}$ according to the principal's signal $D_{(P,A)}$ |
| $G_{(E,T)}$ | a matrix of approximated partial derivatives of the expected score of expert $E$ with respect to policy parameters |
| $G_{(P,T)}$ | a matrix of approximated partial derivatives of the expected reward of principal $P$ with respect to policy parameters |
| $B_{(P,T)}, B_{(E,T)}$ | baseline functions that reduce the variances but do not change the means of rewards |
| $\widehat{Pr_{(A)}}$ | Bayesian posterior probabilities for the principal's signal given all the signals received by the agents, $Pr\left(D_{(P,A)} \middle\vert D_{(1,A)}, \ldots, D_{(m,A)}\right)$ |
| $\widehat{Pr'_{(A)}}$ | Bayesian posterior probabilities for the actions' outcomes given the agents' and the principal's signals, $Pr\left(\Omega_{(A)} \middle\vert D_{(1,A)}, \ldots, D_{(m,A)}, D_{(P,A)}\right)$ |

## Abstract

Information that is of relevance for decision-making is often distributed, and held by self-interested agents. Decision markets are well-suited mechanisms to elicit such information and aggregate it into conditional forecasts that can be used for decision-making. However, for incentive-compatible elicitation, decision markets rely on stochastic decision rules which entails that sometimes actions have to be taken that have been predicted to be sub-optimal. In this work, we propose three closely related mechanisms that elicit and aggregate information similar to a decision market, but are incentive compatible despite using a deterministic decision rule. Following ideas from peer prediction mechanisms, proxies rather than observed future outcomes are used to score predictions. Proxies are observable events that are statistically correlated with the future outcomes of interest. The first mechanism requires the principal to have her own signal, which is then used as a proxy to elicit information from a group of self-interested agents. The principal then deterministically maps the aggregated forecasts and the proxy to the best possible decision. The second and third mechanisms expand the first to cover a scenario where the principal does not have access to her own signal. The principal offers a partial profit to align the interest of one agent and retrieve its signal as a proxy; or alternatively uses a proper peer prediction mechanism to elicit signals from two agents. Aggregation and decision-making then follow the first mechanism. We evaluate our first mechanism using a multi-agent bandit learning system. The result suggests that the mechanism can train agents to achieve a performance similar to a Bayesian inference model with access to all information held by the agents.

## 4.1 Introduction

Consider a company that has developed two alternative product lines and needs to decide which one to move into production. The decision maker (principal) has a noisy signal about the likelihood of success of the product lines and also engages a group of experts who have their own independent noisy signals. The aim of the decision maker is to make the best possible decision, given the experts' and her own signals. The experts are self-interested and require incentives to reveal their information. The principal needs to design the incentives such that the experts cannot benefit from manipulative strategies that result in larger rewards for inaccurate information.

Collective decision-making processes need to address two challenges. Firstly, a process needs to incentivise self-interested participants to provide their information accurately. Secondly, multiple pieces of information need to be aggregated and mapped to the final decision. Several options for eliciting advice for decision-making have been discussed in [26]. One option is to elicit forecasts about the consequences of the available actions and make a decision based on these forecasts. An alternative is to directly solicit recommendations on which action to take.

Prediction markets are suitable mechanisms to elicit forecasts from groups of experts when realised outcomes do not depend on the selected actions. However, when the future outcomes depend on the choice of the principal, it is difficult to design proper incentives because the unselected actions become counterfactual, and forecasts about their consequences cannot be evaluated by a strictly proper scoring rule. A simple mechanism where the principal first elicits forecasts about the available actions and then selects the action that is forecasted to be most beneficial is therefore prone to strategic manipulation by self-interested experts [6], [13].

As a solution to this problem, proper decision markets have been proposed [1]. Proper decision markets use strictly proper scoring rules to elicit forecasts, and then use a stochastic decision rule to map the elicited forecasts to a probability distribution over the available actions. This probability distribution needs to have full support, i.e., assign each action a non-zero probability of being executed. Forecasts about the available actions can then be rewarded such that the expected payoff for a forecast does not depend on which action is selected.

The stochastic decision rule required to properly incentivise decision markets means that the principal needs to select actions that are forecasted to be suboptimal. This introduces inefficiency in the decision-making process. While the probabilities assigned to suboptimal action can be made arbitrarily small, small probabilities in the decision rule lead to reward distributions with high variance

and to large worst-case losses for the principal [1], [8].

Naturally, the interest of the principal is to elicit collective forecasts for the available actions that are as accurate as possible, and then deterministically select the action that is forecasted to be the most desirable one. In this paper, we propose a set of mechanisms which fulfil these requirements and thus fill a gap in the literature on collective decision-making. By eliciting forecasts for observable proxies for the consequences of the available actions, these mechanisms separate the scores for experts from the decision made by the principal and, therefore, can use deterministic decision rules. Such a proxy is defined to be verifiable and statistically correlated with the desirability of the actions; in the simplest case, this could be a signal held by the principal. The principal then elicits forecasts about this proxy and uses the aggregated reports to deterministically select an action. Our work builds on ideas from peer predictions, where peers provide proxies for an unobservable ground truth.

We illustrate our mechanisms with simulations of learning in a contextual bandit problem where contextual information (i.e., signals) is distributed over multiple self-interested agents. Such a setting was used previously to study multi-agent learning under a decision market mechanism [10].

This paper is organised as follows: In Section 4.2, we briefly discuss related work. In Section 4.3, we propose a mechanism that allows a principal to aggregate distributed information and make the decision deterministically with the assumption that the principal has an independent and identically distributed signal. We then extend the mechanism and describe two variants that work under different assumptions. In Section 4.4, we show simulation results to illustrate that one mechanism can be used for multi-agent learning. In Section 4.5, we conclude the work and discuss future directions.

## 4.2 Related Work

The work presented here is related to work on strictly proper scoring rules, prediction markets, and decision markets. Strictly proper scoring rules for probabilistic forecasts are described in [25]. Prediction markets expand these scoring rules to aggregate information from multiple forecasters [4], [7], [20], [29], [58], [59]. Decision markets are mechanisms to aggregate conditional forecasts for decision-making, and are described in [1], [5], [6], [13]. An application of decision markets to multi-agent contextual bandit system is described in Wang and Pfeiffer; detailed background on multi-bandit problems is provided in [47], [54], [60].

The approach to using a proxy to verify forecasts for unobservable outcomes follows a well-established strategy from peer prediction mechanisms [61]–[65]. In brief, peer prediction mechanisms

incentivise participants to truthfully report a signal they have received. In a real-world application, such a signal could be an experience in a restaurant, which is reported in a review and may serve as a proxy for the restaurant's quality. Peer prediction mechanisms use the correlation between different agents' signals to ensure truth-telling is a Nash equilibrium [61], [62]. When agents perform multiple similar tasks, this mechanism can be designed to ensure that agreeing with another reference report is rewarded while blind agreement is penalised [64]. A number of studies have expanded this mechanism along different dimensions [63], [66]–[71].

Our work is closely related to [72] which proposes proper proxy scoring rules to incentivise a forecaster to truthfully reveal her probabilistic belief with a proxy instead of a future outcome. Witkowski, Atanasov, Ungar, *et al.* use existing empirical data to evaluate their proxy scoring rules. Our work follows this work to avoid the disadvantages of a stochastic decision rule in proper decision markets. We test our mechanism in simulations of a multi-agent bandit learning system that allows self-interested agents to converge to optimal collective decision-making (more detail in Section 4.4.2).

## 4.3 Mechanism Design

This section will outline three closely related mechanisms that use proxies to avoid the disadvantages of stochastic decision-making in proper decision markets. In the first mechanism, we assume that, similar to related work on decision markets [1], [5], [6], a principal decides between multiple alternative actions. Moreover, there is a group of agents each of which receives independent and identically distributed (iid) signals about the likely outcome for one of the actions. Unlike previous work on decision markets, we assume that the principal has her own signal about one of the available actions, which is iid from the other agents' signals. The principal elicits and aggregates the agents' signals by using her own signal as a proxy, i.e. she elicits forecasts from the agents about her own independent signal, and then selects one of the available actions based on these forecasts. In the second mechanism, the principal does not have an own signal, but recruits one of the agents as an advisor and uses the advisor's signal to elicit and aggregate the other agents' signals. Because the principal requires only a single signal to elicit all other signals, she can use a fraction of her own reward as an incentive to align the advisor's interest and retrieve the signal. Thirdly, the principal separates two agents from the rest of the agents, uses peer prediction to elicit the signals from these two agents, and then elicits the remaining agents' signals following the procedure of the first mechanism.

For simplicity, we assume for all mechanisms that each action has Bernoulli outcomes (e.g. Success;

Failure), one of which is preferred by the principal. As long as probabilistic forecasts for the outcomes allow to generate preference rankings over the actions, the outcome space can be extended to finite, mutually exclusive sets, as has been considered in existing research [26]. Moreover, signals are assumed to be binary (e.g. 0 or 1), and the probability of receiving a particular signal for an action depends on the Bernoulli outcome for that action. However, in principle more complex signals can be considered as well.

### 4.3.1 Principal with Own Signal

The principal aims to make an informed decision between alternative actions, and has a signal about the Bernoulli outcome for one of the actions. The principal engages with a group of agents who also have independent signals about an action. To make the best possible decision, the principal needs to incentivise agents to reveal their information and aggregate the elicited information to select an action. Instead of using the observed outcome to reward the agents for accurate forecasts, as is done in decision markets and would require a stochastic decision rule, she uses her own signal as a proxy. Rather than eliciting probabilistic forecasts for the Bernoulli outcomes for each action, agents are incentivised to provide forecasts for her signal. Because the outcome from the action is not required to score the agents, this allows for deterministically selecting an action.

Similar to prediction and decision markets, agents report a set of probabilities (one probability for the signal being 1 for each action) in sequential order. The probabilistic reports depend on an agent's signal and the reports from the previous agent in sequence. The first agent uses uninformative odds as prior (i.e., two 50/50s for a two-action scenario), or a common prior. After the last agent makes the reports, the principal will use a strictly proper scoring rule and her signal as a proxy for the corresponding action's outcome to evaluate all probabilistic reports. As is done in properly incentivised prediction markets and decision markets, the principal assigns a reward to the evaluated report that is equal to the difference between the strictly proper score of the reports and the previous reports to ensure efficiency in assigning rewards [1], [2]. This mechanism resembles peer prediction, except that the peer evaluation uses the principal's signal and the principal is assumed to resolve the markets truthfully [62].

Evaluating with a proxy guarantees the incentive combability for agents as their rewards do not depend on the principal's decision. The principal can therefore use a deterministic decision rule that maps her signal and the aggregated reports from the last agent to an action that is predicted to have the highest chance of achieving an outcome desired by the principal. The best deterministic decision

rule can be inferred by the principal if all priors and conditional probabilities are known, and can also be learned, as is illustrated in the simulations of the contextual bandit problem, which are detailed in Section 4.4.

### 4.3.2 Principal with One Agent Acting as an Advisor

Previous work has shown that offering a single agent a portion of the principal's reward provides a simple way to align the agent's interest with the principal's interest [6]. Such an alignment allows the principal to deterministically select an action following the agent's advice [26]. However, how to provide incentives to multiple advisors who might, based on their signals, provide conflicting advice is an open question.

Because the principal requires only one single signal to elicit all other signals, in cases where the principal does not have access to her own signal, she can incentivise one agent to act as an 'advisor'. The principal then elicits the advisor's signal, and uses this signal as a proxy for the ground truth. Note that while in [26], the advisor recommends an action directly according to the received signal, in our mechanism, the truthful announcement of the advisor's signal is required. Since the advisor's and principal's interests are aligned, the advisor can be expected to truthfully provide the signal. The other agents, who are again assumed to have received independent signals, are incentivised to make forecasts about the advisor's signal. Similar to the procedure described in Section 4.3.1, the principal then uses these forecasts to deterministically select an action. Note that for this mechanism to function, the advisor cannot be allowed to participate, or collude with the other agents, in the incentivised forecasting.

### 4.3.3 Principal with Two Agents Evaluated by Peer Prediction

The mechanism proposed in the previous section lifts the assumption that the principal has an iid signal, similar to the agents. However, to incentivise the advisor, the principal requires the outcome of the selected action to materialise. The mechanism will be more flexible if the iid signals can be acquired without verification, as the outcome of actions may take a long time to materialise. There is substantial research about peer prediction mechanisms eliciting information without verification [61]–[66], [68], [69].

The setting of the principal and agents follows [10], in that the agents, but not the principal, have access to iid signals. However, in the third variation of the proposed mechanism, we separate two agents and refer to them as peers. Although any peer mechanism where truthfully announcing

signals is a Nash equilibrium satisfies our requirements, the mechanism with the smallest possible number of peers is best suited because peer signals cannot be aggregated in the market. We use the mechanism proposed by [64] as an example to expand our mechanism with advantages inherited from peer prediction mechanisms. At the procedure's beginning, agents and peers sample the signals. The peers are required to announce the signal to the principal. The principal will evaluate a peer's announcement with the other peer's announcement, and will reward agreement. However, this does not prevent agents from providing uninformative announcements, such as always making identical announcements regardless of the actual signal. We, therefore, follow [64] and resolve this problem by requiring peers to participate in multiple prior similar tasks to compute a statistic to penalise blind agreements. In our problem, the multi-task requirement can be implemented with a memory of the announcements in previous steps. Again, after the principal retrieves the announcement, which is an iid signal guaranteed by peer prediction mechanisms, the procedure follows Section 4.3.1 using the signal of one of the peers as a proxy. Note that the principal must ensure that the peers are not colluding with each other, or with the agents who forecast one of their signals.

## 4.4 Simulations for Multi-agent Bandit Learning

In this section, we describe simulations of multi-agent learning based on the mechanism described in Section 4.3.1. We model the decision-making mechanism as a multi-agent contextual bandit problem with Bernoulli outcomes. A principal aims to decide among several alternative actions, while signals about the quality of the actions are distributed among a group of self-interested agents. Agents and the principal receive independent and identical distributed (iid) signals for one of the actions, this is in contrast to the simulations of multi-agent learning with decision markets as discussed in Chapter 3, where the principal does not have her own signal. The agents sequentially forecast the principal's signal, conditional on their own signals, and they step-by-step aggregate the available information. After the last agent makes the final forecast, the principal scores all forecasts with a strictly proper scoring rule based on her proxy (i.e., her own signal). Once the principal has learned the correct mapping, she deterministically maps her signal and the final reports to an action.

We model each agent as a contextual bandit problem with a continuous action space. For each agent, the input constitutes of the signals and the reports from the previous agent. (The first agent uses even odds, or a common prior as input.) The agents will receive a score from the principal that correlates with their contribution to predicting the principal's signal and perform a learning

algorithm with it. Unlike in previous work [10] the principal is here assumed to learn with policy gradient methods as well, to use her own signal and the forecasts for her own signal to select one of the available actions. This is modelled as contextual bandit problems with discreet action space. Afterwards, the principal observes the Bernoulli outcome of the selected action and updates the policy accordingly.

### 4.4.1 Problem Setup and Notation

The problem setup of this paper generally follows the work by Wang and Pfeiffer [10] with several key differences. The multi-agent contextual bandit problem arises repeatedly and agents can learn from the score from previous rounds. We denote the time step as $T \in \{1, 2, \ldots, n\}$. At the beginning of each time step $T$, the principal $P$ faces a finite and discrete action set and selects an action $A \in \{1, 2, \ldots, k\}$ from it. The outcome space is a $k$ dimensional Bernoulli vector $\Omega \in \{1, 0\}^k$ and the outcome of arm $A$ is denoted as $\Omega_{(A)} \in \{1, 0\}$. The principal desires the actions $A$ with outcome $\Omega_{(A)} = 1$. Each agent $E \in \{1, 2, \ldots, m\}$ and the principal receive iid signals about one of the actions $A$. The signals are denoted as $D_{(P,A)}$ for the principal and $D_{(E,A)}$ for agent $E$. The signals are Bernoulli variables, i.e., $D_{(P,A)} \in \{1, 0\}$, and sampled with a stationary probability distribution according to the outcome type of an action. If outcome $\Omega_{(A)}$ for action $A$ is 1, the iid signal $D_{(P,A)}$ or $D_{(E,A)}$ for that action is 1 at probability 2/3 and 0 at probability 1/3. If outcome $\Omega_{(A)}$ for action $A$ is 0, the iid signal $D_{(P,A)}$ or $D_{(E,A)}$ for that action is 1 at probability 1/3 and 0 at probability 2/3. $D_{(P,A)}$ plays the role of the proxy. It is statistically correlated with the outcome for action $A$, and is observable to the principal, who can reveal it to the agents. The agents bet on the value of the proxy. The action space of agents is therefore a multi-dimensional real number $\mathbb{R}^k$, of which $k$ is the cardinality of the principal's action set. We refer to an agent's actions as reports, which can be in log-odds or in probabilistic format. We denote the probabilistic reports of agent $E$ as $Pr_{(E)} \in [0, 1]^k$ and the report for a certain principal's action $A$ as $Pr_{(E,A)}$, which is the probability that the principal receives a $D_{(P,A)} = 1$ signal about the action $A$. We denote the log-odds report of agent $E$ as $X_{(E,A)}$.

Agents sequentially make reports to forecast the principal's signal [2], after both the principal and agents receive signals. Agent $E$ receives reports $Pr_{(E-1)}$ from the previous agent $E-1$ in the sequence. Both $Pr_{(E-1)}$ and signals $D_{(E)}$ composite the contextual vector that we denote as $C_{(E)}$. The first agent uses even odds instead of received reports. Agent $E$ maintains a policy parameter matrix $\Theta_{(E)}$ that maps the contextual vector to posterior reports $Pr_{(E)}$. The posterior reports $Pr_{(E)}$ are provided to the principal for evaluation, and become part of the subsequent agent's contextual vector. The

procedure repeats until the last agent $E = m$ in the sequence provides the final report $Pr_{(m)}$ to the principal. The ideal final report is essentially an aggregated forecast of the principal's signal based on all the signals from agents.

In the simulations, we set $k = 2$. Agents keep a $6 \times 2$ policy parameter matrix with random initialisation of weights. The contextual information vector $C_{(E)}$ is multiplied with the matrix of learning parameters as follows:

$$\begin{pmatrix} c_{r1} \\ c_{b1} \\ c_{p1} \\ c_{r2} \\ c_{b2} \\ c_{p2} \end{pmatrix}^{\intercal} \times \begin{pmatrix} \theta_{r1}^{(1)} & \theta_{r1}^{(2)} \\ \theta_{b1}^{(1)} & \theta_{b1}^{(2)} \\ \theta_{p1}^{(1)} & \theta_{p1}^{(2)} \\ \theta_{r2}^{(1)} & \theta_{r2}^{(2)} \\ \theta_{b2}^{(1)} & \theta_{b2}^{(2)} \\ \theta_{p2}^{(1)} & \theta_{p2}^{(2)} \end{pmatrix} = \left(\mu_{(E,1)}, \mu_{(E,2)}\right) \tag{4.1}$$

The contextual vector $C_{(E)}$ has six elements. The element $c_{r1}$ is set to 1 if the agent receives signal 1 for action 1 (i.e., $D_{(E,1)} = 1$); $c_{b1}$ is set to 1 if $D_{(E,1)} = 0$, and similarly $c_{r2}$ and $c_{b2}$ are set to 1 when the received signal is signal $D_{(E,2)} = 1$, and $D_{(E,2)} = 0$, respectively. Above elements remain 0 when there is no signal. The elements $c_{p1}$ and $c_{p2}$ are the prior log-odds transformed probabilistic reports for the first and second actions. The result is a pair of parameters $\left(\mu_{(E,1)},\ \mu_{(E,2)}\right)$, where $\mu_{(E,1)}$ is used to sample a report for action 1, and $\mu_{(E,2)}$ is used to sample a report for action 2. The actual log-odds reports $(X_{(E,1)}, X_{(E,2)})$ will be sampled from a normal distribution with the above parameters $\left(\mu_{(E,1)},\ \mu_{(E,2)}\right)$ as means, and a fixed variance to estimate gradients for learning. The probabilistic reports $(Pr_{(E,1)}, Pr_{(E,2)})$ will be converted from log-odds reports $(X_{(E,1)}, X_{(E,2)})$.

The principal evaluates the probabilistic reports of agent $E$ using a proper scoring rule function and her signal described in [25]:

$$s : Pr_{(E,A)} \times D_{(P,A)} \to \mathbb{R} \tag{4.2}$$

The reward $R_{(E)}$ for an agent $E$ is calculated by $s\left(Pr_{(E,A)}, D_{(P,A)}\right) - s\left(Pr_{(E-1,A)}, D_{(P,A)}\right)$. As the latter part of the reward is not dependent on the agent $E$; therefore, at time step $T$, agent $E$ optimises the reward by updating the policy parameters with:

$$\Theta_{(E,T+1)} = \Theta_{(E,T)} + \alpha G_{(E,T)} \tag{4.3}$$

where $G_{(E,T)}$ is a matrix of approximated partial derivatives of expected score $\mathbb{E}[s(Pr_{(E,A)}, D_{(P,A)})]$

with respect to policy parameters. Because the principal uses a proper score function, the expected score maximises when agent $E$ learns to report $Pr_{(E,A)} = Pr\left(D_{(P,A)} \middle| D_{(E,A)}, Pr_{(E-1,A)}\right)$. The scores for agents depend on agents' reports and the principal's signal and, therefore, are decoupled from the principal's decision-making. The principal can resolve the scores for agents immediately after the aggregated report is available. The scores incentivise the agents to learn to accurately report probabilistic reports for the proxy.

In the simulations presented here, the principal is a computational model as well, and maintains a policy parameter matrix $\Theta_{(P)}$. In the simulation, the structure of the policy matrix is the same as the agents' policy matrix. We denote the contextual vector of the principal as $C_{(P)}$, which constitutes the principal's signal $D_{(P)}$ and the reports from the last agent. While the agent uses the output of the multiplication of the context vector and the weight matrix to sample a report from a normal distribution, the principal uses the output to sample an action using the softmax function. The result $\left(\mu_{(P,1)},\ \mu_{(P,2)}\right)$ can be considered as the preference of an action relative to the other actions. Using the soft-max function, the probabilities of the two actions available in the simulations are given by:

$$\begin{cases} \Phi_1 &= \frac{\exp\left(\mu_{(P,1)}\right)}{\exp\left(\mu_{(P,1)}\right)+\exp\left(\mu_{(P,2)}\right)} \\ \Phi_2 &= \frac{\exp\left(\mu_{(P,2)}\right)}{\exp\left(\mu_{(P,1)}\right)+\exp\left(\mu_{(P,2)}\right)} \end{cases} \tag{4.4}$$

The principal sample the action from the distribution $\Phi_{(P)} = \{\Phi_1, \Phi_2\}$ and executes it. At the time step $T$, after observing the outcome of the executed action, the principal will update its policy parameters to maximise the expectation of the desired outcome to materialise:

$$\Theta_{(P,T+1)} = \Theta_{(P,T)} + \alpha G_{(P,T)} \tag{4.5}$$

$G_{(P,T)}$ is a matrix of approximated partial derivatives of policy parameters with respect to the expected rewards. In other words, the principal learns to interpret the signal $D_{(P)}$ and the reports from the last agent in sequence $Pr_{(m)}$ into an action distribution to select the action $A$ that is most likely to lead to an outcome $\Omega_A = 1$. Note that the computational model samples the action for approximating gradients, which serves for the learning purpose in our simulations. A principal who knows (or has learned) the relation between final report, her own signal, and outcomes, can use a deterministically select action without changing the incentives for the agents.

In the simulation, we use a Brier scoring rule. Any proper scoring rules will have consistent results here [20], [25]. Both the principal and the agents use an experience replay buffer technique to speed

| Agent | Action | Signal | Prior | Posterior | Decision | Score |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | (0.5,0.5) | (0.67,0.5) | | − |
| 2 | 2 | 1 | (0.67,0.5) | (0.67,0.67) | | 0 |
| Principal | 1 | 0 | (0.67,0.67) | | Action 2 | + |

Table 4.1: **Example decision-making process with two agents**. In this example, the outcome for action 1 is 0 and for action 2 is 1. Agent $E = 1$ receives a signal $D_{(1,1)} = 1$ for action 1 and agent $E = 2$ receives a signal $D_{(2,2)} = 1$ for action 2. The principal $P$ receives a signal $D_{(P,1)} = 0$ for action 1. Assuming the two agents and the principal are well-trained. Agent $E = 1$ updates the prior $(0.5, 0.5)$ to generate a posterior report $(0.67, 0.5)$. Agent $E = 2$ does not change the report for action 1 but increases the report for action 2 and generates the posterior report $(0.67, 0.67)$. The principal decides to execute action $A = 2$ because her signal breaks the tie in favour of action $A = 2$. This is a correct decision and that is rewarded by $\Omega_{(2)=1}$. While the expected payoff of agent 1 is positive in this case, agent 1 receives a negative score because it makes the reports for the principal's signal less accurate. Agent 2 receives a score of 0 because the principal receives a signal for action $A = 1$ rather than action $A = 2$.

up the training process [55], [56]. For agent $E$, a tuple $\left(C_{(E,T)}, \mu_{(E,T)}, X_{(E,T)}, R_{(E,T)}\right)$ consists of the contextual vector, the mean log-odds, the actual reports and the reward at time step $T$ is called a piece of experience. For the principal $P$, the experience tuple $\left(C_{(P,T)}, \Phi_{(P,T)}, \Omega_{(A,T)}, A_T\right)$ consists of the contextual vector, the action distribution and the materialised Bernoulli outcome (which is essentially a 1-0 reward). In each time step, the principal and agents randomly draw a mini-batch of pieces of experience from the experience reply buffer. We assume the experience tuple at time step $I$ is within the mini-batch sampled at time step $T$. The gradient for the principal can be obtained by:

$$G_{(P,I)} = \begin{cases} C_{(P,I)} \left(\Omega_{(A,I)} - B_{(P,I)}\right) \left(1 - \Phi_{(P,I)} \left(A_I\right)\right) \\ -C_{(P,I)} \left(\Omega_{(A,I)} - B_{(P,I)}\right) \left(\Phi_{(P,I)} \left(a\right)\right), \forall a \neq A_I \end{cases} \tag{4.6}$$

The gradient for the agents can be obtained by:

$$G_{(E,I)} = C_{(E,I)} \left(R_{(E,I)} - B_{(E,I)}\right) \frac{X_{(E,I)} - \mu_{(E,I)}}{\sigma^2} \tag{4.7}$$

In our simulation, $C_{(P,I)}$ and $C_{(E,I)}$ have $6 \times 1$ dimension and $\Phi_{(P,I)}$, $X_{(E,I)}$ and $\mu_{(E,I)}$ have $1 \times 2$ dimension. Therefore, the Gradients have the same dimensionality as the policy matrix. We use a different time step notation $I$ to emphasise that at time step $T$, the training does not necessarily use the experience from time step $T$. In both equations 4.6 and 4.7, $B_{(P,I)}$ and $B_{(E,I)}$ are two baselines functions that do not vary with the action. The baseline function should have no effect on the expectation but influences variance and will speed up the training process. A popular choice of the baseline function is a running average of the reward. Finally, both the principal and agents will learn

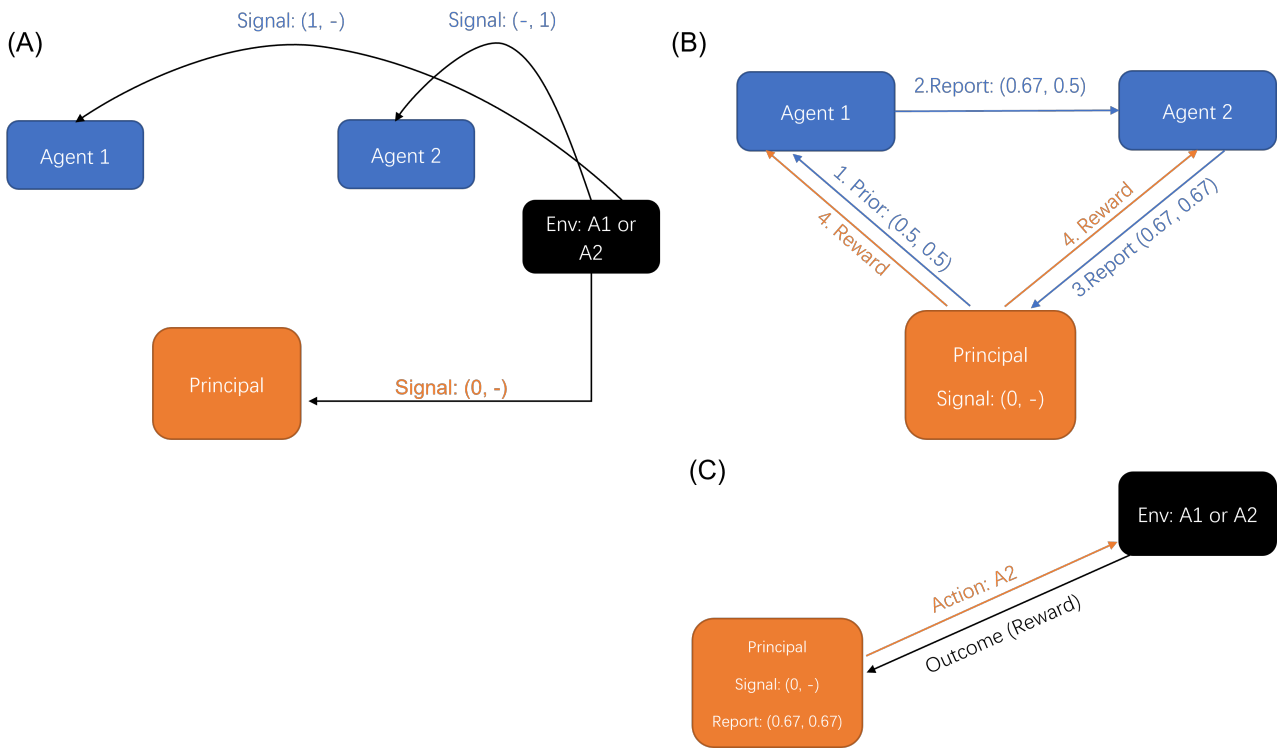


Figure 4.1: **Example decision-making process with two agents**. The three panels show the progress of the system. In Panel (A), the environment presents two actions. Agent 1 receives a signal (1, -), where '1' indicates a positive signal for action 1, and '-' signifies the absence of information regarding action 2. Agent 2 and principal also receive signals as indicated. Panel (B) illustrates the sequence of reporting steps. Firstly, the principal shares the initial prior with the first agent. Agent 1 then adjusts the prior based on the received signal and subsequently transmits it to Agent 2. Agent 2 follows by making its adjustments and passing the final report to the principal. The principal checks the report against the signal she received and rewards the agents accordingly. Panel (C) shows, finally, the principal maps her own signal and the final report to an action (in this case A2) and then executes it. The outcome of the execution is observed subsequently.

with the average of the mini-batch gradients.

In this work, we set a Bayesian updating model as a benchmark. At the first stage, the Bayesian model receives all the same signals $D_{(E)}$ as the agents. The Bayesian model generates a posterior according to the agents' signals $\widehat{Pr_{(A)}} = Pr\left(D_{(P,A)} \middle| D_{(1,A)}, \ldots, D_{(m,A)}\right)$. As once the accessibility of distributed and proprietary signals is allowed, the problem become trivial. Therefore, we consider the Bayesian update posterior as the ideal report that the agents can generate. We evaluate the mean squared error between the aggregated report and the Bayesian updated posterior:

$$Er = \sum_{A} \left( Pr_{(m,A)} - \widehat{Pr_{(A)}} \right)^2 \tag{4.8}$$

To evaluate performance in decision-making, we use a Bayesian model that will take both the principal's signal and the agents' signals into consideration and compute the posterior to $\widehat{Pr'_{(A)}} =$

$Pr\left(\Omega_{(A)}|D_{(1,A)},\ldots,D_{(m,A)},D_{(P,A)}\right)$. We deterministically select the action A that has the highest chance to have outcome $\Omega_{(A)}=1$ according to $\widehat{Pr'_{(A)}}$, and record the frequency at which this procedure selects an action with outcome 1. For comparison, we also record the average frequency that the principal selected action's outcome to be 1.

### 4.4.2 Simulation Results

This section will show the result of a multi-agent bandit learning system that employs our mechanism for deterministic decision-making problems in the setting we mentioned in the previous section.

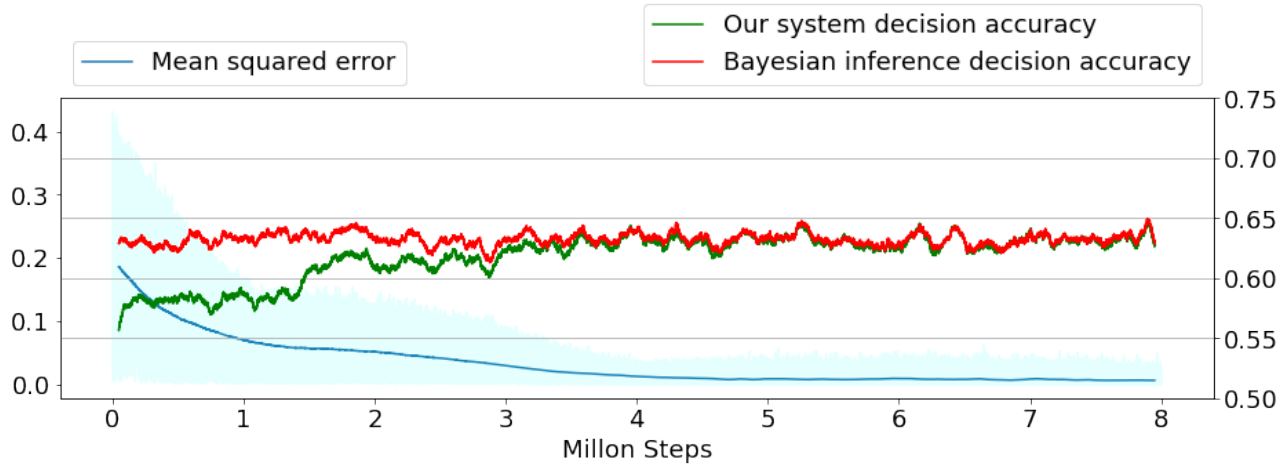


Figure 4.2: **System performance of a three-agent system and the Bayesian inference**. The blue line is the mean squared error $Er$ between the aggregated report $Pr_{(m,A)}$ from the market and a Bayesian inference $\widehat{Pr_{(A)}}$ that uses the same information. The light cyan marks the range of the actual error. The red and green lines are the average times the selected action's outcome that the principal desires.

Figure 4.2 shows that the multi-agent system's error, calculated by equation 4.8, converges after around 5 million steps. After the agents' policy parameters converge, the decision quality of the principal is as good as a Bayesian inference model that can access all the signals. The result suggests that our mechanism achieves the most informed decision-making with a deterministic decision rule.

## 4.5 Conclusion and Discussion

This research proposes a set of closely related mechanisms to address a collective decision-making problem. That is, a principal wants to decide between several actions, but most information about the quality of the actions is distributed and privately owned by self-interested agents. To make an informed decision, the principal needs to elicit the information with incentives and aggregate it in a suitable way for subsequent decision-making. Existing collective decision-making mechanisms

face a dilemma: either decision cannot be made deterministically without triggering manipulative behaviours, or the information cannot be elicited from more than one agent to make a deterministic recommendation [6], [26].

Our mechanisms address information elicitation and aggregation under a deterministic decision rule through the prediction of a principal's proxy. This approach resembles peer prediction mechanisms. In peer prediction, such as scenarios involving experiencing a product and rating its quality, multiple agents (typically more than three) observe signals that are conditional on the type (or quality) of the product. The principal asks the agents to disclose their signals, but the agents do not necessarily truthfully reveal their signal. The principal rewards the agents based on their disclosures. In this scenario, an announcement strategy is considered a best response to the announcements of the other agents if it results in the highest expected payment.

For any given agent, a reference agent will be randomly selected and assigned. The principal then computes a conditional probability based on the agent's announcement and the reference agent's announcement. The principal finally employs a strictly proper scoring rule to assess the probability and determine the payment. Assuming the reference agent reports honestly, the best response for the agent is also to report truthfully. A detailed and formal proof of incentive compatibility in this scenario can be found in [62]. The key difference to our mechanism is that the principal also receives an iid signal, just like the agents, and the principal's signal serves as the reference for all the agents, and as proxy for an outcome that is not necessarily observed. If the agents can trust that the principal will accurately determine the payment based on the signal she receives, then a truthful Nash equilibrium can be achieved. Another significant distinction in our simulations is that the conditional probability is not assumed as prior knowledge and computed, but rather learned through a policy gradient method by the agents. Finally, the principal learns to map her own signal and the aggregated report from agents to conditional probabilities regarding the outcomes of actions, allowing for the application of a deterministic decision rule.

We further expand this mechanism to two additional variants. The first expansion lifts the assumption that the principal requires a proxy herself. The mechanism separates an agent from the rest as an advisor. The principal can elicit the proxy from the advisor by providing a share of her reward to align their interest. The second expansion employs a peer prediction mechanism and separates two agents to predict the signal that the other party possesses. As long as one chooses a proper peer prediction mechanism, the elicited signals are as if the principal owned the information. In both variants, the information of the other agents is aggregated and used to make a deterministic decision. Note the

principal must ensure that collusion does not occur between the advisor and the other agents, or the two peers and the other agents.

We use a multi-agent contextual bandit system to simulate the dynamics of learning for our first mechanism. The result shows that the average frequency of the selected action's outcome desired by the principal using our mechanism is as high as under a Bayesian inference model with identical information. It suggests our mechanism solves the collective decision-making problem with a deterministic decision rule.

There are several future study directions. An important direction is to employ the mechanisms described here to solve a contextual bandit problem that involves multiple agents who are unwilling to share raw signals, i.e., user profiles. Such a scenario fits into a growing research direction of federated learning. Federated learning, unlike the traditional supervised learning model, forbids a centralised party from directly collecting data for learning [73], [74]. Federated bandit problems fall into this category, which train recommendation systems without accessing private user data. Our mechanisms inherently do not require accessing the agents' signals and potentially provide another direction to solve the federated bandit problems.

In the mechanisms presented here, we distinguish between the principal and agents. For many peer decision problems, there is no such distinction. All agents could, in principle, receive signals and get into the position to execute an action. We can expand our mechanism to such a peer decision problem by randomly selecting an agent as the principal at each time step, and the rest of the procedure follows the first mechanism. This mechanism is most useful when the peers have no conflicting preferences for the actions' outcome. For instance, all agents prefer success over failure as the selected action's outcome. When the agents are not picked as the principal, they are self-interested in that they seek to profit from their signals.

We validate our mechanism with self-interested computational agents. It remains to be studied how well the mechanism works for a more complex action and outcome space. It is also worth investigating how well the mechanism works, for instance, with human subjects in a laboratory setting, and how it compares to alternative mechanisms of collective decision-making.

# Chapter 5

# Conclusion

The conclusion chapter summarises the results from Chapters 2-4 and discusses opportunities for future studies. Chapter 2 proposes a design for decision markets with stochastic decision rules that is based on the trading of securities. The design ensures that participants have the identical expected payoff as in Chen, Kash, Ruberry, *et al.*'s scoring rule based design for the same amount of information [1]. The distribution of materialised payoffs, however, may differ. The securities-based decision markets allow more flexibility in adjusting worst-case loss with specially designed contracts. The principal can even outsource its liability by allowing some participants to 'gamble' over which action the principal will select. Such a 'side game' reduces the principal's liability, and the principal can better approximate deterministic decision rules without facing a higher worst-case loss if the specially designed contracts are set to hedge the risk. The securities-based implementation presented in this chapter makes the mechanism more attractive to human forecasters.

Decision-making problems are studied in computer science as well, in particular in the rapidly growing fields of machine learning and AI. A typical framework that studies optimising the decision-making process in a repeated game is the multi-armed bandit problem [9]. Classic multi-armed bandit problems, in which a single agent repeatedly makes a decision over the same action set, are well investigated. Multi-agent bandit problems have gained much attention recently in the context of multi-agent systems, because some problems in real-life are distributed in nature.

Chapter 3 presents a multi-agent bandit system that employs a decision market with a stochastic decision rule to address a collective decision-making problem by learning. The chapter assumes a group of self-interested agents make forecasts based on local and private information for a principal to solve a contextual bandit problem. The principal evaluates the forecasts by a decision market with a stochastic decision rule, and assigns rewards. The agent learns to map the private information to a forecast that

maximises the expectation of the assigned rewards. Based on the nature of the decision market, such a mapping can be accurate or strategic. Our simulations demonstrate that the multi-agent system based on a decision market with a stochastic decision rule can achieve equivalent learning efficiency to a centralised counterpart that can access all the local and private information. Furthermore, I simulate the dynamics of single-agent and three-agent systems with deterministic decision rules. For the single-agent system, I observed the strategic manipulation as described by Othman and Sandholm. The interactions observed in three-agent systems are more complex. I observe a substantial first-mover advantage and surprisingly accurate aggregated forecasts despite the fact that all three agents provide 'manipulative' reports.

Decision markets with stochastic decision rules require the principal to commit to a randomised decision rule, which is not in her interest. Mitigation exists but comes with other limitations, such as higher worst-case loss or more complex design. How to deterministically elicit and aggregate information distributed among a group of self-interested agents remains an open question. Chapter 4 addresses this question with three proxy-based mechanisms. Firstly, I assume that the principal, like the agents, has her own independent and identically distributed signal, which is a piece of information that correlates with the outcome of one of the available actions and serves as a proxy. Instead of requiring the agents to forecast the outcome of each action as is done in Chapter 3, the principal makes the agents forecast the nature of her signal. The principal then evaluates the forecasts with her signal and a strictly proper scoring rule. This guarantees that the agents maximise their expected reward by accurately forecasting conditional on their local and private signals. The principal deterministically maps the forecasts and her own signal to a decision.

In further mechanisms described in Chapter 4, I lift the assumption that the principal must have her own signal: the principal separates one agent from the rest and incentivises it with a partial profit to align the interests. The principal can then retrieve truthful information about the signal possessed by the agent and use it as a proxy to elicit forecasts from the other agents. Alternatively, the principal separates two agents from the other agents and uses a peer prediction mechanism to elicit accurate information about signals possessed by these two agents. I extend the multi-agent bandit learning system described in Chapter 3 to simulate the dynamics under the first mechanism. The simulation result shows that the mechanism allows the system to make decisions as efficiently as a Bayesian inference model with access to all private and local information.

## Limitations and future study

This research has several limitations, and corresponding directions to extend the work.

A substantial limitation of Chapter 2 is the use of stochastic decision rules, which may create a barrier to applications. Because it is not in the interest of the decision maker to make a stochastic choice once the forecasts have been elicited [26]. Therefore, it is crucial for the forecasters to ensure that the principal is committed to such a choice. This can be implemented through a trusted third party or a commitment without trusted third party. A future direction related to Chapter 2 is to explore methods to implement securities based decision markets with smart contracts on blockchain. It has inherent advantages to allow for an irreversible commitment by the principal. The limitation from stochastic decision rules is further addressed in Chapter 4.

In Chapter 3, the artificial models for agents map the contextual vectors to forecasts in a linear manner. As a result, the agents may fail to learn complex strategies, as this requires non-linear mapping as is the case for decision markets with deterministic decision rules. Another potential limitation is that the gradient methods only search locally. Therefore, learning may not find global optima when the problem is non-convex or even non-smooth. This is because the agents may be stuck in the local optima, when learning with gradient-based methods [75]. Multi-agent reinforcement learning to find the global Nash Equilibrium is challenging for agents in such a situation, because learning the global NE often requires the agents to explore learning parameters with lower expected scores first [12], [76]. One can extend the work with a more sophisticated exploration strategy to help the agents find the global NE.

In Chapter 4, the mechanism that separates two agents and uses a peer prediction mechanism to retrieve the truthful signals, bears the inherent limitation from peer prediction mechanisms. That is truth-telling is not a strict global NE point as so called 'permutation strategies' can always be another global NE point[71], [77]. The 'permutation strategies' mean the agents always make the opposite report given the signals, i.e., agents lower the probability after they receive type 1 signal instead of increasing or vice versa. The limitation becomes more severe when multiple agents are learning simultaneously.

Multi-agent learning is a rapidly developing field, and a recent development is federated learning. Federated bandit learning is a framework for solving bandit problems such that the contextual information is from multiple sources and remains hidden while training the bandit models. Several

solutions with different focuses have been proposed[41], [43], [44]. It is worth studying how decision-market based designs, and designs with proxy forecasting can help to make good decisions when user data sharing is restricted, but predictions can be shared.

Chapters 3 and 4 are based on economic mechanisms designed for humans. Note that, in economics, humans are assumed to act risk-neutral and fully rational even though literature on human decision making in real-world shows that this is often not the case [78]. Therefore, the multi-agent systems described in Chapters 3 and 4 can work as well in a human-robot-mixed scenario. An interesting direction would be designing experiments that allow humans and well-trained agents to decide jointly. Such experiments would help investigating the impact of 'irrational' human behaviour on hybrid decision making.

# Bibliography


[1] Y. Chen, I. Kash, M. Ruberry, and V. Shnayder, "Decision Markets with Good Incentives," in *Internet and Network Economics. WINE 2011. Lecture Notes in Computer Science, vol 7090.*, N. Chen, E. Elkind, and E. Koutsoupias, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 72–83.

[2] R. D. Hanson, "Combinatorial information market design," *Information Systems Frontiers*, vol. 5, no. 1, pp. 107–119, 2003.

[3] J. Wolfers and E. Zitzewitz, "Interpreting Prediction Market Prices as Probabilities," National Bureau of Economic Research, Tech. Rep. 12200, 2006.

[4] D. M. Pennock and R. Sami, "Computational aspects of prediction markets," in *Algorithmic Game Theory*, vol. 9780521872, School of Engineering and Applied Sciences, Harvard University, United States: Cambridge University Press, 2007, pp. 651–676.

[5] R. D. Hanson, "Decision Markets," *IEEE Intelligent Systems*, vol. 14, no. 3, pp. 16–19, 1999.

[6] A. Othman and T. Sandholm, "Decision rules and decision markets," in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems*, vol. 1, 2010, pp. 625–632.

[7] Y. Chen and D. M. Pennock, "A utility framework for bounded-loss market makers," in *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence*, 2007, pp. 49–56.

[8] W. Wang and T. Pfeiffer, "Securities Based Decision Markets," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13170 LNAI, pp. 79–92, 2022.

[9] T. Lattimore and C. Szepesvári, *Bandit Algorithms*. Cambridge University Press, 2020.

[10] W. Wang and T. Pfeiffer, "Decision Market Based Learning For Multi-agent Contextual Bandit Problems," *arXiv preprint*, arXiv:2212.00271, 2022.

[11] A. Charpentier, R. Élie, and C. Remlinger, "Reinforcement Learning in Economics and Finance," *Computational Economics*, vol. 62, no. 1, pp. 425–462, 2021.

[12] A. Nowé, P. Vrancx, and Y.-M. De Hauwere, "Game Theory and Multi-agent Reinforcement Learning," in *Reinforcement Learning: State-of-the-Art*, M. Wiering and M. van Otterlo, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 441–470.

[13] Y. Chen and I. A. Kash, "Information elicitation for decision making," in *10th International Conference on Autonomous Agents and Multiagent Systems*, vol. 1, International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 2011, pp. 161–168.

[14] C. R. Plott, "Markets as Information Gathering Tools," *Southern Economic Journal*, vol. 67, no. 1, pp. 2–15, 2000.

[15] Y. Chen, S. Dimitrov, R. Sami, *et al.*, "Gaming Prediction Markets: Equilibrium Strategies with a Market Maker," *Algorithmica*, vol. 58, no. 4, pp. 930–969, 2010.

[16] A. Dreber, T. Pfeiffer, J. Almenberg, *et al.*, "Using prediction markets to estimate the reproducibility of scientific research," *Proceedings of the National Academy of Sciences*, vol. 112, no. 50, pp. 15 343–15 347, 2015.

[17] C. R. Plott, J. Wit, and W. C. Yang, "Parimutuel betting markets as information aggregation devices: Experimental results," *Economic Theory*, vol. 22, no. 2, pp. 311–351, 2003.

[18] C. F. Manski, "Interpreting the predictions of prediction markets," *Economics Letters*, vol. 91, no. 3, pp. 425–429, 2006.

[19] G. Tziralis and I. Tatsiopoulos, "Prediction markets: An extended literature review," *The journal of prediction markets*, vol. 1, no. 1, pp. 75–91, 2007.

[20] R. D. Hanson, "Logarithmic market scoring rules for modular combinatorial information aggregation," *The Journal of Prediction Markets*, vol. 1, no. 1, pp. 3–15, 2007.

[21] J. E. Berg, F. D. Nelson, and T. A. Rietz, "Prediction market accuracy in the long run," *International Journal of Forecasting*, vol. 24, no. 2, pp. 285–300, 2008.

[22] K. J. Arrow, R. Forsythe, M. Gorham, *et al.*, "The Promise of Prediction Markets," *Science*, vol. 320, no. 5878, pp. 877–878, 2008.

[23] C. R. Plott and K.-Y. Chen, "Information aggregation mechanisms: Concept, design and implementation for a sales forecasting problem," California Institute of Technology, Division of the Humanities and Social Sciences, No.1131 Working paper, Tech. Rep., 2002.

[24] J. E. Bickel, "Some comparisons among quadratic, spherical, and logarithmic scoring rules," *Decision Analysis*, vol. 4, no. 2, pp. 49–65, 2007.

[25] T. Gneiting and A. E. Raftery, "Strictly Proper Scoring Rules, Prediction, and Estimation," *Journal of the American Statistical Association*, vol. 102, no. 477, pp. 359–378, 2007.

[26] Y. Chen, I. A. Kash, M. Ruberry, and V. Shnayder, "Eliciting predictions and recommendations for decision making," *ACM Transactions on Economics and Computation*, vol. 2, no. 2, pp. 1–27, 2014.

[27] C. E. Boutilier, "Eliciting forecasts from self-interested experts: Scoring rules for decision makers," in *11th International Conference on Autonomous Agents and Multiagent Systems 2012, AAMAS 2012: Innovative Applications Track*, vol. 2, Department of Computer Science, University of Toronto, Toronto, Canada: International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 2012, pp. 1008–1015.

[28] P. H. Garthwaite, J. B. Kadane, and A. O'Hagan, "Statistical methods for eliciting probability distributions," *Journal of the American Statistical Association*, vol. 100, no. 470, pp. 680–701, 2005.

[29] Y. Chen and J. W. Vaughan, "A new understanding of prediction markets via no-regret learning," in *Proceedings of the ACM Conference on Electronic Commerce*, 2010, pp. 189–198.

[30] C. Oesterheld and V. Conitzer, "Decision Scoring Rules," in *16th International Conference on Web and Internet Economics*, 2020, p. 468.

[31] F. Teschner, D. Rothschild, and H. Gimpel, "Manipulation in Conditional Decision Markets," *Group Decision and Negotiation*, vol. 26, no. 5, pp. 953–971, 2017.

[32] L. Jian and R. Sami, "Aggregation and manipulation in prediction markets: effects of trading mechanism and information distribution," *Management Science*, vol. 58, no. 1, pp. 123–140, 2012.

[33] L. Busoniu, R. Babuska, and B. De Schutter, "A Comprehensive Survey of Multiagent Reinforcement Learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 156–172, 2008.

[34] R. Lowe, Y. I. WU, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," in *Advances in Neural Information*

*Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., vol. 30, Curran Associates, Inc., 2017, pp. 6379–6390.

[35] M. L. Littman, "Markov Games as a Framework for Multi-Agent Reinforcement Learning," in *Proceedings of the Eleventh International Conference on International Conference on Machine Learning*, ser. ICML'94, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, pp. 157–163.

[36] D. Silver, A. Huang, C. J. Maddison, *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[37] D. Silver, J. Schrittwieser, K. Simonyan, *et al.*, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[38] B. Baker, I. Kanitscheider, T. Markov, *et al.*, "Emergent Tool Use From Multi-Agent Autocurricula," in *International Conference on Learning Representations*, 2020.

[39] J. Z. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel, "Multi-agent reinforcement learning in sequential social dilemmas," in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 1, 2017, pp. 464–473.

[40] T. Rashid, M. Samvelyan, C. Schroeder, G. Farquhar, J. Foerster, and S. Whiteson, "QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning," in *Proceedings of the 35th International Conference on Machine Learning*, J. Dy and A. Krause, Eds., ser. Proceedings of Machine Learning Research, vol. 80, PMLR, 2018, pp. 4295–4304.

[41] T. Li, L. Song, and C. Fragouli, "Federated Recommendation System via Differential Privacy," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 2592–2597.

[42] J. Qi, Q. Zhou, L. Lei, and K. Zheng, "Federated reinforcement learning: techniques, applications, and open challenges," *Intelligence & Robotics*, pp. 1–39, 2021.

[43] C. Shi and C. Shen, "Federated Multi-Armed Bandits," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, pp. 9603–9611.

[44] Z. Zhu, J. Zhu, J. Liu, and Y. Liu, *Federated Bandit: A Gossiping Approach*. Association for Computing Machinery, 2021, vol. 49, pp. 3–4.

[45] M. Tan, "Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents," in *Readings in Agents*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, pp. 487–494.

[46] A. Tampuu, T. Matiisen, D. Kodelja, *et al.*, "Multiagent cooperation and competition with deep reinforcement learning," *PLOS ONE*, vol. 12, no. 4, pp. 1–15, 2017.

[47] R. Agrawal, "The continuum-armed bandit problem," *SIAM journal on control and optimization*, vol. 33, no. 6, pp. 1926–1951, 1995.

[48] A. G. Barto and P. Anandan, "Pattern-recognizing stochastic learning automata," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-15, no. 3, pp. 360–375, 1985.

[49] K. Liu and Q. Zhao, "Distributed Learning in Multi-Armed Bandit With Multiple Players," *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5667–5681, 2010.

[50] D. Martinez-Rubio, V. Kanade, and P. Rebeschini, "Decentralized Cooperative Stochastic Bandits," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 4529–4540.

[51] I. Bistritz, T. Baharav, A. Leshem, and N. Bambos, "My fair bandit: Distributed learning of max-min fairness with multi-player bandits," in *International Conference on Machine Learning*, PMLR, 2020, pp. 930–940.

[52] R. Kleinberg, "Nearly tight bounds for the continuum-armed bandit problem," in *Proceedings of the 17th International Conference on Neural Information Processing Systems*, 2004, pp. 697–704.

[53] R. J. Williams, "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning," *Machine Learning*, vol. 8, no. 3–4, pp. 229–256, 1992.

[54] R. S. Sutton and A. G. Barto, *Reinforcement learning, Second Edition: An Introduction*, Second. MIT press, 2018.

[55] L.-j. Lin, "Reinforcement Learning for Robots Using Neural Networks," Ph.D. dissertation, Carnegie Mellon University, UMI Order No. GAX93-22750, 1993.

[56] V. Mnih, K. Kavukcuoglu, D. Silver, *et al.*, "Playing Atari with Deep Reinforcement Learning," *arXiv preprint*, arXiv:1312.5602, 2013.

[57] W. Wang and T. Pfeiffer, "Proxy Forecasting to Avoid Stochastic Decision Rules in Decision Markets," *arXiv preprint*, arXiv:2303.10857, 2023.

[58] M. Chakraborty and S. Das, "Trading on a rigged game: Outcome manipulation in prediction markets," *IJCAI International Joint Conference on Artificial Intelligence*, vol. 2016-Janua, pp. 158–164, 2016.

[59] R. Freeman, S. Lahaie, and D. M. Pennock, "Crowdsourced outcome determination in prediction markets," *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, pp. 523–529, 2017.

[60] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[61] D. Prelec, "A Bayesian truth Serum for subjective data," *Science*, vol. 306, no. 5695, pp. 462–466, 2004.

[62] N. Miller, P. Resnick, and R. Zeckhauser, "Eliciting informative feedback: The peer-prediction method," *Management Science*, vol. 51, no. 9, pp. 1359–1373, 2005.

[63] R. Jurca and B. Faltings, "Mechanisms for making crowds truthful," *Journal of Artificial Intelligence Research*, vol. 34, pp. 209–253, 2009.

[64] A. Dasgupta and A. Ghosh, "Crowdsourced Judgement Elicitation with Endogenous Proficiency," in *Proceedings of the 22nd International Conference on World Wide Web*, ser. WWW '13, New York, NY, USA: Association for Computing Machinery, 2013, pp. 319–330.

[65] Y. Kong and G. Schoenebeck, "An information theoretic framework for designing information elicitation mechanisms that reward truth-telling," *ACM Transactions on Economics and Computation*, vol. 7, no. 1, pp. 1–33, 2019.

[66] J. Witkowski and D. C. Parkes, "Peer prediction without a common prior," in *Proceedings of the ACM Conference on Electronic Commerce*, vol. 1, 2012, pp. 964–981.

[67] J. Witkowski and D. C. Parkes, "A robust Bayesian truth serum for small populations," in *Proceedings of the National Conference on Artificial Intelligence*, vol. 2, 2012, pp. 1492–1498.

[68] P. Zhang and Y. Chen, "Elicitability and knowledge-free elicitation with peer prediction," *13th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2014*, vol. 1, pp. 245–252, 2014.

[69] Y. Liu and Y. Chen, "Machine-Learning Aided Peer Prediction," in *Proceedings of the 2017 ACM Conference on Economics and Computation*, ser. EC '17, New York, NY, USA: Association for Computing Machinery, 2017, pp. 63–80.

[70] Y. Kong and G. Schoenebeck, "Water from two rocks: Maximizing the mutual information," in *ACM EC 2018 - Proceedings of the 2018 ACM Conference on Economics and Computation*, ser. EC '18, New York, NY, USA: Association for Computing Machinery, 2018, pp. 177–194.

[71] Y. Liu, J. Wang, and Y. Chen, "Surrogate Scoring Rules," *EC 2020 - Proceedings of the 21st ACM Conference on Economics and Computation*, pp. 853–871, 2020.

[72] J. Witkowski, P. Atanasov, L. H. Ungar, and A. Krause, "Proper proxy scoring rules," in *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, 2017, pp. 743–749.

[73] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated Learning: Strategies for Improving Communication Efficiency," *arXiv preprint*, arXiv:1610.05492, 2016.

[74] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1–19, 2019.

[75] E. V. Mazumdar, M. I. Jordan, and S. S. Sastry, "On Finding Local Nash Equilibria (and Only Local Nash Equilibria) in Zero-Sum Games," *arXiv preprint*, arXiv:1901.00838, 2019.

[76] Y. Yang and J. Wang, "An Overview of Multi-Agent Reinforcement Learning from Game Theoretical Perspective," *arXiv preprint*, arXiv:2011.00583, 2020.

[77] Y. Kong, "Dominantly truthful multi-task peer prediction with a constant number of tasks," *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, vol. 2020-Janua, pp. 2398–2411, 2020.

[78] D. Kahneman, *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011.

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

# STATEMENT OF CONTRIBUTION
# DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the student and the student's main supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the student's contribution as indicated below in the Statement of Originality.

| | |
|---|---|
| Student name: | Wenlong Wang |
| Name and title of main supervisor: | Professor Thomas Pfeiffer |
| In which chapter is the manuscript/published work? | Chapter 2 |
| What percentage of the manuscript/published work was contributed by the student? | 50% |

Describe the contribution that the student has made to the manuscript/published work:

The student is the sole contributor to this work, besides his supervisor.

Please select one of the following three options:

- (●) **The manuscript/published work is published or in press**

  Please provide the full reference of the research output:

  Distributed Artificial Intelligence. DAI 2021. Lecture Notes in Computer Science(including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol 13170, pp79-92

- ( ) **The manuscript is currently under review for publication**

  Please provide the name of the journal:

- ( ) **It is intended that the manuscript will be published, but it has not yet been submitted to a journal**

| | | | |
|---|---|---|---|
| Student's signature: | Digitally signed by Wenlong Wang Date: 2023.09.07 14:50:20 +01'00' | Main supervisor's signature: | Thomas Pfeiffer Digitally signed by Thomas Pfeiffer Date: 2023.09.08 11:13:33 +12'00' |

*This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.*

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

# STATEMENT OF CONTRIBUTION
# DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the student and the student's main supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the student's contribution as indicated below in the Statement of Originality.

| | |
|---|---|
| Student name: | Wenlong Wang |
| Name and title of main supervisor: | Professor Thomas Pfeiffer |
| In which chapter is the manuscript/published work? | Chapter 3 |
| What percentage of the manuscript/published work was contributed by the student? | 50% |

Describe the contribution that the student has made to the manuscript/published work:
The student is the sole contributor to this work, besides his supervisor.

Please select one of the following three options:

- ○ **The manuscript/published work is published or in press**
  Please provide the full reference of the research output:

- ◉ **The manuscript is currently under review for publication**
  Please provide the name of the journal:
  The 19th Conference on Web and Internet Economics (WINE 2023)

- ○ **It is intended that the manuscript will be published, but it has not yet been submitted to a journal**

| | | | |
|---|---|---|---|
| Student's signature: | Digitally signed by Wenlong Wang Date: 2023.09.07 14:54:25 +01'00' | Main supervisor's signature: | Thomas Pfeiffer Digitally signed by Thomas Pfeiffer Date: 2023.09.08 11:13:57 +12'00' |

*This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.*

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

# STATEMENT OF CONTRIBUTION
# DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the student and the student's main supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the student's contribution as indicated below in the Statement of Originality.

| | |
|---|---|
| Student name: | Wenlong Wang |
| Name and title of main supervisor: | Professor Thomas Pfeiffer |
| In which chapter is the manuscript/published work? | Chapter 4 |
| What percentage of the manuscript/published work was contributed by the student? | 50% |

Describe the contribution that the student has made to the manuscript/published work:
The student is the sole contributor to this work, besides his supervisor.

Please select one of the following three options:

- ○ **The manuscript/published work is published or in press**
  Please provide the full reference of the research output:

- ◉ **The manuscript is currently under review for publication**
  Please provide the name of the journal:
  The 19th Conference on Web and Internet Economics (WINE 2023)

- ○ **It is intended that the manuscript will be published, but it has not yet been submitted to a journal**

| | | | |
|---|---|---|---|
| Student's signature: | Digitally signed by Wenlong Wang Date: 2023.09.07 14:56:00 +01'00' | Main supervisor's signature: | Thomas Pfeiffer Digitally signed by Thomas Pfeiffer Date: 2023.09.08 11:14:24 +12'00' |

*This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.*