UNIVERSITY OF CALIFORNIA,
IRVINE

Beyond Nash Equilibrium: Mechanism Design with Thresholding Agents

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Information and Computer Science

by

Wen Shen

Dissertation Committee:
Professor Cristina Lopes, Chair
Professor Amelia Regan
Professor David Redmiles

2019

DEDICATION

To my family.

TABLE OF CONTENTS

Page

LIST OF FIGURES **vi**

LIST OF TABLES **viii**

LIST OF ALGORITHMS **ix**

ACKNOWLEDGMENTS **x**

CURRICULUM VITAE **xi**

ABSTRACT OF THE DISSERTATION **xiii**

1 Introduction **1**
1.1 Main Contributions of The Dissertation 3
1.2 Outline of The Dissertation 3

2 Background and Concepts **6**
2.1 The Mechanism Design Theory 6
2.1.1 The Mechanism Environment 7
2.1.2 Mechanism Design for Public Good 9
2.1.3 Mechanism Design for Profit 9
2.2 Challenges and Existing Solutions 10
2.2.1 Challenges 10
2.2.2 Existing Solutions 11
2.3 Key Ingredients of The MDTA Approach 12
2.3.1 Thresholding Agents 12
2.3.2 Contests Design 13
2.3.3 Indirect Preference Revelation 13

3 Theory of Mechanism Design with Thresholding Agents **15**
3.1 The MDTA Framework 15
3.1.1 Agent Modeling 16
3.1.2 Preference Ranking 18
3.1.3 Contest Design 19
3.1.4 Indirect Preference Revelation 20
3.2 Evaluation 21
3.3 Applicability 21

4 Information Design for Crowdfunding with Thresholding Agents 23
4.1 Background . . . 24
4.2 Decision Making in Crowdfunding . . . 27
4.2.1 Preliminaries . . . 28
4.2.2 Backers' Decision Model . . . 29
4.2.3 The Entrepreneur's Optimization Problem . . . 30
4.3 Optimal Information Design . . . 31
4.3.1 Solution Concepts . . . 31
4.3.2 Excessive Disclosure Shrinks Revenue . . . 33
4.3.3 Immediate Disclosure is not Always Optimal . . . 35
4.4 Dynamic Information Design . . . 36
4.5 Empirical Evaluation . . . 42
4.6 Summary . . . 45

5 A Post-Price Online for Autonomous Mobility-on-Demand Systems 47
5.1 Background . . . 48
5.2 Ridesharing in AMoD Systems . . . 50
5.3 The Online Mechanism . . . 51
5.3.1 Preliminaries . . . 51
5.3.2 IORS Mechanism . . . 54
5.4 Benchmark . . . 60
5.5 Experimental Results . . . 61
5.5.1 Experimental Settings . . . 61
5.5.2 Results and Discussion . . . 62
5.6 Summary . . . 64

6 Multi-Winner Contests for Strategic Network Diffusion 66
6.1 Background . . . 67
6.2 Strategic Diffusion in Social Networks . . . 69
6.2.1 Preliminaries . . . 70
6.2.2 Mechanism Design for Strategic Diffusion . . . 71
6.2.3 False-Name Attacks . . . 73
6.2.4 Solution Concepts . . . 74
6.3 Multi-Winner Contests Mechanism . . . 76
6.3.1 Virtual Credits for Diffusion Contributions . . . 76
6.3.2 Diffusion Rewards . . . 77
6.3.3 The MWC Mechanism . . . 79
6.4 Experiments . . . 81
6.4.1 Experimental Settings . . . 81
6.4.2 Results . . . 83
6.5 Summary . . . 87

7 A Generalized Contest Mechanism for Strategic Network Diffusion **89**
7.1 Background . . . 89
7.2 Strategic Network Diffusion . . . 92
7.2.1 Notations . . . 92
7.2.2 The Mechanism Design Problem . . . 94
7.3 Design Constraints . . . 95
7.3.1 False-Name Proofness . . . 96
7.3.2 Collusion-Proofness . . . 97
7.3.3 Other Constraints . . . 99
7.4 Generalized Contest Mechanism . . . 100
7.4.1 Superadditivity . . . 101
7.4.2 Increasing and Bounded Functions . . . 101
7.4.3 The GCM . . . 103
7.5 Experiments . . . 106
7.5.1 Experimental Settings . . . 107
7.5.2 Results . . . 109
7.6 Summary . . . 111

8 Countering Free Riding in Utility Sharing with Sequential Contests **112**
8.1 Background . . . 113
8.2 The Free Rider Problem in Utility Sharing . . . 115
8.2.1 Notations . . . 115
8.2.2 The Free Rider Problem . . . 116
8.2.3 Mechanism Design in Utility Sharing . . . 118
8.3 Sequential Contest Mechanism . . . 119
8.4 Experiments . . . 122
8.4.1 Experimental Settings . . . 122
8.4.2 Results . . . 124
8.5 Related Work . . . 129
8.6 Summary . . . 130

9 Conclusions and Future Directions **131**
9.1 Our Contributions . . . 131
9.2 Future Directions . . . 133

Bibliography **134**

A Derivations and Proofs for Chapter 4 **143**
A.1 Proofs . . . 143
A.2 Expert Algorithms . . . 147

B Derivations and Proofs for Chapter 6 **151**
B.1 Proofs . . . 151

LIST OF FIGURES

Page

2.1 An illustration of mechanism design. 8

3.1 An illustration of mechanism design with thresholding agents. 16
3.2 An illustration of a thresholding agent' decision-making process. 17
3.3 An illustration of contest design. 19
3.4 An illustration of indirect preference revelation. Here, a is not affordable when b is chosen; b is not affordable when c is chosen; a and c cannot be compared directly. 20
3.5 An illustration of different evaluation methods. 21

4.1 Procedure of a typical crowdfunding campaign. 28
4.2 An illustration of the Dynamic Shrinkage with Heuristic Selection algorithm. 37
4.3 A comparison of performance. 45

5.1 A simple scenario of ridesharing: dashed lines indicate the real demand; solid lines with arrow indicate actual routes. 48
5.2 An abstraction of ridesharing in an AMoD system operating in a grid city. . 51
5.3 A comparison of demand distribution, the social welfare scores and computational time of a system with three different approaches: the IORS, an auction-based mechanism and the optimal solution. 63
5.4 A comparison of the performance of a system with three approaches: the IORS, an auction-based mechanism, and the optimal solution. 64

6.1 A typical directed acyclic graph. 70
6.2 False-name attacks at node 2: nodes $2_1, 2_2, ..., 2_m$ are replicas of node 2. . . . 74
6.3 Probability density function of the means of players' abilities by four different groups: HO, HL, HH and DI. 83
6.4 A comparison of total contributions with different bias factors. 84
6.5 A comparison of number of participants and average contributions per player on Twitter. 85
6.6 Rewards (normalized) obtained by a player that creates different number of false identities for groups with different noise factors. 87

7.1 A simple illustration of strategic network diffusion. 92
7.2 A typical directed acyclic graph for modeling strategic network diffusion. . . 92
7.3 Three types of false-name attacks at node d. 96
7.4 An example of strategic network diffusion with no collusion and with collusion under a typical geometric mechanism. 98

7.5 Average rewards (normalized by the highest number in each dataset) for each player that creates different false identities. 108
7.6 Average rewards (normalized by the highest number in each dataset) for each player that is in different level of collusion. 109

8.1 An illustration of the free rider problem. 114
8.2 A comparison of four effort metrics by varying the noise factor from 0.0 to 1.0. 125
8.3 A comparison of four utility metrics by varying the noise factor from 0.0 to 1.0. 126
8.4 A comparison of total efforts and total utility by varying the discounting factor from 1.00 to 1.10 with threshold $\theta = 0.50$, noise factor $\rho = 0.06$. 127
8.5 A comparison of total efforts and total utility by varying the threshold from 0.0 to 1.0 with noise factor $\rho = 0.06$ and discounting factor $\lambda = 1.06$. 128
8.6 A comparison of total efforts and total utility by percentage of free riders from 0% to 100% with threshold $\theta = 0.50$, noise factor $\rho = 0.06$ and discounting factor $\lambda = 1.06$. 128

LIST OF TABLES

Page

6.1 Dataset configuration (M.D.–maximum degree, A.D.–average degree) 82

7.1 A comparison of the total task efforts for different groups 107

LIST OF ALGORITHMS

Page

4.1 Dynamic Shrinkage with Heuristic Selection 38
4.2 Shrink 39
4.3 Meta 41
5.1 The Fare Estimation Algorithm 55
5.2 The Pickup Assignment Algorithm 56
6.1 Multi-Winner Contests Mechanism 80
7.1 Generalized Contest Mechanism 105
8.1 Sequential Contest Mechanism 121

ACKNOWLEDGMENTS

It would be definitely impossible for me to finish this thesis without the long-lasting support from many individuals over the years. Among them, I would like to particularly thank my advisor, my dissertation committee members, my collaborators, my peers, and my family.

I would like to express my deepest gratitude to my advisor, Prof. Cristina Lopes. Crista has been a reliable source of inspiration for me since 2014. She has always been encouraging and supportive, especially when I have difficulties in my academic life. Even though my research interests diverge from her core expertise, Crista gives me the freedom to explore new exciting topics and provides me with helpful resources whenever I need them. She not only provides me with insightful feedback about my research but also points out the right directions for me to improve. I have benefited a lot from the conversations with her. I consider myself very fortunate to work with her for my PhD study.

I am very grateful to the other two members of my doctoral committee – Prof. Amelia Regan and Prof. David Redmiles. Prof. Regan is very knowledgeable and highly supportive. She not only gives me many insightful comments about my research, but also provides the pointers to the resources that I need. For some reason, I was struggling to find a dissertation committee member for a long time. Prof. Redmiles is so kind as to be willing to serve on my committee, saving me at the last minute. He gives me a lot of constructive comments that help improve the quality of my dissertation substantially. I also want to thank my advancement committee Prof. James Jones, Prof. Wenlong Jin and Prof. Alfred Kobsa for their useful feedback.

Much of this thesis draws on the collaborative work with my coauthors. Of them, I am especially indebted to my former advisor, Prof. Jacob Crandall, and my friend Prof. Ke Yan for providing invaluable assistance in research as well as my job hunting. I have learned a lot from their suggestions and comments.

I would like to extend thanks to my lab mates Rohan Achar, Eugenia Gabrielova, Vaibhav Saini, Hitesh Sajnani, Thomas Debeauvais, Arthur Valadares, Di Yang, Farima Farmahini-Farahani, Maruf Zaber, and other close friends at UCI, including Yang Feng, Qi Lou, and Zhendong Wang. They have offered a friendly learning environment for me. I would like to thank my other peers even though I am unable to list their names completely.

My heartfelt appreciation goes to my family for their endless support. I want to thank my parents, grandparents, siblings and other relatives for creating a worry-free environment for me to concentrate on my study. I would like to thank my little nephew Boyue for the joy he has brought over video calls, even though I haven't meet him in person yet. Most importantly, my earnest appreciation belongs to Junzhi Wu for her unconditional love, true understanding, and inspirational words of encouragement.

My graduate studies were supported in part by the National Science Foundation through grant no. CCF-1526593.

CURRICULUM VITAE

Wen Shen

EDUCATION

Doctor of Philosophy in Information and Computer Science **2019**
University of California, Irvine — Irvine, California, USA

Master of Science in Computing and Information Science **2013**
Masdar Institute of Science and Technology — Abu Dhabi, UAE

Bachelor of Engineering in Software Engineering **2009**
Northwestern Polytechnical University — Xi'an, China

SELECTED PUBLICATIONS

W. Shen, Y. Feng, C.V. Lopes. Multi-Winner Contests for Strategic Diffusion in Social Networks. *AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

K. Yan, **W. Shen**, H. Lu, Q. Jin. Emerging Privacy Issues and Solutions in Cyber-Enabled Sharing Services: From Multiple Perspectives. *IEEE Access*, 2019.

W. Shen, R. Achar, C.V. Lopes. oward Understanding the Impact of User Participation in Autonomous Ridesharing Systems. *Winter Simulation Conference (WSC)*, 2018.

W. Shen, J. W. Crandall, K. Yan, C. V. Lopes. Information Design in Crowdfunding under Thresholding Policies. *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2018.

K. Yan, Z. Ji, **W. Shen**. Online Fault Detection Methods for Chillers Combining Extended Kalman Filter and Recursive One-class SVM. *Neurocomputing*, 2017.

W. Shen, A. A. Khemeiri, A. Almehrezi, W. Al-Enezi, I. Rahwan, J. W. Crandall. Regulating Highly Automated Robot Ecologies: Insights from Three User Studies. *International Conference on Human-Agent Interaction (HAI)*, 2017.

W. Shen, C. V. Lopes, J. W. Crandall. An Online Mechanism for Ridesharing in Autonomous Mobility-on-Demand Systems *International Joint Conference on Artificial Intelligence (IJCAI)*, 2016.

W. Shen, C. V. Lopes. Managing Autonomous Mobility on Demand Systems for Better Passenger Experience. *International Conference on Principles and Practice of Multi-Agent Systems (PRIMA)*, 2015.

WORKING PAPERS

W. Shen, R. Achar, C.V. Lopes. A Simulation Analysis of Large Contests with Thresholding Agents. *Preprint*, 2019.

W. Shen, K. Yan, C.V. Lopes. Manipulation-Resistant Mechanism Design for Strategic Network Diffusion. *Preprint*, 2019.

W. Shen, C.V. Lopes. Countering Free Riding in Utility Sharing with Sequential Contests. *Preprint*, 2019.

ABSTRACT OF THE DISSERTATION

Beyond Nash Equilibrium: Mechanism Design with Thresholding Agents

By

Wen Shen

Doctor of Philosophy in Information and Computer Science

University of California, Irvine, 2019

Professor Cristina Lopes, Chair

In many real-world scenarios, individual agents' interests are often not fully aligned, or even conflicting, with a principal's objectives. The principal needs to take measures to influence agents' decisions or behavior to achieve desirable system-wide outcomes. A powerful tool for motivating self-interested agents to cooperate is to offer incentives for their efforts (e.g. cooperation or sacrifices) by committing to some allocation and payment rules. This approach to implementing the incentive rules is called mechanism design. Mechanism design has many promising applications in a variety of critical domains that include spectrum allocation, online marketplace, transportation management, power grids, education, and health care. Despite its promising prospects in addressing some of the most challenging societal issues, mechanism design has not leveraged its full potential due to assumptions such as full rationality, direct preference revelation, and no group manipulation.

In this work, I introduce a unified framework called mechanism design with thresholding agents (MDTA) to relax some of those unrealistic assumptions. The proposed approach integrates a series of new techniques that include modeling agents' decision-making with cutoff policies, indirect preference revelation, and using contests to increase competition among agents to counter group manipulations. I demonstrate the power of the proposed

framework by applying it to real-world problems that arise in crowdfunding, transportation systems, information diffusion, and utility sharing. My work extends traditional mechanism design by providing a systematic approach to influencing agents' behavior for desirable objectives.

Chapter 1

Introduction

In many real-world applications, a stakeholder's objectives are often conflicting or at least not fully aligned with individual players' interests [Myerson, 1982, Youn et al., 2008, Shen et al., 2017]. For instance, a seller often wants to sell a product at a high price to increase revenue while buyers usually would like to buy goods or services with low budgets. In this case, the buyers may not cooperate with the seller, making it challenging for the seller to generate revenue. In transportation systems, the transportation authority of a city is interested in optimizing the throughput of the overall road networks. The individual commuters, however, care more about the traffic conditions of the particular routes they have selected. They may choose a route that already encounters heavy traffic instead of using alternative routes. Commuters' non-cooperative behavior often results in high price of anarchy of the system efficiency [Youn et al., 2008, Shen et al., 2018a]. Another example is that most people receive benefits from open source software development, whereas few are willing to devote efforts to it [Hippel and Krogh, 2003]. As a result, fewer open source software products are available to the public. Such uncoordinated behavior often brings the *tragedy of the commons* [Hardin, 1968].

To achieve desired objectives, a stakeholder must take measures to influence individual agents' decisions or behavior [Shen et al., 2017]. A powerful tool to influence agents' behavior is mechanism design [Clarke, 1971, Myerson and Satterthwaite, 1983, Nisan and Ronen, 2001, Hurwicz and Reiter, 2006]. It coordinates agents' decisions or behavior by committing a social choice function (i.e., an allocation rule and a payment rule) that implements the desired objectives in an equilibrium (e.g., Nash Equilibrium, Bayesian Nash Equilibrium) [Nisan and Ronen, 2001]. Since its inception, mechanism design has witnessed great successes in many critical domains including spectrum management [Huang et al., 2006], transportation sectors [Kamyab et al., 2016], power grids [Samadi et al., 2012], online marketplaces [Edelman et al., 2007], education [Abdulkadiroğlu and Sönmez, 2003], health care [Roth et al., 2004], sponsored search, and social networks [Zhao et al., 2018].

Despite its promising prospects in addressing some of the most challenging societal issues, mechanism design has not leveraged its full potential. There are several challenges that need to be addressed before it becomes fully fledged. Traditional mechanism design assumes that the agents are fully rational [Clarke, 1971, Myerson and Satterthwaite, 1983, Nisan and Ronen, 2001]. However, this assumption is often violated in many cases due to factors such as limited computational power, cognitive biases, and time constraints [Gigerenzer and Goldstein, 1996]. Second, most mechanisms rely on direct preference revelation in the sense that they require agents to report their preferences directly to the mechanisms [Clarke, 1971, Myerson and Satterthwaite, 1983]. In practice, agents are likely to be unwilling or unable to report their preferences directly due to privacy concerns or uncertainty [Naor et al., 1999]. Third, while much research has been focused on efficient mechanisms or optimal mechanisms, little attention has been given to mechanisms that are resistant to manipulations such as false-name attacks, collusion and free riding [Shen et al., 2019]. Traditional mechanism design typically assume that agents do not perform group manipulations [Varian and Harris, 2014]. In many real-world scenarios, however, agents have the incentives to form collusive coalitions to gain benefits [Hendricks and Porter, 1989, Conitzer and Sandholm, 2006]. Therefore, a unified framework to address these challenges is needed to fully leverage the power of mechanism design.

1.1 Main Contributions of The Dissertation

In this dissertation, we introduce a unified framework called Mechanism Design with Thresholding Agents to relax the full rationality assumption, liberate the restriction of direct preference revelation, and counter group manipulations. The framework consists of three integral components: thresholding agents, indirect revelation, and contest design. Specifically, the framework relaxes the full rationality assumption by modeling the agents as the ones that use cutoff policies to make decisions. Instead requiring agents to directly report their preferences, the framework uses post-price methods to infer agents' preferences. To counter group manipulations, the framework introduces contests to increase the competitions among agents so that performing group manipulations are not profitable. We demonstrate the promising prospects of the proposed approach by applying it to addressing real-world challenges in crowdfunding, transportation systems, strategic information diffusion, and utility sharing. To the best of our knowledge, this is the first approach that can address all the three challenges when agents use cutoff policies. Our approach complements traditional mechanism design by extending it to situations that consist of agents with bounded rationality.

1.2 Outline of The Dissertation

A brief outline of the remaining dissertation is as follows: Chapter 1 introduces the background of dissertation including the mechanism design theory, the existing challenges and a brief of our approach. Chapter 3 presents theory of mechanism design with thresholding agents. Chapters 4 5, 6, 7, and 8 present our studies on applying the framework of mechanism design with thresholding agents to solving real-world problems in crowdfunding, ridesharing systems, strategic network diffusion and utility sharing. Chapter 9 concludes the dissertation work and discuss future directions. Specifically,

Chapter 2 discusses the background of the dissertation. It first introduces the theory of mechanism design, including mechanism design for public good and mechanism design for profit. It then discuss the challenges existing in mechanism design. After reviewing the related work on addressing the challenges, it presents the concepts of mechanism design with thresholding agents.

Chapter 3 introduces the theory of mechanism design with thresholding agents (MDTA). It first presents the four stages of MDTA and then discuss the evaluation methods as well as the applicability of MDTA.

Chapter 4 studies the information design problem in which an entrepreneur voluntarily reveals the project status of a campaign to backers to influence their beliefs of the project's probability of success so that the entrepreneur can solicit sufficient contributions before the deadline. We first review some related work in crowdfunding. We then present agents' decision models and mechanism design problem. After an analysis of optimal information design, we introduce the dynamic information design approach to help the entrepreneur increase the revenue. We then report the experimental settings and the results. Finally, we summarize the work. This chapter is based in part on our published work, [Shen et al., 2018b].

Chapter 5 introduces an online post-price mechanism to reduce the operational cost per unit demand in order to promote rideshairng in autonomous mobility-on-demand systems. We first discuss the background of ridesharing in autonomous mobility-on-demand systems. After we formally introduce the mechanism design problem, we introduce the post-price, integrated online ridesharing mechanism to help reduce the operational cost of the ridesharing systems. This chapter is based in part on our published paper, [Shen et al., 2016].

Chapter 6 proposes a novel multi-winner contests mechanism that is resistant to false-name attacks in strategic information diffusion in social networks. We first introduce the motivation behind this work. We then formulate the mechanism design problem and the design constraints. After we present the multi-winner contests mechanism, we demonstrate its competitiveness and robustness via extensive simulations on real-world datasets. Our work sheds light on how to integrate competitions into the design of novel mechanisms to counter manipulations. This chapter is based in part on our published research, [Shen et al., 2019].

Chapter 7 presents a generalized contest mechanism that is resistant to both false-name attacks and collusion. The generalized contest mechanism counters both manipulations through an elegant integration of two functions (i.e., the superadditive function, and the increasing and bounded function) into the design of the diffusion reward function. We demonstrates that the generalized contest mechanism is false-name-proof and collusion-proof. Experimental results shows that the generalize contest mechanisms can outperform the multi-winner contests mechanism in terms of total task efforts with a variety of supperadditive functions and increasing and bounded functions. Our work provides a systematic approach to manipulation-resistant mechanism design.

Chapter 8 describes a study on countering free riding in utility sharing. In this chapter, we introduce a sequential contest mechanism that is resistant to free riding. The mechanism allocates a number of tokens to each agent that has made contributions to the utility production before a private deadline, and then conducts a ratio-form contest among the qualified agents to distribute the rewards.

Chapter 9 concludes the dissertation and discusses possible directions for future research. Specifically, we first summarize the dissertation research and then highlight the major contributions of our work. In the end, we discuss the future work.

Chapter 2

Background and Concepts

In this chapter, we first introduce the theory and mechanism design. In particular, we focus on the two types of mainstream mechanism design: *mechanism design for public good* and *mechanism design for profit*. We then discuss the challenges in traditional mechanisms and review some related work that aim to address them. Finally, we introduce the concepts of the unified framework—mechanism design with thresholding agents (MDTA) to address the challenges.

2.1 The Mechanism Design Theory

Mechanism design is a subfield of game theory. It takes an engineering approach to designing incentives (i.e., allocation rules and payment rules) to achieve desired outcomes in the environment that consists of multiple self-interested players [Hurwicz and Reiter, 2006]. Unlike a typical game that focuses on the consequences of strategic interactions among players given the rules, mechanism design takes a desired objective as the prior knowledge to find out the proper rules that can influence agents' strategic interactions. Hence, mechanism design is also called *reverse game theory* [Nisan and Ronen, 2001].

Depending on the nature of the objectives, mechanism design can be generally categorized into two groups: mechanism design for public good [Clarke, 1971] and mechanism design for profit [Krishna and Perry, 1998, Myerson and Satterthwaite, 1983]. In mechanism design for public good, the stakeholder's goal is to maximize the social welfare of all the players while an individual agent is interested in his own payoffs [Börgers, 2015]. The stakeholder's goal is partially aligned with that of an individual agent. In mechanism design for profit, the stakeholder aims at to optimize the revenue or the profit [Myerson, 1981]. The individuals players, on the other hand, care more about their own utilities or profits.

2.1.1 The Mechanism Environment

A typical mechanism design setting consists of three components: individual agents, social choices, private preferences [Jackson, 2014]. We discuss each in turn:

- **Individual agents:** A mechanism setting involves a finite group N of self-interested individuals.
- **Social choices:** In a mechanism environment, the principal has a set of potential social choices or social decisions D with $d \in D$ be a generic social choice. A social choice $d \in D$ represents a allocation rule and a payment rule that determine which individual agents get the goods $g \in G$ at what price. The set of social choice may be finite or infinite depending on the application.
- **Private preferences:** Individual agents have private information of their values. They may be unwilling to disclose the private information without incentives. Let $\theta_i \in \Theta_i$ denote the private information of a player $i \in N$, the set of all the agents' private information is $\Theta = \times_i \Theta_i$ with $\theta = (\theta_i)_{i \in N}$.

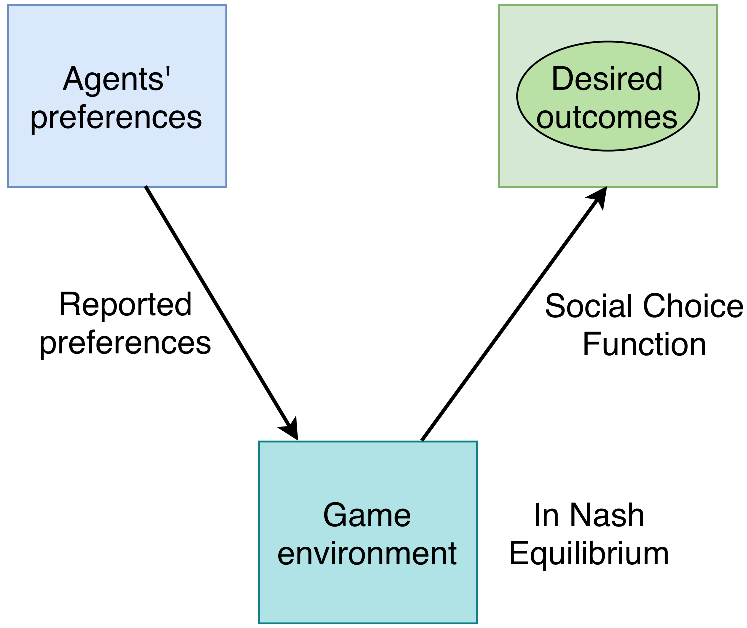


Figure 2.1: An illustration of mechanism design.

A mechanism M is an algorithm that asks the individual agents in N to directly reveal their private information θ, allocates the goods G to the players in N and collects payments from the players in N (See Fig. 2.1 for an illustration).

Players may have incentives to lie about the reports of their preferences. In order to be implementable in practice, a mechanism needs to implement the social choice function in equilibrium (e.g., dominant strategy equilibrium, Nash equilibrium, Bayesian Nash equilibrium) [Nisan and Ronen, 2001]. In a dominant strategy equilibrium, it is in the interest of an agent to truthfully report his preference regardless of other agents' strategies. In Nash equilibrium, it is incentive compatible for an agent to truthfully reveal his preference if others do not change their strategies. In Bayesian Nash equilibrium, an agent would reveal his true valuation if he believes that others also truthfully reveal their preferences.

2.1.2 Mechanism Design for Public Good

In mechanism design for public good, the principal aims at social welfare maximization for all the agents. That is, the principal is interested in the efficiency of the allocation of the goods. The principal wants to allocate the goods to the agents that have the highest values. A famous mechanism that implements the efficient outcomes in dominant strategy is the *Vickrey-Clarke-Groves (VCG)* mechanism [Hurwicz and Reiter, 2006]. Under the VCG mechanism, it is dominant-strategy incentive compatible for a player to truthfully report his preferences to the mechanism. The mechanism allocates the goods to the players with the highest valuations. Each player pays his social cost— the marginal cost he brings to the society due to his existence [Hurwicz and Reiter, 2006].

The VCG mechanism is individually rationally and does not require distributional assumptions. It has many applications in practice, including sponsored search, spectrum allocation and online marketplaces. However, it is not budget balanced and may receive low revenue [Conitzer and Sandholm, 2004]. Besides the VCG mechanism is prone to other manipulations such as collusion [Conitzer and Sandholm, 2004].

2.1.3 Mechanism Design for Profit

In mechanism design for profit, the mechanism designer focuses on the revenue optimization. To achieve the optimal revenue, the mechanism designer needs to have prior information about the distribution of agents' valuation. A celebrated mechanism that implement the optimal outcomes in Bayesian Nash equilibrium is the Myerson's auction. In Myerson's optimal mechanism, the mechanism allocates the goods to the players that have the highest virtual values [Myerson, 1981]. The virtual value of an agent is the surplus that can be extracted from the agent given the value distribution of all the agents. The prices for the winning players are determined according to the VCG mechanism.

The Myerson's optimal mechanism has practical applications [Hurwicz and Reiter, 2006]. For instance, a seller often wants to sell products to potential buyers to maximize the profits. If the seller has the information of agents' value distributions, the seller can commit the Myerson's auction to achieve her objective. The Myerson' optimal mechanism requires prior knowledge about the agents' value distributions, which is not always feasible in practice. It is also vulnerable to manipulations such as collusion.

2.2 Challenges and Existing Solutions

The application of traditional mechanism design has been restricted due to some of the unrealistic assumptions: the full rationality assumption, the direct preference revelation assumption, and the no group manipulation assumption. There are a few attempts that aim to address the challenges.

2.2.1 Challenges

Despite its promising prospects in addressing some of the most challenging societal issues, mechanism design has not leveraged its full potential. There are four challenges that need to be addressed before it becomes fully fledged. First, traditional approaches to designing mechanisms typically require substantial prior knowledge about participating agents' value distributions. Nevertheless, mechanism designers usually do not have such knowledge in hand. Second, mechanism designers often must assume that the agents are fully rational. However, this assumption is often violated in many cases due to factors such as limited computational power, cognitive biases, and time constraints [Gigerenzer and Goldstein, 1996]. Third, efficient mechanism design relies on directly revealed preferences. In many real-world environments, people are reluctant to reveal their preferences directly due to privacy concerns [Naor et al., 1999]. Forth, while much research has been focused on efficient or optimal mechanisms, little attention has been given to mechanisms that are resistant to manipulations such as false-name attacks, and collusion.

2.2.2 Existing Solutions

New lines of research have begun to address the four challenges. To relax the requirement of agents' value distributions, Devanur and Hartline [2009] introduced the problem of *prior-free mechanism design*. In general, there are three methods to prior-free mechanism design: deterministic empirical distribution [Nisan et al., 2007], random sampling [Devanur and Hartline, 2009], and consensus estimates [Goldberg and Hartline, 2003, Ha and Hartline, 2013]. While random sampling can provide a good approximation of optimal mechanisms in many scenarios, deterministic empirical distribution and consensus estimates may perform arbitrarily poor if the feasibility constraints or the consensus-estimate functions are not satisfied. All the three methods require direct revelation of agents' preferences, making them less appealing to be deployed in practice. To relax the rationality assumption, Ghosh and Kleinberg [2014] introduced the concept of "simple agents" where agents only reason on whether to participate in or not. Their work assumes that the agents have homogeneous valuations. It remains unknown how to extend their approach to the scenarios where agents have different value distributions. Most research on mechanism design restricts their attention to direct mechanisms where agents must explicitly reveal their preferences. In practice, however, posted-price mechanisms are more favorable due to its efficiency and ease of implementation [Wang et al., 2008]. Besides, direct mechanisms are also usually vulnerable to manipulations such as false-name attacks [Nath et al., 2012] and collusion [Laffont and Martimort, 2000]. To address the four challenges, a new unified framework to mechanism design is needed

2.3 Key Ingredients of The MDTA Approach

To relax the unrealistic assumptions in traditional mechanism design, we introduce a unified framework called *mechanism design with thresholding agents* (MDTA). The framework consists of three integral components: threshoding agents, contest design, and indirect preference revelation.

2.3.1 Thresholding Agents

Previous research on mechanism design usually assumes that agents are fully rational. However, this assumption is often not grounded [Shen et al., 2017, 2018a]. To address this problem, we proposed to use cutoff policies to model agents' decision-making processes [Shen et al., 2018b]. In doing so, each agent has a private threshold value to decide to participate in the mechanism or not. This method relaxes the assumption that each agent is fully rational and allows mechanism designers to model agents' valuations with greater flexibility instead of making strong assumptions about their rationality. In repeated interactions (e.g., repeated auctions), it is also convenient for the mechanism designers to learn and make inference about agents' valuation or types with preference data using the cutoff policies. When designing manipulation-resistant mechanisms, the mechanism designers often must estimate agents' costs of performing a specific action (e.g., false-name attacks). In many cases, however, the designers are unable to make an accurate estimation for every agent. We introduced a parameter called *cost coefficient* to help model agents' costs [Shen et al., 2019]. The cost coefficient is privately known to an agent only. This approach relieves the mechanism designers' burden of precisely calculating agents' costs of efforts. It also makes it possible for the designers to learn agents' cost functions through empirical data in repeated interactions.

2.3.2 Contests Design

Many online platforms (e.g., online marketplaces, crowdfunding sites, social media platforms) rely on high-quality online customer reviews to build reputations. However, online ratings are often vulnerable to manipulations. For instance, a user may post fake reviews for a seller in exchange for a free item or a discount. Such activities will undermine customers' confidence or trust in the online platforms. Another example is that a user may create multiple fake accounts to make profits without making any contributions towards promoting the task owners' goals in social media marketing [Conitzer et al., 2010]. To address this problem, we proposed to use multi-winner contests to design manipulation-resistant mechanisms [Shen et al., 2019]. We demonstrated that proper peer competition could significantly reduce the incentives of manipulations (e.g., false-name attacks) as well as increase agents' aggregated efforts in social network marketing. It will be interesting to explore novel contest structures to counter agents' collusive behaviors. Another direction is to design contest mechanisms to maximize agents' aggregated efforts.

2.3.3 Indirect Preference Revelation

Efficient mechanism design typically requires agents to reveal their valuations or preferences directly. However, this requirement is often problematic for two reasons. First, agents may be unaware of their accurate valuations. They may overestimate or underestimate their valuations. Second, even if they know their actual values, they might be reluctant to disclose the information to the mechanism designers due to privacy concerns or for fear that the designers will use this information in an unwanted way in the future. To address this issue, the designers can implement indirect (i.e., posted-price) mechanisms [Shen et al., 2016, 2019]. This method does not require agents to report their valuations. However, it does require the designers to have prior knowledge about agents' value distributions. When the designers do not have prior knowledge about agents' value distributions, it is possible to design randomized mechanisms to learn the value distributions by offering free items with monotonic increment in the agents' valuations. This method is usually applicable to online scenarios when agents' value distributions are learnable. It may result in a loss of optimality when agents' valuations are sufficiently large.

Chapter 3

Theory of Mechanism Design with Thresholding Agents

This chapter presents the theory of *mechanism design with thresholding agents* (MDTA), followed by a discussion of evaluation methods and the potential application domains of MDTA.

3.1 The MDTA Framework

The MDTA framework consists of four stages: agent modeling, preference ranking, contest design, and indirect preference revelation (See Fig. 3.1 for an illustration). In agent modeling, the principal (i.e., the mechanism designer) isolates agents' primary alternatives of interest and convert the preference relations to suitable utility functions given that agents have their private thresholds to trigger their decisions. In preference ranking, the principal identifies the relative orders of agents' preferences based on ground truth or reasoning. In contest design, the principal selects optimal parameters for the contest through simulations. Finally, the principal allocates the goods to the agents based on the outcomes of the contest and charge them accordingly.

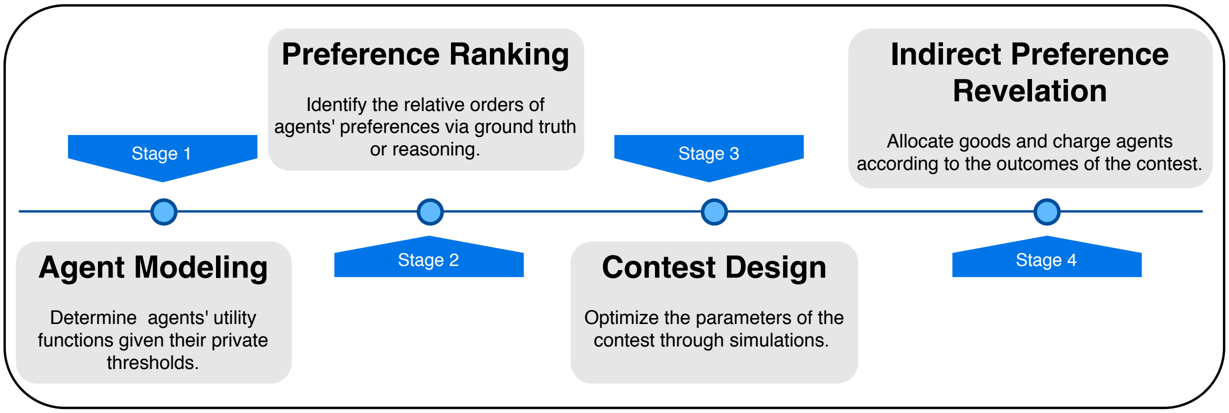


Figure 3.1: An illustration of mechanism design with thresholding agents.

3.1.1 Agent Modeling

In agent modeling, the principal identifies agents' primary alternatives of interest and determines their utility functions given that agents have their own private thresholds to trigger their decisions (See Fig. 3.2 for an illustration of a thresholding agent' decision-making process). In this work, we assume that agents' thresholds are fixed. The thresholds serve as the priors that the agents hold to represent their perceptions of the acceptable consequence should they participate (or do not participate) in the activities set by the principal. The perceptions might be fully or partially correct and thus the degree that the thresholds represents the reality vary from agent to agent.

When making decisions, a thresholding agent determines a threshold (i.e., a prior) that indicates the tipping point that the agent take actions either towards or against the principal's objectives. For instance, a buyer often has a budget to buy a product. The upper bound of the budget is a threshold of the buyer. That is, the buyer would only buy the product if its price is no greater than the threshold. If the seller sets the price lower than to equal to the threshold, then the buyer's action is favorable for the seller. Otherwise, the buyer's action is conflicting with the seller's interest (i.e., revenue maximization).

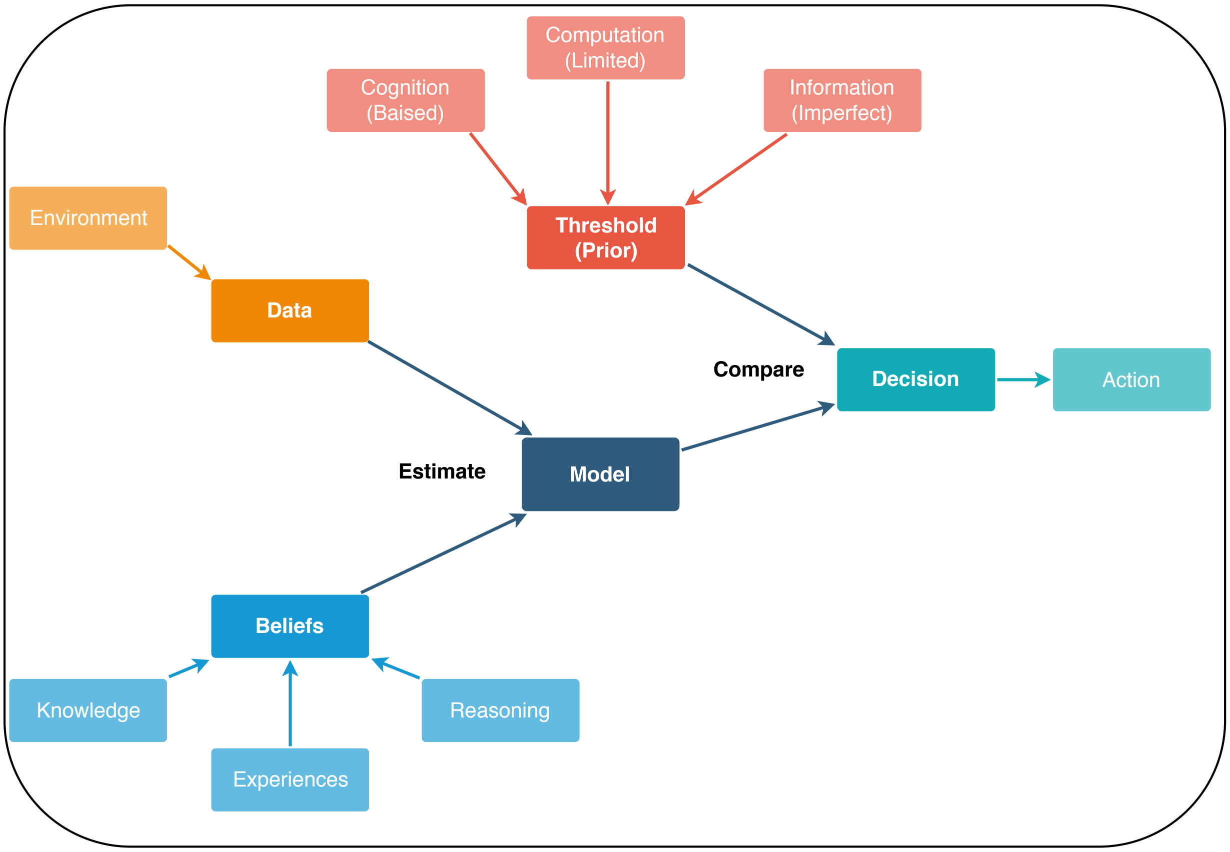


Figure 3.2: An illustration of a thresholding agent' decision-making process.

Agents' thresholds are often constrained by their cognitive abilities, available information and computational capabilities [Kau and Hill, 1972, Kahneman, 2003, Zhou et al., 2005, Shen et al., 2018b]. Thus, agents' decisions can be optimal or suboptimal depending on whether their thresholds are correct or not. When all the agents have correct thresholds or priors, their behaviors are rational.

Before making a decision, the agent also needs to establish a model of the world. The model derives from an estimation based on the data and the agents' beliefs. For example, before a buyer decides to buy the product, he needs to know the details of the transaction (e.g., the quality of the good, the delivery policy and return policy). The model serves as the agent's benefit evaluation or risk assessment.

If the agent's estimated value exceeds the his threshold, then the agent would take an action that either go towards or against the principal's objective.

3.1.2 Preference Ranking

In preference ranking, the mechanism designer identifies the relative orders of agents' preferences via ground truth or reasoning. When selling a product, the seller needs to identify the partial order that all the buyers would prefer to buy the product with a lower price rather than a higher price. This partial order is drawn from empirical observations.

In decision theory, a preference ordering $a \succsim b$ means that choice a is at least as good as b. A preference ordering $a \succ b$ refers to that choice a is strictly preferred to b. Rational choice satisfies the following two properties: completeness and transitivity.

Definition 3.1 (Rational Choice)**.** A preference relation $\succsim$ is rational if it satisfies:

- **Completeness:** for all a, b, we have $a \succsim b$ or $b \succsim a$.
- **Transitivity:** for all a, b, c, if $a \succsim b$ and $b \succsim c$, we have $a \succsim c$.

A preference relation that is complete and transitive can be represented by a utility function (See Def. 3.2).

Definition 3.2 (Utility Function)**.** A utility function $u : X \to \mathbb{R}$ represents preference relation $\succsim$ if, for all a, b,

$$a \succsim b \Leftrightarrow u(a) \geq u(b) \ . \tag{3.1}$$

The utility function measures the level of pleasure or satisfaction that an agent receives due to a choice. Note that not all preference relations have a utility function representation. Only preferences orderings that are complete and transitive can be represented by a utility function.

3.1.3 Contest Design

In contest design, the mechanism designer selects the optimal parameters for the contest success function and the allocation rules to achieve her desired goals (See Fig. 3.3 for an illustration).

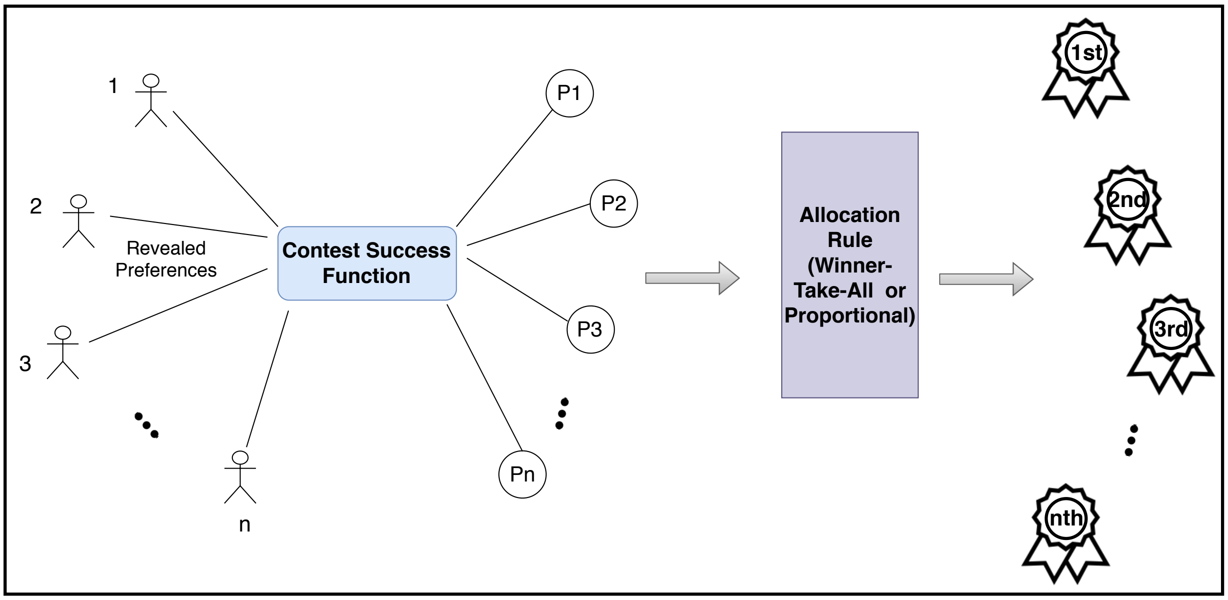


Figure 3.3: An illustration of contest design.

In this stage, the principal determines the probability of winning for each agent that has revealed his preference (e.g., participation, efforts) and the rules for allocating the goods. The probability of winning is determined by a contest success function that takes the observed preferences of all the agents as the input. The contest success function can be either ration-form or difference-form. The former determines the probability of winning by a function of the ratio of the respective preferences (commitments) while the later is a function of the differences between agents' preferences (commitments). The allocation rules can be winner-take-all or proportional. In winner-take-all allocation, the principal only distributes the goods to the winners. In proportional allocation, the goods are distributed proportionally according to the probability of winning.

3.1.4 Indirect Preference Revelation

When determining the payment for each agent, the principal uses a post-price mechanism. Instead of asking the agents to directly reveal their preferences, the principal can provide a price, observe agents' choice, infer and learn their preferences. For example, given the observed preferences shown in Fig. 3.4, if choice b is selected by an agent, then choice a is not affordable to him. Similarly, b is not affordable when he choose c. Choice a and choice c cannot be directly compared. However, by transitivity, we can infer that if the previous relations all hold, then c is revealed indirectly as preferred to a.

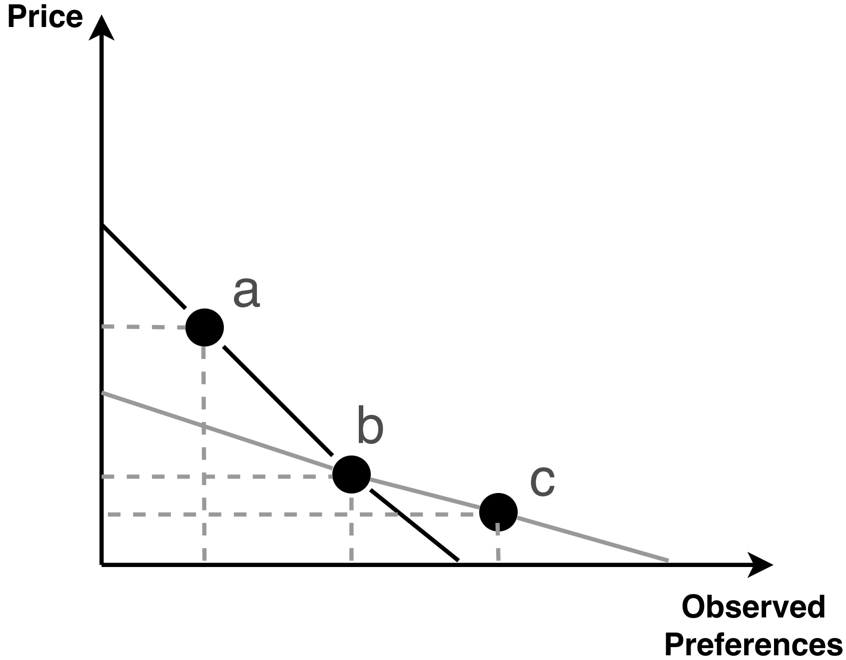


Figure 3.4: An illustration of indirect preference revelation. Here, a is not affordable when b is chosen; b is not affordable when c is chosen; a and c cannot be compared directly.

The observed preferences can be recorded for future reference (e.g., learning and inference). This stage avoids seeking direct preference revealtion from the agents and helps protect the privacy of agents.

3.2 Evaluation

Evaluation methods vary from low-fidelity simulations to high-fidelity simulations and to deployment in practice (See Fig. 3.5 for an illustration). When the principal has no empirical data available, she can produce synthesis data according to some rules or assumptions and execute the mechanism on the dataset with different parameter settings. When empirical data is available, the principal can conduct realistic simulations that bring more insights about the performance of mechanism in real-world scenarios. If possible, it would be ideal to deploy the mechanism in the selected applications. However, simulations are still necessary because real-world deployment are often costly and time-consuming.

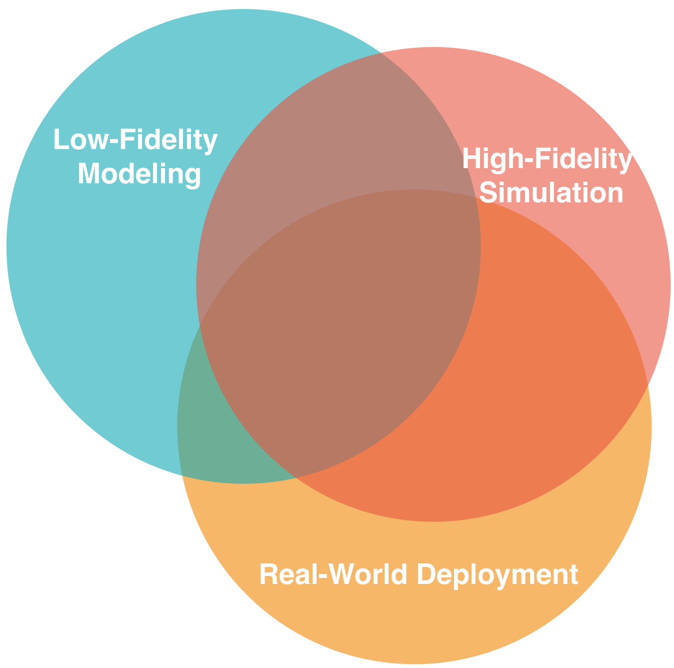


Figure 3.5: An illustration of different evaluation methods.

3.3 Applicability

The MDTA framework is most suitable to address conflicting scenarios when a stakeholder and individual agents have diverging interests or objectives in complex systems that consists of a large population of agents. The method can be applied to addressing traditional mechanism design problems such as mechanism design for social good, mechanism design for profit, as well as manipulation-resistant mechanism design such as countering false-name attacks, collusion and free riding.

Chapter 4

Information Design for Crowdfunding with Thresholding Agents

Crowdfunding has emerged as a prominent way for entrepreneurs to secure funding without sophisticated intermediation. In crowdfunding, an entrepreneur often has to decide how to disclose the campaign status in order to collect as many contributions as possible. Such decisions are difficult to make primarily due to incomplete information. In this chapter, we propose information design as a tool to help the entrepreneur to improve revenue by influencing backers' beliefs. We introduce a heuristic algorithm to dynamically compute information-disclosure policies for the entrepreneur, followed by an empirical evaluation to demonstrate its competitiveness over the widely-adopted immediate-disclosure policy. Our results demonstrate that the immediate-disclosure policy is not optimal when backers follow thresholding policies despite its ease of implementation. With appropriate heuristics, an entrepreneur can benefit from dynamic information disclosure. Our work sheds light on information design in a dynamic setting where agents make decisions using thresholding policies. This chapter is based on our work appeared in [Shen et al., 2018b].

4.1 Background

Crowdfunding reinvents the way that entrepreneurs raise external funding for implementing creative ideas. It has created a rapidly growing market that contributes an annual economic impact of tens of billions of US dollars globally [Yu et al., 2017]. Unfortunately, not all the crowdfunding campaigns are successful because most campaigns will get funded only if they have reached the fundraising goal within a deadline [Short et al., 2017]. In fact, less than 40% of the crowdfunding projects reach the targeted goals and receive the funds within the campaign deadlines [Short et al., 2017].

Mounting research has begun to investigate the determinants of the success of crowdfunding projects. Although there might be many factors (e.g., project descriptions [Marelli and Ordanini, 2016], product value [Agrawal et al., 2014], geography effect [Agrawal et al., 2011], reward details [Marelli and Ordanini, 2016], entrepreneurs' reputation [Kuppuswamy and Bayus, 2015] and the social network effect [Agrawal et al., 2014]) that influence a campaign's success, a recent study indicates that the number of donations made by early backers of a project is often the only difference between that project being funded or not [Solomon et al., 2015]. A substantial body of both theoretical analyses [Agrawal et al., 2014, Mollick, 2014, Alaei et al., 2016] and empirical evidence [Kuppuswamy and Bayus, 2015, Colombo et al., 2015, Marelli and Ordanini, 2016, Skirnevskiy et al., 2017] demonstrate that the amount of early contributions has a strong positive effect on the success of crowdfunding campaigns. These prior studies unanimously confirm the crucial role of early contributions in the success of crowdfunding projects.

There are two main reasons why the amount of early contributions matters. First, information about contributions received early in the campaign signals to potential backers the quality of the project, which in turn can trigger social learning behavior [Bandura, 1989] causing potential backers to also contribute to the campaign [Colombo et al., 2015]. An empirical study on a sample of 25,058 Kickstarter projects indicates that prospective backers usually make their pledging decisions based on how much of the project goal has already been funded by others [Kuppuswamy and Bayus, 2015]. Second, backers who have made an early contribution are likely to circulate the information of the project to their friends or families, which may attract additional contributions [Colombo et al., 2015, Skirnevskiy et al., 2017]. Both rationales indicate that it is of interest to entrepreneurs to attract as many contributions from early backers as possible.

In crowdfunding, backers are often reluctant to donate in the early days of a campaign due to high uncertainty [Mollick, 2014, Alaei et al., 2016, Kuppuswamy and Bayus, 2015, Solomon et al., 2015, Colombo et al., 2015]. A major source of uncertainty is the probability of success that the campaign will get funded (i.e., *Probability of Success, or PoS*) [Colombo et al., 2015, Kuppuswamy and Bayus, 2015]. Prospective backers are often uncertain about entrepreneurs' abilities to collect sufficient contributions to get the project funded. For instance, 64.12% of the crowdfunding projects in Kickstarter failed to reach the target goals [Kickstarter, 2017]. A backer [1] experiences a monetary or non-monetary opportunity cost if the fundraising goal is not achieved (and the project not funded), even if he is refunded upon the failure of the campaign [Alaei et al., 2016].

To attract as many early contributions as possible, an entrepreneur must take appropriate measures to coordinate backers' actions. To do this, the entrepreneur needs to have prior knowledge about the backers' arrival process, their valuation of the project (if funded), and how they estimate the probability that the campaign will be funded. However, none of this information is perfectly known to the entrepreneur. Thus, it is challenging for the entrepreneur to figure out what actions will make backers, especially early backers, be more willing to contribute.

[1] We will use "she" to denote an entrepreneur and "he" a backer/agent.

If conditions permit, the entrepreneur can manipulate backers' payoffs by offering appealing discounts to early backers that face high uncertainty [Ellman and Hurkens, 2015, Strausz, 2016]. The problem of devising allocation and payment schemes falls into the field of *mechanism design* [Nisan and Ronen, 1999]. While illuminating, it requires additional budgets and thus diminishes the entrepreneur's revenue [Ellman and Hurkens, 2015, Strausz, 2017]. Absent from sophisticated or even unrealistic assumptions of the backers' private types (e.g., valuation, arrival time, departure time), it is rather difficult or even unfeasible for the entrepreneur to implement effective mechanisms [Ellman and Hurkens, 2015, Strausz, 2017]. This is particularly the case in online settings where the entrepreneur has little knowledge about how the backers make their projections of the campaign's PoS.

Alternatively, the entrepreneur can improve backers' beliefs of the campaign's PoS by choosing what information backers see. In particular, the entrepreneur can and is permitted to voluntarily disclose the project status (i.e., how many contributions have been collected up to a given timestamp), a critical factor that influences backers' beliefs of the campaign's PoS [Kuppuswamy and Bayus, 2015, Alaei et al., 2016]. The problem of determining which pieces of information are disclosed to whom is called *information design* [Taneva, 2015].

Prior work on information design has generally assumed that backers' strategic behavior was perfectly rational and that games were well-defined (e.g., signaling games) [Bergemann and Pesendorfer, 2007, Taneva, 2015, Bergemann and Wambach, 2015, Bergemann and Morris, 2016, Alonso and Camara, 2016]. However, studies on consumer purchasing behavior show that buyers usually follow thresholding policies to decide whether to purchase goods or not [Kau and Hill, 1972, Kahneman, 2003, Zhou et al., 2005]. They often buy products when the prices are no more than their reserved values. This is particularly the case when consumers face high degrees of uncertainty and have little knowledge about the environment or the future, as frequently observed in clinical decision making [Pauker and Kassirer, 1980], crowdsourcing contests [Easley and Ghosh, 2015], airline ticket sales [Zhou et al., 2005], online shopping [Lee and Lin, 2005], management science [Su, 2007], societies of autonomous machines [Shen et al., 2017] and crowdfunding [Mollick, 2014, Alaei et al., 2016]. Under certain circumstances, thresholding policies are optimal policies and hence represent rational behavior [Ohannessian et al., 2014]. We thus consider the scenario where backers follow thresholding policies when they decide whether to contribute to a project or not.

In this paper, we study the information design problem in which an entrepreneur voluntarily reveals the project status to backers to influence their beliefs of the project's probability of success. Our work contributes to the state of the art in the following ways:

1. We show that excessive information disclosure weakly shrinks the entrepreneur's revenue. We identify conditions when immediate disclosure is optimal in crowdfunding when agents follow thresholding policies. We demonstrate that immediate disclosure is optimal if the funding goal has been achieved and if the project status increases monotonically by at least one contribution each time.

2. We introduce a heuristic algorithm called *Dynamic Shrinkage with Heuristic Selection (DSHS)* to to help the entrepreneur make decisions on information-disclosure policies.

3. Our experimental results demonstrate that the immediate-disclosure policy is not optimal when agents follow cutoff policies despite that it is more efficient to compute than the heuristic methods. Entrepreneurs can benefit from dynamic information design with appropriate heuristics.

4.2 Decision Making in Crowdfunding

This section first introduces key notations. It then formalizes backers' decision model and the entrepreneur's optimization problem.

4.2.1 Preliminaries

We consider discrete time $t \in \mathcal{T} = \{1, 2, 3, ..., T\}$, where T is the deadline for the campaign. Before launching the campaign, the entrepreneur must determine a fundraising goal G to get funded, a deadline T for reaching the goal, the number of rewards N, the minimal amount of contributions for a reward P, and a detailed description of the project such as motivation, product, milestones, and profiles of the team. All this information is fixed and is disclosed to all the backers. See Figure 4.1 for the procedure of a typical crowdfunding campaign.

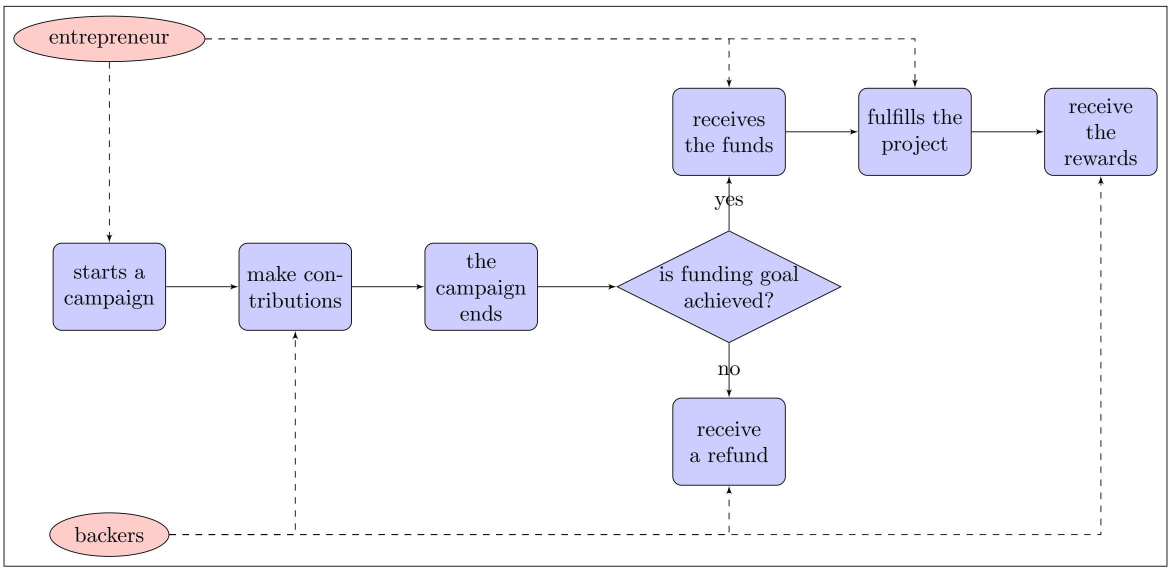


Figure 4.1: Procedure of a typical crowdfunding campaign.

After the campaign starts to accept contributions, backers arrive at the campaign sequentially with at most one each time. This is without loss of generality because batch arrivals can be viewed as a special case where the time interval is minimal [Shen et al., 2016, Alaei et al., 2016]. Let $b(t) \in \{0, 1\}$ denote the number of arrivals at time $t \in \mathcal{T}$.

At the beginning of time t, the entrepreneur discloses the state of the campaign (i.e., *project status*) $s(k)$ to each backer i that is in the campaign. Here, $s(k) = (S_k, k)$ where S_k refers to the percentage of funds that have been raised up to time $k \leq t$ (k not included), with respect to the fundraising goal G. For simplicity, let $|s(k)| = S_k$. We denote the entrepreneur's decision on information disclosure for backer i at time t by: $d(i,t) = (s(k), t)$ s.t. $k \in \mathcal{T}, k \leq t$. Here, $|s(1)| = 0$. The disclosed project status $s(k)$ must reflect the true state of the project at time k, which is enforced by the crowdfunding platform. In real-world crowdfunding campaigns, entrepreneurs are allowed to voluntarily disclose truthful project status [Solomon et al., 2015, Alaei et al., 2016]. In our work, we assume that any information about the project status observed by the backers is directly revealed by the entrepreneur. Future work should address the scenarios when backers have exogenous information due to information contagion [Arthur and Lane, 1993].

4.2.2 Backers' Decision Model

It is widely known that backers' beliefs of PoS are usually correlated with the entrepreneur's updates of the project status [Kuppuswamy and Bayus, 2015, Marelli and Ordanini, 2016, Alaei et al., 2016, Kuppuswamy and Bayus, 2017]. However, the exact correlation is privately known to a backer himself only and not observed by the entrepreneur. This makes it difficult to accurately model backers' decision making process. To tackle this problem, we formalize backers' decision model using the same pattern as that in the work [Ohannessian et al., 2014] by Ohannessian et al., which also assumes that agents use thresholding policies.

Let $r_i(t, d(i,t)) \in [0,1]$ represent high-value backer i's estimate of the campaign's PoS given the report $d(i,t)$, and $\phi_i \in (0,1]$ be his threshold on $r_i(\cdot)$ to contribute. We denote backer i's decision on whether to contribute or not at time t by $\alpha_i(t) \in \{0,1\}$, where 0 indicates *Not Pledging*, and 1 represents *Pledging*. Backer i's expected utility u_i is determined as follows:

$$u_i(t, \alpha_i(t), d(i,t)) = \begin{cases} c_i \cdot \alpha_i(t) \ , & \text{if } r_i(t, d(i,t)) \geq \phi_i; \\ 0 \ , & \text{otherwise.} \end{cases} \tag{4.1}$$

Here, $c_i > 0$ is backer i's expected utility if he contributes (i.e., $\alpha_i(t) = 1$) when his estimate of PoS is no less than the threshold ϕ_i. Note that c_i, $r_i(\cdot)$ and ϕ_i are all private information known to backer i only, while his arrival and pledging behavior are observed by the entrepreneur through the platform. In practice, backer i may adapt to the environment and update his threshold accordingly. In this case, his threshold ϕ_i can be treated like the upper bound of all the updated thresholds. Without loss of generality, we assume that each contributing backer pledges the same amount P of fund to the project for a reward.

Backer i stays at the campaign for at most $l_i \in \{1, 2, 3, .., L\}$ periods, where l_i is known to backer i only. This is without loss of generality because although backers may dynamically enter and exit the system and check the progress, these situations can be viewed as the case that the backers stay in the system for a sufficient period.

Let $\mathcal{I}(t)$ denote the group of backers who have arrived at the campaign before or at time t, have at least one time period to leave and have not yet claimed a contribution. At time t, for each backer $i \in \mathcal{I}(t)$, his objective function is

$$B_i(t) = \max_{\alpha_i(t)} u_i(t, \alpha_i(t), d(i,t)) \text{ s. t. } G, T, N, P, l_i \,, \tag{4.2}$$

where u_i is determined by Equation 4.1. At time t, backer i will leave the campaign either if he claims a contribution (i.e., $\alpha_i(t) = 1$) or his own deadline l_i is reached.

4.2.3 The Entrepreneur's Optimization Problem

In crowdfunding, the entrepreneur is interested in attracting as many contributions as possible within the deadline so that her project will get funded. Specifically, her objective is to set the disclosure policy such that the number of contributions is maximized until a given deadline T.

Let $M(t)$ denote the funds that the entrepreneur has raised up to time t (t included) when she uses the disclosure policy $DP(t)$. Here, $DP(t) = (((d(i,t'))_{i \in \mathcal{I}(t')})_{t' \leq t}$. The entrepreneur's expected contributions at time t is defined as follows:

$$M(t) = \sum_{t'=1}^{t} \sum_{i \in \mathcal{I}(t')} \alpha_i(t') \cdot P \ \text{ s. t. } G, T, N, P \ . \tag{4.3}$$

Due to the deadline constraint, the entrepreneur's optimization problem (i.e., optimal information design) is formalized as follows:

Definition 4.1 (Optimal Information Design)**.** An optimal information design in crowdfunding is to find a disclosure policy $DP_{opt}(T)$, such that $M(T)$ is maximized, i.e., $DP_{opt} = \underset{DP(T)}{\operatorname{argmax}} M(T)$.

Due to the dynamic nature of crowdfunding, the design of disclosure policy $DP(t)$ cannot be based on backers' later decisions $(\alpha_j(\hat{t})_{j \in \mathcal{I}(\hat{t})})_{\hat{t} > t}$, or use later project status $(s(\hat{t}))_{\hat{t} > t}$. This constraint is called *No Clairvoyance*.

4.3 Optimal Information Design

After introducing the solution concepts, we show that excessive information weakly shrinks revenue. We further identify conditions under which immediate disclosure is optimal in crowdfunding.

4.3.1 Solution Concepts

In his seminal work [Blackwell et al., 1953], Blackwell formulated a partial order that compares the quality of two pieces of information (see Theorem 4.1). According to Blackwell's theorem, if a piece of information ζ_2 is Blackwell-inferior to ζ_1, then an agent will always weakly prefer ζ_1 to ζ_2.

Theorem 4.1 (Blackwell's theorem [Blackwell et al., 1953])**.** *Let ζ_1 and ζ_2 represent two pieces of information, the following conditions are equivalent:*

1. *When the agent chooses ζ_1, her expected utility is always at least as big as the expected utility when she chooses ζ_2, independent of the utility function and the distribution of the input.*

2. *ζ_2 is a garbling of ζ_1.*

Blackwell's theorem implies two types of information: vertical information and horizontal information, which are key solution concepts used in this work. Given two pieces of information, if one is always (weakly) preferred whatever the information receivers' types are, then they are *vertical* information (see Definition 4.2). If the two pieces of information are not comparable without prior knowledge about the receivers' types, then the information is *horizontal* (definition omitted since it complements vertical information).

Definition 4.2 (Vertical Information)**.** Given two pieces of information ζ_1 and ζ_2, where $\zeta_1 \neq \zeta_2$, if $\forall i \in \mathcal{I}$, $u_i(\zeta_1) \geq u_i(\zeta_2)$, then $\zeta_1 \succsim \zeta_2$, where $\mathcal{I}$ denotes a set of agents and $\succsim$, indicating *preferred or indifferent to*, is independent of agent i's private type. If $\zeta_1 \succsim \zeta_2$, the information is vertical.

In crowdfunding, backer i's estimate of the campaign's PoS (i.e., $r_i(t, d(i,t))$) is both time and state-dependent. Both the amount of funding (measured by $|s(k)|$) raised, and the time of the project status (denoted by k) are important. We identify three scenarios of vertical information. First, a higher state of project status is always more favorable if the time of the state is the same (e.g., $(20\%, 5) \succsim (10\%, 5)$), which is obvious. Second, the earlier report of project status is always (weakly) preferred if the project status of the two reports are the same (see Proposition 4.1). Third, the later report of project status is always (weakly) preferred if the revenue increases by more than P each time between the timestamps of the two statuses. (see Proposition 4.2). All the proofs can be found in Appendix A.1.

Proposition 4.1. *An earlier report of project status is always weakly preferred if the project status of the two reports are the same. Formally, given project status* $s(k_1)$ *and* $s(k_2)$*, where* $k_1 < k_2$, $|s(k_1)| = |s(k_2)|$*, we have* $\forall t \geq k_2, \forall i \in \mathcal{I}(t)$*:* $(s(k_1), t) \succsim (s(k_2), t)$.

Proposition 4.2. *A later report of project status is always weakly preferred if the revenue increases by more than* P *each time between the period of the two statuses. Given status reports* $\varepsilon_1 : (s(k_1), t_1)$ *and* $\varepsilon_2 : (s(k_2), t_2)$*, where* $k_1 < k_2, t_1 < t_2$ *and* $|s(k_2)| - |s(k_1)| \geq (k_2 - k_1) \cdot P/G$*, we have* $\forall t \geq k_2, \forall i \in \mathcal{I}(t)$*:* $\varepsilon_2 \succsim \varepsilon_1$.

If the conditions of vertical information cannot be identified, the information is horizontal (see Example 4.1). Without prior information about backers' private types (e.g., arrival process, valuation, the estimate of the campaign's *PoS*, and the correlation between them), it is not feasible for the entrepreneur to identify optimal information design. However, an effective disclosure policy should capture both the vertical and horizontal component, making the information design problem particularly challenging for the entrepreneur.

Example 4.1. Given $T = 30, P = 0.1G, |s(10)| = 30\%$ and $|s(15)| = 40\%$, without prior knowledge about backer i's projection of *PoS* (i.e., $r_i(\cdot)$), it is unclear which project status is more favorable by i. This is because $|s(15)| - |s(10)| = 0.1 < 0.1 \cdot (15 - 10)$.

4.3.2 Excessive Disclosure Shrinks Revenue

Given two project status reports, if their partial order can be identified according to Proposition 4.1 and 4.2, then the entrepreneur only needs to disclose the one with higher order.

This is because revealing the low-order report does not increase the chance that backers contribute to the campaign (see Lemma 4.1).

Lemma 4.1. *If the order of two project status reports can be identified, the low-order report does not increase the change of backers' contribution. Formally, given two reports* $\varepsilon_1 : (s(k_1), t_1)$ *and* $\varepsilon_2 : (s(k_2), t_2)$*, if* $\forall t \geq \max\{k_1, k_2\}, k_1 \neq k_2, \forall i \in \mathcal{I}(t) : \varepsilon_2 \gtrsim \varepsilon_1$*, we have :* $E(\alpha_i = 1|\varepsilon_2) \geq E(\alpha_i = 1|(\varepsilon_1, \varepsilon_2))$ *, where* $E(\alpha_i = 1|\varepsilon)$ *denotes the expectation that backer* i *contributes to the campaign given the information* ε*.*

If the partial order of the two reports cannot be identified, the entrepreneur should also refrain from disclosing additional information. Depending on how backers estimate the campaign's *PoS*, revealing more information than necessary can decrease the revenue. The reason is that excessive information disclosure can decrease backers' projections of the *PoS* (See Lemma 4.2).

Lemma 4.2. *If the partial order of the two project status reports cannot be identified, excessive information disclosure weakly decrease backers' projections of PoS. Formally, given two reports* $\varepsilon_1 : (s(k_1), t_1)$ *and* $\varepsilon_2 : (s(k_2), t_2)$ *where* $k_1 \neq k_2$*, if the partial order of the two cannot be identified by the entrepreneur, then* $\forall t \geq \max\{k_1, k_2\}, \forall i \in \mathcal{I}(t)$*, we have:* $r_i(t, (\varepsilon_1, \varepsilon_2)) \leq \max\{r_i(t, \varepsilon_1), r_i(t, \varepsilon_2)\}$.

In our model of crowdfunding, excessive information disclosure weakly diminishes a backer's willingness to contribute (see Theorem 4.2) and thus weakly shrinks the revenue as well as the entrepreneur's ability to implement optimal information-disclosure policies. The entrepreneur should not disclose more information about the project status than necessary in order to collect as many contributions as possible.

Theorem 4.2. *Excessive information disclosure weakly diminishes the chance that a backer will contribute. Formally, given two project status reports* $\varepsilon_1 : (s(k_1), t_1)$ *and* $\varepsilon_2 : (s(k_2), t_2)$

where $k_1 \neq k_2$, let $\alpha_i^{'}$ denote backer i's decision on pledging if given report either ε_1 or ε_2, and $\alpha_i^{''}$ denote his decision on contribution if given $(\varepsilon_1, \varepsilon_2)$. $\forall t \geq \max\{k_1, k_2\}, \forall i \in \mathcal{I}(t)$, we have: $E(\alpha_i^{'} = 1) \geq E(\alpha_i^{''} = 1)$.

4.3.3 Immediate Disclosure is not Always Optimal

The immediate-disclosure policy (see Definition 4.3) is widely adopted by entrepreneurs on major crowdfunding platforms (e.g., *Kickstarter*, *Indiegogo*) due to its ease of implementation [Alaei et al., 2016]. It is thus important to investigate if immediate disclosure is optimal.

Definition 4.3 (Immediate Disclosure)**.** An immediate-disclosure policy always reveals the current project status to all the backers in the campaign. Formally let DP_{im} denote immediate disclosure, we have $DP_{im}(t) = (((d(j,t'))_{j \in \mathcal{I}(t')})_{t' \leq t}$ s.t. $d(j,t') = (s(t'), t')$.

If the entrepreneur and the backers have identical information, immediate disclosure is optimal [Rayo and Segal, 2010, Kamenica and Gentzkow, 2011, Au, 2015]. It still holds if the entrepreneur has some unique information provided that such information does not affect the backers' decisions. Unfortunately, in our model, all information about the campaign (e.g., G, T, P, N) except the project status is known to both the entrepreneur and the backers. Backer i's estimate of PoS (i.e., $r_i(t, d(i,t))$) is influenced by the project status $s(k)$ that the entrepreneur reveals. Thus, it is critical for the entrepreneur to identify conditions when immediate disclosure is optimal.

From Proposition 4.2, we see that to improve backer i's belief of the campaign's PoS, the entrepreneur should always disclose the project status that is preferred by all the backers if available. Otherwise, the information design is not optimal. With this intuition in mind, we show that before the campaign reaches the fundraising goal, immediate disclosure is optimal if and only if the project status increases monotonically in time by at least one contribution each time (see Lemma 4.3). This condition characterizes the relationship between the growth rates of the revenue and the maximum possible arrival rate of the backers. Though the entrepreneur does not have prior knowledge of the backers' types (e.g., beliefs, thresholds), she can observe the progress of the project and determine if immediate disclosure is optimal given the tracking record of project status.

Lemma 4.3. *Before the campaign reaches the fundraising goal, immediate disclosure is optimal if and only if the project state increases monotonically in time by at least one contribution each time. Formally, if $M(t) < G$, then we have:*
$DP_{im}(t) = \operatorname*{argmax}_{DP(t)} M(t) \iff \forall t' \leq t : |s(t')| - |s(t'-1)| \geq P/G.$

After successfully reaching the funding objective, it is certain that the campaign will get funded, so immediate disclosure is optimal (see Lemma 4.4).

Lemma 4.4. *After the campaign reaches the fundraising goal, Immediate disclosure is optimal. Formally, if $M(t) \geq G$, then we have: $DP_{im} = \operatorname*{argmax}_{DP(t)} M(t)$.*

Lemma 4.3 shows that immediate disclosure is not always optimal during the crowdfunding campaign when backers follow a thresholding policy. According to Definition 4.1 and Equation 4.3, in order to compute the optimal solution, the entrepreneur must have a prior knowledge of the sequence of decisions $((\alpha_i(t))_{i\in\mathcal{I}(t)})_{t\in\mathcal{T}}$ in advance. However, such assumption violates the *No Clairvoyance* constraint and is not implementable in practice.

4.4 Dynamic Information Design

Instead of restricting our attention to optimal information design, we introduce a heuristic algorithm, called *Dynamic Shrinkage with Heuristic Selection (DSHS)*, to help the entrepreneur make decisions on information disclosure (See Fig. 4.2 for an illustration of DSHS).

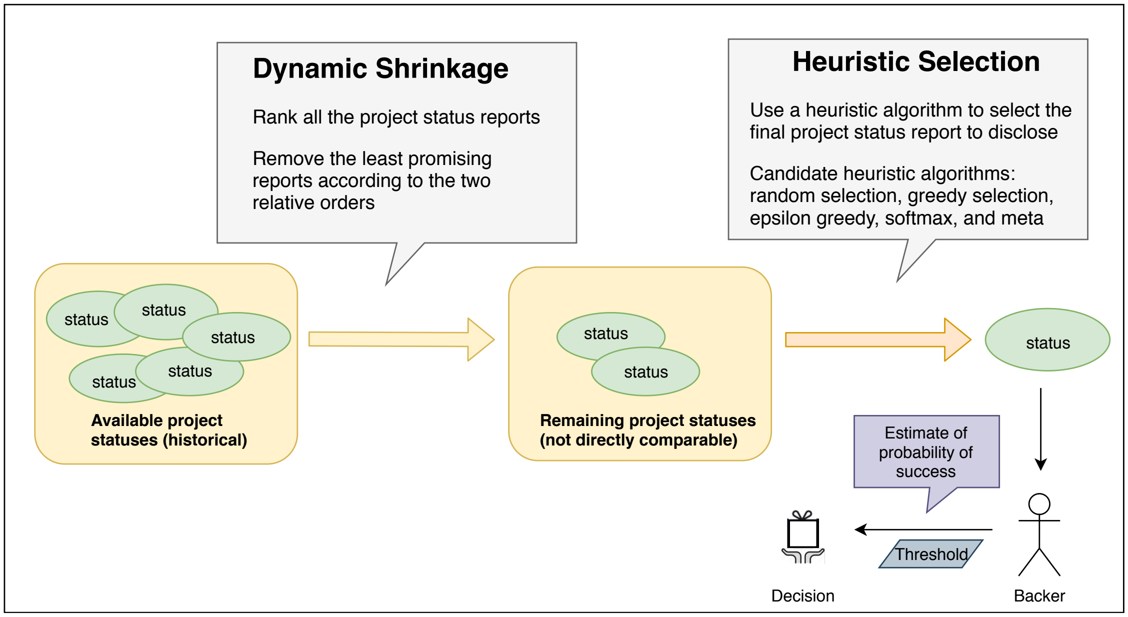


Figure 4.2: An illustration of the Dynamic Shrinkage with Heuristic Selection algorithm.

DSHS treats the two conditions separately: before and after project success (see Algorithm 4.1). Before the campaign reaches the fundraising goal, the algorithm determines the disclosure policy according to two processes: *dynamic shrinkage* (see Algorithm 4.2) and *heuristic selection*. After the campaign reaches the fundraising goal (if it happens), the algorithm discloses information immediately.

Dynamic Shrinkage. In the dynamic-shrinkage process, *DSHS* ranks all the available choices, and removes the least promising choices which are less preferred by the backers according to Propositions 4.1 and 4.2. By doing so, the entrepreneur avoids excessive information disclosure that weakly shrinks revenue.

Initially, *DSHS* includes all the project status disclosures $s(k)$ since the last disclosure for backer i into a set $H_i(t)$. That is,

$$H_i(t) \leftarrow \{s(k)\}_{k\in\{k_0,k_0+1,\ldots,t\}} \text{ s.t. } d(i,t') = (s(k_0),t') \ , \tag{4.4}$$

Algorithm 4.1 Dynamic Shrinkage with Heuristic Selection

Input: t - time; $s(t)$ - project status at time t; $\mathcal{I}(t)$- backers in the campaign.
Output: $(d(i,t))_{i \in \mathcal{I}(t)}$- the entrepreneur's decisions on information disclosure for backers in the campaign at time t.

if $t \leq T$ **then**
    for each backer $i \in \mathcal{I}(t)$ **do**
        if current revenue $M(t) < G$ **then** ▷ before success
            Include all the available project status into $H_i(t)$
            Sort $H_i(t)$ in the ascending order of $|s(k)|$
            Remove the least promising candidates in $H_i(t)$
            Select the project status $s(k_{sel})$ using heuristics
            Finalize disclosure decision $d(i,t) \leftarrow (s(k_{sel}), t)$
        else ▷ after success
            Disclosure current status, i.e., $d(i,t) \leftarrow (s(t), t)$
        end if
        Update revenue $M(t+1) = M(t) + \alpha_i(t) \cdot P$
        Update project status $s(t+1) = M(t+1)/G$
    end for
end if

where $t' \in \{1, 2, ..., t\}$. It then sorts $H_i(t)$ in the ascending order of $|s(k)|$. This sorting problem can be easily solved by *Quicksort* [Hoare, 1962] with time complexity $O(|H_i(t)| \log |H_i(t)|)$. Since $|H_i(t)| \leq T$, the worst-case complexity for the function is $O(T \log T)$.

After the sorting process, *DSHS* removes the least promising candidates through the function *Shrink* (see Algorithm 4.2). While there are at least two choices available, the *Shrink* algorithm removes the project status with later time if the two statuses have the same progress (see line 4, Algorithm 4.2) according to Proposition 4.1. This process is equivalent to removing duplicates in a sorted array, which can be solved in $O(T)$ time. Given two project statuses, if they satisfy the relation in Proposition 4.2, then the algorithm removes the project status with the earlier time (see line 7, Algorithm 4.2) . This step takes $O(T \log T)$ time in the worst case. The algorithm does nothing if only one disclosure strategy exists.

Algorithm 4.2 Shrink

Require: H- sorted project status disclosures
Ensure: H'-remaining status disclosures after shrinkage

```
if |H| ≥ 2 then
    while s(k_1), s(k_2) ∈ H, k_1 < k_2 do
        if |s(k_1)| = |s(k_2)| then
            H ← H \ {s(k_2)}                          ▷ By Proposition 4.1
        end if
        if |s(k_2)| − |s(k_1)| ≥ (k_2 − k_1) · P/G then
            H ← H \ {s(k_1)}                          ▷ By Proposition 4.2
        end if
    end while
end if
H' ← H
```

Heuristic Selection. After the shrinkage process, if there are still at least two choices available (i.e., $|H_i(t)| \geq 2$), then the remaining set $H_i(t)$ is horizontal. The entrepreneur has to select some $s(k_{sel}) \in H_i(t)$ to attract as many contributions as possible. This optimization problem is similar with the renowned *restless bandit problem* [Whittle, 1988], which is not solvable due to incomplete information. However, simple heuristics such as *random selection*, *greedy selection*, *ϵ-greedy exploration*, and *softmax exploration* can be used to produce acceptable results. See Appendix A.2 for details of each algorithm.

We further introduce a meta algorithm (See Algorithm 4.3). The intuition is that the algorithm can improve the quality of decisions by only using the experts that have a satisficing performance for producing the final results [Crandall, 2014]. Besides, we take an ensemble approach to calculate the final selection instead of directly applying the results produced by the selected experts. The benefit is that the algorithm can further reduce potential performance loss due to biases of a single individual expert [Shen et al., 2013, Kuncheva and Whitaker, 2003].

Before describing the meta algorithm, we first introduce the notations used. Let X denote the set of experts, where $x \in X$ is one of the four heuristics. We write $z^t(x)$ for expert x's expected revenue at time t and write $w^t(x)$ for expert x's revenue prospect. Here, $z^t(x)$ is computed by:

$$z^t(x) = \sum_{s(k) \in H'} \sum_{i \in \mathcal{I}(t)} \rho_x^t(s(k)) \cdot \Upsilon_x(s(k), i, t) \ , \tag{4.5}$$

where $\rho_x^t(s(k))$ denotes the probability that $s(k)$ is selected as the targeted project status by expert x at t, and $\Upsilon_x(s(k), i, t)$ is the entrepreneur's expected increase of revenue given $(s(k), t)$ for backer $i \in \mathcal{I}(t)$ by using expert algorithm x. Details of computing $\Upsilon_x(s(k), i, t)$ for each expert x is described in Appendix A.2. Initially, $w^t = \max_{x \in X}\{z^t(x)\}$.

Each time the algorithm selects a subset X' of experts whose expected revenue is higher than a learned threshold—the minimum learned prospect $w^t(x)$. This step eliminates the experts that fail to produce better expected revenue than the threshold. The algorithm then performs a majority vote from the results generated by each expert $x \in X'$. The selected project status $s(k_{sel})$ is the one with the most votes. Ties are broken by choosing the result generated by the expert with the highest $z^t(x)$. This step aims to improve the robustness of selection by reducing the performance loss caused by biases of a single expert.

When a new expert algorithm is selected, the prospect for the expert is updated by

$$w^t(x) = (1 - \sigma) q^t(x) + \sigma w^{t-\delta}(x) \ . \tag{4.6}$$

Here, δ is the number of periods that expert x has been used, and $\sigma \in [0, 1]$ is the learning rate ($\sigma = 0.9$ in our paper). $q^t(x)$ is the entrepreneur's average revenue gain per time by using expert algorithm x in the last δ periods. It is calculated by:

$${}^t(x) = \sum_{t'=t-\delta}^{t} \sum_{j \in \mathcal{I}(t')} \frac{\alpha_j(t')}{\delta} \ . \tag{4.7}$$

Algorithm 4.3 Meta

Require: H'-remaining status disclosures after shrinkage; X-the set of experts
Ensure: $s(k_{sel})$-the selected project status disclosure

```
Compute z^t(x) for x ∈ X
Initialize w^t(x) = max_{x∈X} z^t(x)
while t < T do
    X' = {x : z^t(x) ≥ min w^t(x)}
    Perform a majority vote for s(k) ∈ H_{X'}
    Select s(k_sel) as the s(k) with the majority rule
    Update w^t(x) = (1 − σ)q^t(x) + σw^{t−δ}(x) if a new expert x is selected
    Update q^t(x) and z^t(x) for each x ∈ X
end while
```

DSHS is highly flexible in the sense that it allows the entrepreneur to easily customize both the shrinkage process and the selection process with different methods.

4.5 Empirical Evaluation

This section describes the experimental settings and the results.

Experimental Setup. We collected the campaign data from a randomly selected subset of Kickstarter[2] projects using a web crawler. The dataset contains 1,569 projects which satisfy the following conditions: (1) they were all-or-nothing, reward-based campaigns; (2) the campaigns lasted for exactly 1,440 hours (60 days) with both the starting time and the ending time falling between 07/15/2016 and 10/15/2016. Each campaign includes hourly project status, the fundraising goal, the deadline, the minimal amount of contribution, and the number of contributions every hour. The data samples allow us to mimic the operation of real-world crowdfunding projects when the underlying factors and correlations that impact them are yet to be identified [Mollick, 2014, Alaei et al., 2016].

Most of the projects on Kickstarter offer several tiers of perks for the backers to choose. We only selected the early bird pledges and the regular pledges that would offer a product to the backers to compensate backers' financial support. We adjusted the funding goal for each project accordingly. The early bird pledges were proportioned to the regular pledges. For instance, an early bird pledge with value 8 is equivalent to 0.8 regular pledge with value 10.

Due to Kickstarter's API constraints, we were unable to track the number of backers who visited the campaign per hour. We simulated backers' arrivals using *Poisson* [Ross, 1996] distribution for each project. The arrival process was independently and identically distributed with a mean $\vartheta(t)$ across time, where $t \in \mathcal{T}$. The mean of $(\vartheta(t))_{t\in\mathcal{T}}$ was 0.1 (consistent with the empirical arrival rates of backers in crowdfunding projects [Marwell, 2015]).

[2] https://www.kickstarter.com

We used the anticipating random walk [Alaei et al., 2016] model for simulating backers' projections of *PoS* because it is tailored for computing backers' estimates of *PoS* in crowdfunding. For each project, the backers' valuation of a reward was *Gaussian* [Ross, 1996] distributed with a mean equivalent to the value of the reward P and a randomly selected standard deviation ranging 0.05 of the mean to 0.5 of the mean.

We performed six groups of experiments: immediate disclosure (*immediate*), and *DSHS* with five heuristics (*random*, *greedy*, ϵ-*greedy*, *softmax*, and *meta*) . Each group was run 30 times with the same 2.9GHz quad-core machine.

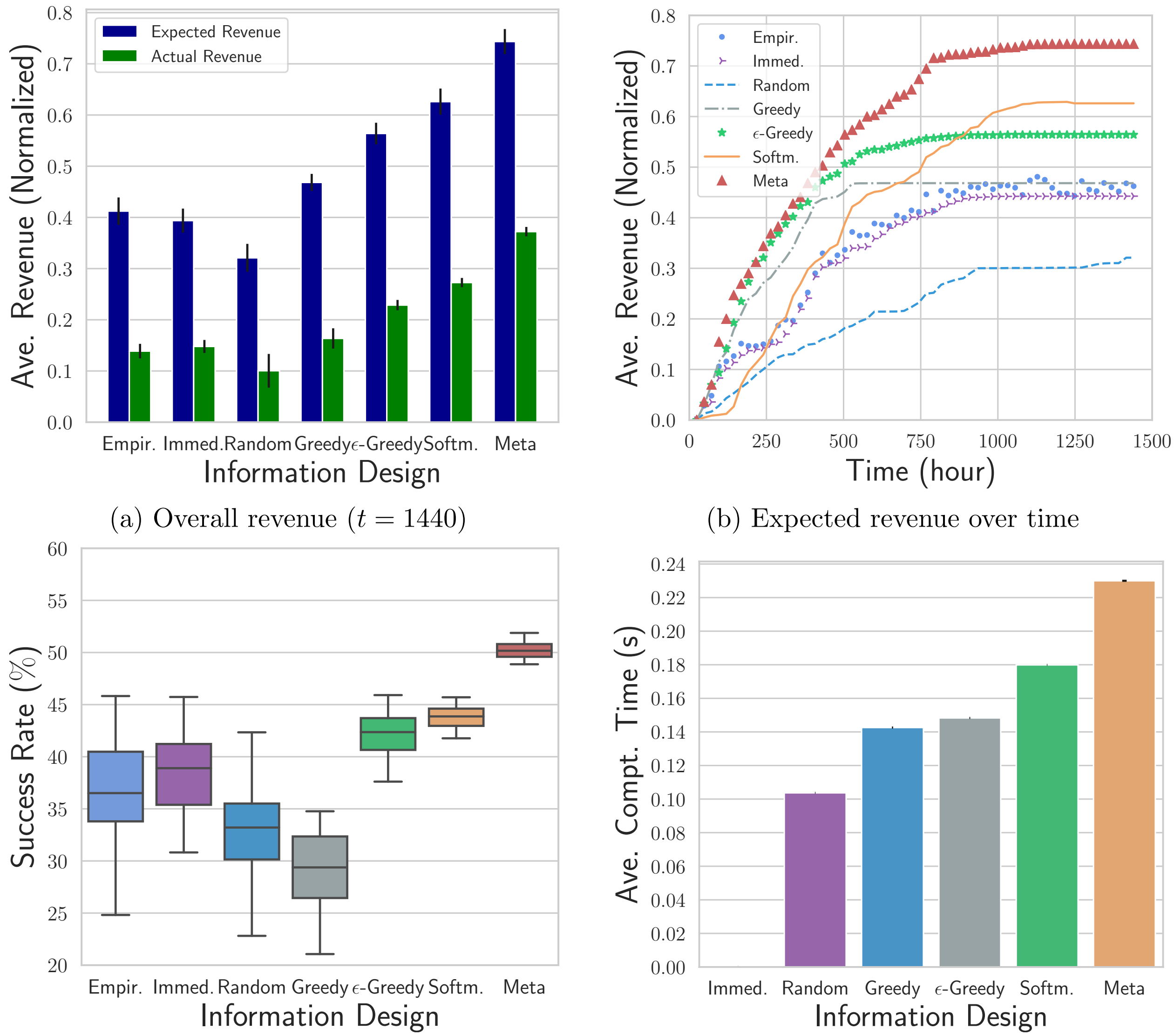
Results. Figure 4.3a shows the average revenue (normalized by the highest revenue achieved in all experiments) obtained in the end by each group. The actual revenue excluded the projects that failed ($M(T) < G$), while the expected revenue included all the projects regardless of whether they succeeded to meet the funding goal or not. Not surprisingly, the expected revenue of each group was significantly higher than their respective actual revenue. This is because the majority of the campaigns failed due to not having met the funding goal within the deadline (see Figure 4.3c). Among the six groups, the *meta* group scored the best for both the expected revenue (mean = 0.7435, std = 0.0244) and the actual revenue (mean = 0.3722, std = 0.0092), followed by the *softmax* group, and the ϵ-*greedy* group. The *greedy* group and the immediate group performed better than the *immediate* group in the expected revenue, but not in the actual revenue due to a lower success rate. The *random* group received the lowest scores in terms of both the expected revenue (mean = 0.3210, std = 0.0273) and the actual revenue (mean = 0.1004, std = 0.0329).

At the beginning, the *immediate*, the *meta*, the *greedy*, and ϵ-*greedy* groups performed better than the other two (see Figure 4.3b). As time progressed, the *meta* and the ϵ-*greedy* groups continued to lead the way until the later left behind the former at around $t = 400$. The

ϵ-*greedy* group kept the second until at time $t = 900$ that it was surpassed by the *softmax* group. One explanation is that the ϵ-greedy algorithm initially encouraged exploration to a higher degree than the softmax algorithm. However, it acted more greedily than the *softmax* exploration over time, which was not favorable since better choices were rarely explored. The *meta* group used a set of refined policies to produce more robust decisions than the others. The *random* group performed the worst possibly because the random algorithm completely ignored the history of backers' responses.

The *meta* algorithm took the most computation time (mean = 0.2300 std = 0.0290), while the *random* method required the least time (mean = 0.1037, std = 0.0034) (see Figure 4.3d). Immediate disclosure required no additional time.

In summary, although the *meta* group required the most computation time, it performed consistently the best among all the groups in terms of both actual and expected revenue. This echoes our previous findings that immediate disclosure is not always optimal in crowdfunding.



(a) Overall revenue ($t = 1440$)

(b) Expected revenue over time

(c) Project success rate (%)

(d) Computation time

Figure 4.3: A comparison of performance.

4.6 Summary

In this paper, we present the very first study on information design where a sender interacts with multiple receivers that follow thresholding policies. Our work demonstrates that excessive information disclosure weakly shrinks the revenue in crowdfunding when backers use cutoff policies. It further shows that the widely-adopted immediate-disclosure policy is not optimal. We also present how the entrepreneur can benefit from dynamic information disclosure with appropriate heuristics. To further evaluate the performance of the heuristic algorithm, user studies and real-world deployment are needed.

Although our analysis is in the context of crowdfunding, extensions to other domains (e.g., transportation systems, smart grids, and online shopping) where agents typically use thresholding policies can be straightforward. For instance, online shopping marketplace can employ DSHS variants to dynamically reveal the number of products available or the number of products sold to attract potential buyers to buy the products. Further research is required to assess the variants' performance in these domains.

Chapter 5

A Post-Price Online for Autonomous Mobility-on-Demand Systems

With proper management, Autonomous Mobility-on-Demand (AMoD) systems have great potential to satisfy the transport demand of urban populations by providing safe, convenient, and affordable ridesharing services. Meanwhile, such systems can substantially decrease private car ownership and use, and thus significantly reduce traffic congestion, energy consumption, and carbon emissions. To achieve this objective, an AMoD system requires private information about the demand from passengers. However, due to self-interestedness, passengers are unlikely to cooperate with the service providers in this regard. Therefore, an online mechanism is desirable if it incentivizes passengers to truthfully report their actual demand. In this chapter, we hereby introduce a posted-price, integrated online ridesharing mechanism (IORS) that satisfies desirable properties such as ex-post incentive compatibility, individual rationality, and budget-balance. Numerical results indicate the competitiveness of IORS compared with two benchmarks, namely the optimal assignment and an offline, auction-based mechanism. This chapter is based on our work appeared in [Shen et al., 2016].

5.1 Background

The rise of private car ownership and use has brought many social and environmental challenges, including traffic congestion, increased greenhouse gas emissions [Poudenx, 2008]. One possible solution to address the challenges is to promote ridesharing [Furuhata et al., 2013, Caulfield, 2009, Levofsky and Greenberg, 2001] among passengers by providing incentives (e.g., lower fares for the shared trips than individual trips) to them. In such a scenario, a limited number (depending on the seat capacity of the vehicle) of passengers who have similar itineraries share a ride and split the fares. Ridesharing (as shown in Figure 5.1) increases the occupancy of vehicles during traveling, making it possible to transport more passengers with fewer vehicles running on the roads.

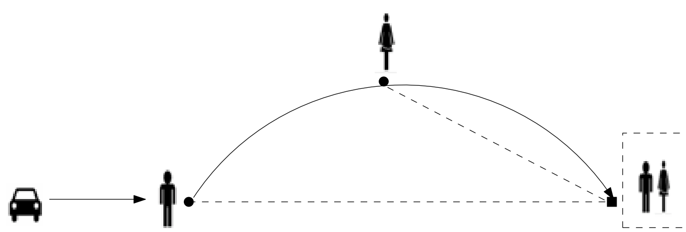


Figure 5.1: A simple scenario of ridesharing: dashed lines indicate the real demand; solid lines with arrow indicate actual routes.

Rideshare experiences can be significantly improved as autonomous vehicle technologies mature. This is because AMoD systems have the potential to provide safe, convenient, and affordable mobility solutions for passengers, while reducing greenhouse emissions and private car ownership [Mitchell, 2010, Chong et al., 2013, Shen and Lopes, 2015, Spieser et al., 2014]. Unlike traditional mobility-on-demand systems (e.g., taxis, shuttles), an AMoD system is equipped with a fleet of self-driving, electric cars with no drivers needed. This enables seamless cooperation between the information center and the autonomous vehicles (AVs).

Several mechanisms [Kleiner et al., 2011, Kamar and Horvitz, 2009, Cheng et al., 2014, Zhao et al., 2015] have been introduced to promote ridesharing in traditional mobility-on-demand systems. These mechanisms require passengers to directly reveal the valuation of the rides.

While interesting and insightful from a theoretical perspective, they may not work well in practice since they require passengers to reveal the exact value of their rides, which could be problematic [Naor et al., 1999, Larson and Sandholm, 2001, Babaioff et al., 2015]. In this case, posted-price mechanisms are more appealing because passengers only need to accept offers with value greater than the posted price, without revealing their actual valuations to service providers.

Some of the mechanisms require additional constraints (e.g., dual ride shares only, linear in commitment) to satisfy desirable properties such as strategy-proofness and budget-balance [Kleiner et al., 2011, Zhao et al., 2015]. Besides, many mechanisms assume that passengers are only motivated by monetary incentives (i.e., lower fares) [Kleiner et al., 2011, Kamar and Horvitz, 2009, Cheng et al., 2014, Zhao et al., 2015]. They neglect the fact that non-monetary factors such as time, comfortability and privacy, are also important, or even critical when people make decisions on whether to use the service or not.

Another drawback of these mechanisms is that they process the ride requests in batch and do not work in online environment [Kleiner et al., 2011, Kamar and Horvitz, 2009, Cheng et al., 2014, Zhao et al., 2015]. In AMoD systems, service providers are committed to offering an immediate response to each request sent by passengers via a smart device. Besides, they assume that the demand is fixed without consideration of the dynamic nature of demand responsive systems.

A desirable mechanism is expected to be truthful and online [Parkes and Singh, 2004, Gallien, 2006, Nisan et al., 2007]. It should be able to provide a fare quote immediately after the submission of a request. It needs to consider major non-monetary factors (e.g., latest departure time) as well as the dynamic nature of demand-driven systems. However, such a mechanism is yet to be designed. To bridge the gap and transcend conventional transport models like private car ownership, we introduce a truthful online mechanism called IORS for AMoD systems. We implement a simple, abstracted, yet powerful simulator that enables efficient modeling of ridesharing in AMoD systems. Numerical results show that the IORS mechanism outperforms the cutting-edge auction-based mechanism for last-mile mobility systems [Cheng et al., 2014] substantially. It has a very close performance compared to the optimal solution, but requires a shorter time to compute and requires no future knowledge about the demand.

5.2 Ridesharing in AMoD Systems

An AMoD system (see Figure 5.2) can be viewed as a multi-agent system consisting of an information center, a fleet of autonomous vehicle agents, and self-interested passengers who dynamically enter and exit the system. The working principle of the AMoD system is straightforward: when a passenger needs a ride, she sends the ride request to the information center using a smart device. This initiates the demand for mobility. The information center next computes a fare quote and sends it to the passenger. If the passenger accepts the fare estimate, the information center then calculates an assignment. As long as a plan has been calculated, it will be sent to both the AV assigned and the passenger who has just submitted the request. Both the passenger and the AV are committed to executing it. Otherwise, the passenger will be subject to penalties.

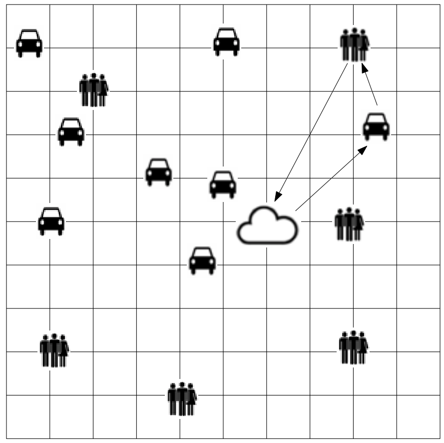


Figure 5.2: An abstraction of ridesharing in an AMoD system operating in a grid city.

5.3 The Online Mechanism

5.3.1 Preliminaries

In our work we consider discrete time $\mathcal{T} = \{0, 1, 2, ...\}$, with passengers that arrive and depart over time. Without loss of generality, we assume that the AVs never exit the AMoD system. The information center has full knowledge of the AVs at each time. However, passengers' demand information is private and hidden from the center. The mechanism designer should incentivize passengers to truthfully reveal their demand for better system-wide optimization.

In our model we make a realistic assumption that the passengers are impatient [Horn, 2002]. They will leave the AMoD system and switch to other alternatives if the pickup time is later than their latest departure time.

Let $\mathcal{I}^t$ denote a group of passengers who have mobility demand at time $t \in \mathcal{T}$. At each time, a passenger $i \in \mathcal{I}^t$ submits a request $r_i^t \in \mathcal{R}^t$ to the information center. The request r_i^t can be represented as a tuple $(o_i, d_i, t_i, \bar{t}_i)$, where o_i, d_i, t_i and $\bar{t}_i$ are passenger i's origin, destination, arrival time and latest departure time, respectively. Here, $t_i = t$. To quantify the transport demand, we introduce the effective demand ℓ_i, which indicates the minimum distance from passenger i's origin o_i and destination d_i. Assuming that the effective demand is independent of request submission time, we have:

$$\forall t \in \mathcal{T}, i \in \mathcal{I}^t, \quad \ell_i^{t+1} = \ell_i^t\ . \tag{5.1}$$

Equation 5.1 indicates that the effective demand of passenger $i \in \mathcal{I}^t$ will not change if the passenger delays its request submission from time t to time $t+1$.

Once a passenger i has sent her request r_i^t, the system needs to provide a fare estimate $q_{r_i^t}$ to the passenger immediately, which enables the passenger to make a prompt decision on whether to accept the quote or not. It is important to note that the quote is the upper bound of the fare rather than the final payment. The passenger only accepts the service if the quote is lower than the amount that she is willing to pay. If the passenger accepts the quote and if the system is able to provide the service given the time and routing constraints, she will be serviced with an assignment provided. A final payment $p_{r_i^t}$ will be calculated upon the completion of the ride. Otherwise, the request will be rejected.

In our model we make the passengers who use the ride service split the operational cost of the vehicles. This enables the AMoD system to provide service without seeking for external subsidies. The mechanism designer's goal is to minimize the cost per unit effective demand. Since the operational cost is split by all the passengers, this objective reflects the social welfare. The mechanism needs to collect truthful information about the requests from passengers. However, passengers are selfish and motivated to minimize their own cost for the rides. For instance, they may delay their request submissions, or claim a shorter waiting time to reduce their cost. Therefore, incentives should be provided to counter the manipulations. Let C^t denote the total cost of the system incurred up to now from $t = 0$, and $\mathcal{I}'^t \subseteq \mathcal{I}^t$ be the set of passengers been serviced at t. We have the cost per demand of the system:

$$W' = \frac{C^t}{\sum_{t' \in [0,t]} \sum_{i \in \mathcal{I}'^{t'}} \ell_i} \ . \tag{5.2}$$

Let $W = 1/W'$, the mechanism designer's goal is equivalent to maximizing the social welfare W. Initially, the total cost is zero. That is, $C^0 = 0$. We assume that the total cost of the system C^t is non-decreasing. It quantifies the minimum operational cost that the system needs to transport the passengers. Since the total cost is independent of the time and the orders of the request submissions, the following inequation holds:

$$\forall t \in \mathcal{T}, \quad C^{t+1} \geq C^t \ . \tag{5.3}$$

If the requests are delayed from t to $t+1$:

$$\forall t \in \mathcal{T}, \quad C^{t+1} = C^t \ . \tag{5.4}$$

Let $\delta_{r_i^t}$ denote the increase on the operational cost if passenger i is serviced when she submits the request r_i^t at time t, the following equation always holds for every $t \in \mathcal{T}$:

$$C^{t+1} - C^t = \sum_{\substack{t'=t+1 \\ i \in \mathcal{I}'^{t'}}} \delta_{r_i^{t'}} \ . \tag{5.5}$$

Equation 5.5 shows that the increase on the total operational cost from time t to time $t+1$ always equals to the summation of the marginal cost that is incurred by the requests being served at time $t+1$.

Let $\mathbb{V}$ denote the set of vehicles in operation and $\mathcal{V}^t \subseteq \mathbb{V}$ be the set of vehicles that have at least one seat available at time t. Initially, we divide the vehicles into N groups (or coalitions), where $N = |\mathbb{V}|$. At each time, a request $r \in \mathcal{R}^t$ is added into a group $\mathcal{X}_v^t \subseteq \mathbb{X}^t$

according to the mechanism policy, where $\mathbb{X}^t$ is the set of all groups at time t. At time t, all passengers in a group share the same fare rate (cost per unit demand). Let ρ denote the cost per unit demand, we have:

$$\rho_{\mathcal{X}_v^t} = \rho_{r_i^t} \ , \tag{5.6}$$

where $\quad (i \in \{i \mid r_i^t \in \mathcal{X}_v^t, t \in \mathcal{T}\})$.

5.3.2 IORS Mechanism

The IORS mechanism consists of three parts: fare estimation, pickup assignment and payment calculation. In the fare estimation process, the mechanism calculates a quote for each request. In the pickup assignment phase, the mechanism computes an optimal plan that minimizes the cost per unit demand. Finally, the mechanism provides payments immediately after successful completion of the rides.

Fare Estimation

As the passengers arrive and depart dynamically, the mechanism can only rely on the known information to compute the upper bound of the fare. The fare estimation process is described as follows (as shown in Algorithm 5.1): for each request r_i^t at time t, the mechanism first checks if a vehicle v (with vacancies) satisfies the passenger's latest departure time. If such a vehicle is found, then the mechanism compares the cost per unit demand before and after adding the request r_i^t into the coalition, respectively. If the cost per unit demand decreases, then the fare is calculated and stored in a set Q'. The mechanism selects the maximum fare in the set as the quote. Otherwise, the system rejects the request. Note that the mechanism picks the highest (instead of the lowest) cost per unit demand as the upper bound of the fare estimate. This is because the mechanism needs to adjust the assignments so that the system can produce the lowest cost per unit demand in general. Besides, it is a necessary condition for individual rationality and incentive compatibility. The algorithm takes $\mathcal{O}(n^3)$ time in the worst case.

Algorithm 5.1 The Fare Estimation Algorithm

Input: t - Time, $\mathcal{R}^t$ - A set of requests from passengers $\mathcal{I}^t$ at time t
Output: $\mathcal{Q}_{\mathcal{R}^t}$-The fare estimate for requests $\mathcal{R}^t$
Initialize: $\mathcal{Q}_{\mathcal{R}^t} \leftarrow \emptyset$

while $r_i^t \in \mathcal{R}^t$ **do**
  $\mathcal{Q}' \leftarrow \emptyset$
  while $v \in \mathcal{V}^t$ **do**
    Compute t'
    if $t' \leq \bar{t}_i$ **then**
      Compute $\rho_{\mathcal{X}_v^{t-1} \cup \{r_i^t\}}$
      if $\rho_{\mathcal{X}_v^{t-1} \cup \{r_i^t\}} < \rho_{\mathcal{X}_v^{t-1}}$ **then**
        $q' \leftarrow \ell_i^t \rho_{\mathcal{X}_v^t}$
        $\mathcal{Q}' \leftarrow \mathcal{Q}' \cup \{q'\}$
      end if
    end if
  end while
  if $\mathcal{Q}' \neq \emptyset$ **then**
    $q \leftarrow \operatorname*{argmax}_{q' \in \mathcal{Q}'} q'$
    $\mathcal{Q}_{\mathcal{R}^t} \leftarrow \mathcal{Q}_{\mathcal{R}^t} \cup \{(r_i^t, q)\}$
  end if
end while
return $\mathcal{Q}_{\mathcal{R}^t}$

Pickup Assignment

Let n_{v^t} denote the number of seats available in vehicle v at time t. Each vehicle can only service at most N_v passengers, where N is the seat capacity of the vehicle. That is, $0 \leq n_{v^t} \leq N_v$. When there are multiple requests that decrease the cost per unit demand of a coalition, the coalition selects the one that produces the lowest cost per unit demand. If there is a tie, the mechanism breaks it by choosing the one with the highest demand at random. The pickup assignment procedure is shown in Algorithm 5.2. The mechanism selects the n_t requests that produces the lowest cost per unit demand, where n_t is determined as following:

$n_t = \min\{n_{v^t}, n_{\mathcal{R}^t}\}$, where $n_{\mathcal{R}^t}$ is the number of requests submitted. The time complexity of Algorithm 5.2 is $O(n^2 \log n)$.

Algorithm 5.2 The Pickup Assignment Algorithm

Input: t - Time; $\mathcal{R}'^t$ - A set of requests from passengers $\mathcal{I}^t$ who accept the fare quotes at time t

Output: Π^t-The set of assignment

Initialize: $\Pi^t \leftarrow \emptyset$, $A \leftarrow \emptyset$

while $v \in \mathcal{V}^t$ **do**
    while $r \in \mathcal{R}'^t$ **do**
        Compute t'
        if $t' \leq \bar{t}_i$ **then**
            Compute $\rho_{\mathcal{X}_v^{t-1} \cup \{r\}}$
            if $\rho_{\mathcal{X}_v^{t-1} \cup \{r\}} < \rho_{\mathcal{X}_v^{t-1}}$ **then**
                $c' \leftarrow \rho_{\mathcal{X}_v^t}$
                $\mathcal{A} \leftarrow \mathcal{A} \cup \{(v, c')\}$
            end if
        end if
    end while
end while
$\mathcal{V} \leftarrow \mathcal{V}^t$
$\mathcal{I} \leftarrow \{i \mid r_i \in \mathcal{R}^t\}$
while $\mathcal{A} \neq \emptyset$ **and** $\mathcal{V} \neq \emptyset$ **and** $\mathcal{I} \neq \emptyset$ **do**
    $\mathcal{A} \leftarrow quicksort(\mathcal{A})$;
    $(v, c') \leftarrow \operatorname*{argmin}_{(v,c') \in \mathcal{A}} c'$
    $\Pi^t \leftarrow \Pi^t \cup \{(\hat{v}, \hat{r}) \mid \hat{v} = v, c_{\hat{r}} = c'\}$
    $\mathcal{A} \leftarrow \mathcal{A} \setminus \{(v^*, c^*) \mid v^* = v, (v^*, c^*) \in \mathcal{A}\}$
    if $n_v < 1$ **then**
        $\mathcal{V} \leftarrow \mathcal{V} \setminus \{v\}$
    end if
    $\mathcal{I} \leftarrow \mathcal{I} \setminus \{i \mid c_{r_i} = c', r_i \in \mathcal{R}'^t\}$
end while
      return Π^t.

Payment Calculation

When a passenger accepts a fare quote and is not assigned with a vehicle, her request will be added to time $t+1$ if the threshold $\bar{t}$ satisfies. In this process, the mechanism assumes that all the passengers accept the fare estimate. This is because if a passenger rejects the quote, the mechanism simply ignores the request and assumes that the passenger never submits it. We assume that the system can calculate the marginal cost and the optimal routes as quickly as necessary, although it might be time-consuming in real-world application due to limited computational power and the complexity of the traffic dynamics. However, it can be computed with meta heuristics [Hansen and Zhou, 2007]. At time t, the cost per unit demand for all requests assigned to vehicle v under the assignment of $\pi_v \in \Pi^t$ is determined as following:

$$\rho_{\mathcal{X}_v^t} = \frac{\sum_{t' \in [0,t]} \sum_{r \in \{r|(v,r) \in \Pi^{t'}\}} \delta_r}{\sum_{t' \in [0,t]} \sum_{r \in \{r|(v,r) \in \Pi^{t'}\}} \ell_r}. \tag{5.7}$$

Therefore, the final payment of passenger i at time t is calculated as following:

$$p_i^t = \ell_i \rho_{\mathcal{X}_v^t}, \tag{5.8}$$

where p_i^t can be calculated in $O(n^2 T \log n)$ time.

Ex-post Incentive Compatibility

We show that the IORS mechanism satisfies ex-post incentive compatibility.

Lemma 5.1. A passenger can not decrease her cost by delaying the submission of the request, provided that all other passengers report their demand truthfully and do not change their decisions on fare quotes. That is, for all $\tau_1, \tau_2, t \in \mathcal{T}$ and submissions $\mathbb{R}$ and $\mathbb{R}'$, where $0 \le \tau_1 < \tau_2 \le t$, $\mathbb{R} = \{\mathcal{R}^0, ..\mathcal{R}^{\tau_1}, \mathcal{R}^{\tau_2}, ..., \mathcal{R}^t\}$, and

$$\mathbb{R}'(t) = \begin{cases} \mathcal{R}^{\tau_1} \setminus \{r_i^{\tau_1}\} & : t = \tau_1 \\ \mathcal{R}^{\tau_2} \cup \{r_i^{\tau_1}\} & : t = \tau_2 \\ \mathbb{R}(t) & : \textit{otherwise} \ , \end{cases} \tag{5.9}$$

We have:

$$p^t_{\mathbb{R}(\tau_1)} \leq p^t_{\mathbb{R}'(\tau_2)} \tag{5.10}$$

Proof. Depending on whether r_i is serviced or not, we distinguish two cases:

- The request is not serviced: if the passenger delays her request from τ_1 to τ_2, then her latest departure time $\hat{\bar{\tau}}_1 = \bar{\tau}_1 - 1 < \bar{\tau}_1$. If the pickup time $t = \bar{\tau}_1$, then she will not be serviced at time τ_2, which is obviously less favorable than being serviced. Another situation is that the addition of the request at time τ_2 does not decrease the cost per unit demand of the coalitions at time τ_2, or the new cost per unit demand is less than the threshold determined by Algorithm 5.2.

- The request is serviced: If the passenger delays her request from τ_1 to τ_2. Assuming that,

$$p^t_{\mathbb{R}(\tau_1)} > p^t_{\mathbb{R}'(\tau_2)} \tag{5.11}$$

  We prove the theorem by contradiction. If $0 \leq \tau < \tau_1$, we have $\mathbb{R}'(\tau) = \mathbb{R}(\tau)$. By equation 5.1 and 5.4, the operational cost and the total unit demand are independent of the request submission time. That is, $C' = C$, $\sum \ell' = \sum \ell$. By equation 5.5, 5.7 and 5.8, we have $p^t_{\mathbb{R}(\tau)} = p^t_{\mathbb{R}'(\tau)}$. Thus, inequation 5.11 does not hold. This is also true if $\tau_2 < \tau \leq t$. If $\tau_1 < \tau \leq \tau_2$, since $\mathbb{R}'(\tau) = \mathcal{R}^{\tau_1} \setminus \{r_i^{\tau_1}\}$, we have the cost per unit demand $\rho_{\mathbb{R}(\tau_1)} \leq \rho_{\mathcal{R}^{\tau_1} \setminus \{r_i^{\tau_1}\}}$, and the total demand $\ell_{\mathbb{R}(\tau_1)} < \ell_{\mathcal{R}^{\tau_1} \setminus \{r_i^{\tau_1}\}}$. By multiplying the left and right sides of the two inequations, we get $\rho_{\mathbb{R}(\tau_1)} \ell_{\mathbb{R}(\tau_1)} \leq \rho_{\mathcal{R}^{\tau_1} \setminus \{r_i^{\tau_1}\}} \ell_{\mathcal{R}^{\tau_1} \setminus \{r_i^{\tau_1}\}}$. That is, $p^t_{\mathbb{R}(\tau)} \leq p^t_{\mathbb{R}'(\tau)}$. Hence, inequation 5.11 does not hold when $\tau_2 < \tau \leq t$. Therefore, the assumption is invalid and inequation 5.10 holds.

By incorporating the above cases, we prove the lemma. □

Lemma 5.2. The passenger can not decrease her cost by misreporting its latest departure time, provided that all other passengers report their demand truthfully and do not change their decisions on fare quotes. That is,

$$p'_r \leq p_r, \quad (\forall \hat{\bar{t}} \neq \bar{t}). \tag{5.12}$$

Proof sketch. If passenger claims an earlier latest departure time (i.e., $\hat{\bar{t}} < \bar{t}$), according to Algorithm 5.1 and 5.2, the search space of the vehicles may be reduced and the request might be rejected. If $\hat{\bar{t}} > \bar{t}$, the search space will be increased. However, a passenger will reject the assignment if the pickup time exceeds the $\bar{t}$ according to the assumption made in section 5.3.1. By equation 5.1, 5.4, 5.5, 5.7 and 5.8, the fare $\hat{\bar{p}}$ does not increase in either scenario. □

Theorem 5.1. *The IORS mechanism is ex-post incentive compatible provided that all other passengers report their demand truthfully and do not change their decisions on fare quotes.*

Proof sketch. A passenger i can not lie about her origin o_{ui} and destination d_i. She is unable to claim an earlier arrival t_i. According to Lemma 5.1, she will not benefit from delaying the request submission provided that all other passengers report their demand information truthfully and do not change their decisions on whether to accept the quotes or not. The passenger will not gain from misreporting the latest departure time according to lemma 5.2. □

Discussion

Note that the IORS mechanism does not require passengers to specify the deadlines for the latest delivery to their destinations. This is because passengers are likely to misreport the deadlines to rule out potential ridesharing assignments (e.g., by claiming an earlier deadline). However, mechanism designers may set constraints (e.g., the longest time, the maximum number of passengers, and the maximum rate) on a shared ride if necessary.

The IORS mechanism also satisfies other properties such as individual rationality and budget balance. For example, it is individual rational because passengers' final payments never exceed their quotes. The budget balance property is met for the reason that the total cost is split by the passengers who are provided with the ride services. Due to space limitations, we omit the poofs for these properties.

5.4 Benchmark

For evaluation purposes, we compute the optimal assignment as a benchmark to evaluate the efficiency of the IORS mechanism. The goal of the optimal assignment is to minimize the overall cost per unit demand (equivalent to maximizing W) under the constraints in the AMoD system. This is a minimum maximal matching problem, which is NP-hard [Hopcroft and Karp, 1973], and cannot be solved in polynomial time. We use a linear programming solver *LpSolve* solver [Berkelaar et al., 2004] for optimization in the experiment.

Auction-based mechanisms have been proven to be efficient in some of the existing ridesharing systems such as carpooling and shuttles [Kleiner et al., 2011, Cheng et al., 2014, Coltin and Veloso, 2013, Kamar and Horvitz, 2009]. Some of them even have very close performance compared with the optimal solution [Cheng et al., 2014]. In our work, we compare the performance of the IORS with the state-of-the-art, offline, auction-based mechanism (bottom-up) described in [Cheng et al., 2014]. The auction-based mechanism can not be solved in polynomial time.

5.5 Experimental Results

To evaluate the performance of the IORS mechanism, we developed an AMoD simulator to model the transportation system of a grid city with 101×101 blocks (a scenario similar to Figure 5.2).

5.5.1 Experimental Settings

In the experiment, we assumed that the number of AVs in the system is fixed. We set the number $N = 1000$. For each simulation, the system ran for 500 rounds unless specified otherwise. For each round, we generated a random number of requests $\mathcal{R}_t \in \mathbb{R}$. We initialized $\mathbb{R}$ with a set of $N = 500$ integers randomly drawn from a normal distribution with mean $\mu = 1000$ and standard deviation 100 (see Figure 5.3a). We assumed that each AV can transport up to four passengers at the same time.

We then generated the $\mathcal{R}_t$ requests respectively using the following method: the request time is the current round number; the waiting time is randomly drawn from the range 10 to 100; both the origins and destinations are randomly selected within a radius of 50 blocks in the grid. The operational cost per unit distance (block) is 1. The speeds for all vehicles are the same: 0.5 block per unit time (round). Initially, all the AVs depot at the center of the grid city. At time $t = 0$, the AVs become available for servicing passengers.

To be fair for evaluation and in the interest of saving time, we calculated the shortest paths between any two vertices in the city grid using the A^* algorithm [Hart et al., 1968] and saved it into a dictionary for further use in all the simulations

We ran all the simulations on a 2.9GHz quad-core machine with a 32GB RAM.

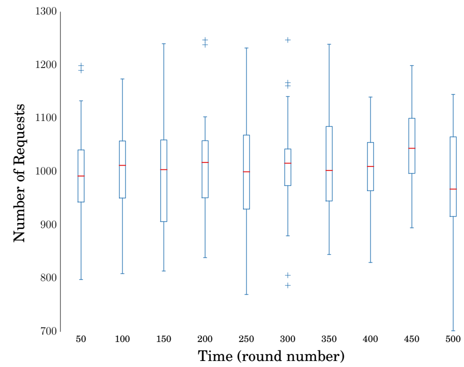
5.5.2 Results and Discussion

We performed simulations using the IORS mechanism. For comparison purposes, we computed the optimal assignment under the same experimental settings as a benchmark. we also conducted experiments on an AMoD system with the offline, auction-based mechanism described in section 5.4. To counter the effect of the fluctuations caused by the randomization techniques used, we ran all the experiments 20 times and calculated the mean and the standard deviation of the metrics evaluated.

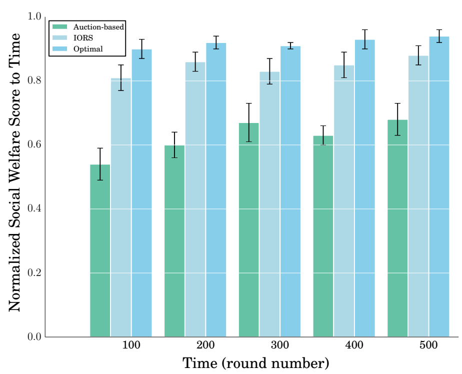
We computed the social welfare scores over time for the systems using one of the following mechanisms: IORS mechanism, auction-based mechanism and the optimal assignment solution. The result (see Figure 5.3b) clearly shows that the IORS mechanism performs significantly better (with a 95% confidence interval) than the auction-based mechanism, with an increase of 22.73%. Although it is a little inferior to the optimal solution, it performs fairly well (with a score equals to 93.62% of the optimal solution) with 79.35% less computational time on average (see Figure 5.3c) and no future knowledge of demand required.

The IORS mechanism adds a request only if this addition decreases the cost per unit demand of a group. However, it might suffer from local minima and produce suboptimal solutions due to a myopic view of the demand. The auction-based mechanism, on the other hand, processes the aggregated requests at once. It removes the requests with the lowest ranks. Although the mechanism might make better plans than the IORS mechanism at processing time because they have a better knowledge of demand distributions, it performs worse than the IORS at all the other time.

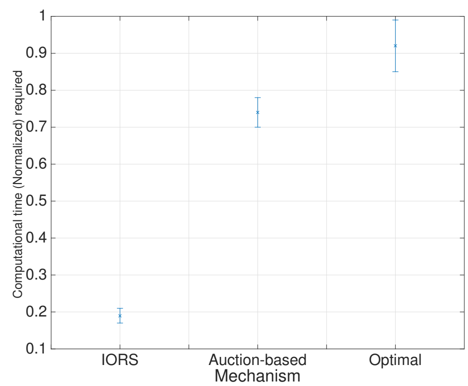
The revenue of IORS system is slightly lower than the optimal system, and much higher than the auction-based system (see Figure 5.4a). The effective demand of the system with the IORS mechanism fluctuates around 0.7, while the demand for optimal solution first increases



(a) Demand distribution at each time.



(b) Social welfare scores to time.

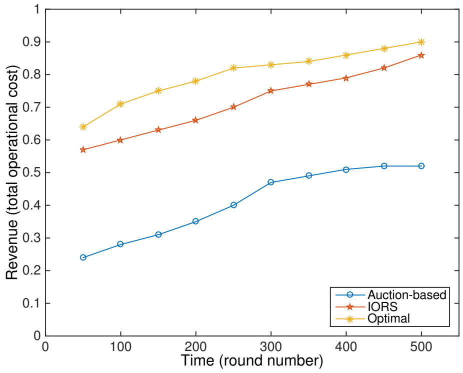


(c) Computational time.

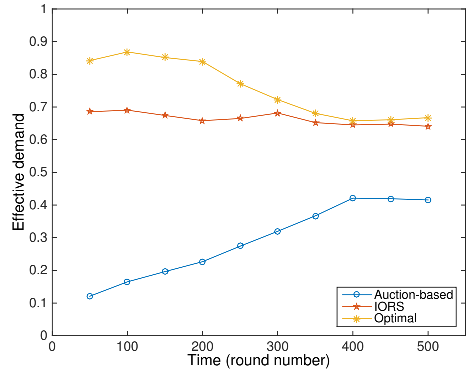
Figure 5.3: A comparison of demand distribution, the social welfare scores and computational time of a system with three different approaches: the IORS, an auction-based mechanism and the optimal solution.

and then drops down to a level very close to that of the IORS system. This is due to the increased demand from passengers. As shown in Figure 5.4b, the demand for auction-based system keeps increasing and then reaches a plateau. For each time measured, the scores of the auction-based system are the lowest.

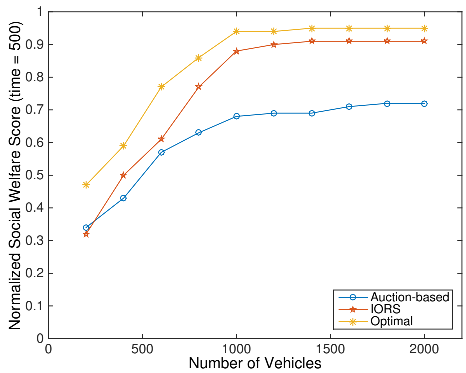
When the demand is high, obviously, it is effective to increase the supply (i.e., number of vehicles) at first. However, once the number of the AV fleet reaches some point, it will not help to improve the social welfare (i.e., W) (see Figure 5.4c).



(a) Revenue (total operational cost).



(b) Total effective unit demand.



(c) Varying the numbers of vehicles.

Figure 5.4: A comparison of the performance of a system with three approaches: the IORS, an auction-based mechanism, and the optimal solution.

In summary, the IORS mechanism outperforms the offline, auction-based mechanism overwhelmingly in promoting ridesharing in AMoD systems. Although it is still inferior to the optimal solution, it can achieve a very close performance with substantially less computational time needed and no future knowledge of demand required.

5.6 Summary

To promote ridesharing in AMoD systems, we introduce a posted-price, integrated online mechanism, namely IORS. We show that IORS is ex-post incentive compatible. Simulation results demonstrate its competitiveness compared with the optimal assignment solution and the offline, auction-based mechanism. Although IORS is tailored for AMoD systems, it is applicable to traditional demand responsive transport systems such as taxis and shuttles, provided that the dispatchers have full control over the vehicles. Besides, IORS can be applied to distributed scenarios by dividing a city into multiple zones where each zone has a control center individually processing the requests.

Future directions include coalition structure generation for optimal groups of shared riders, mechanism design to address ethics and privacy problems in ridesharing. Another direction is to develop more complex and realistic simulation platforms as benchmarks for future evaluation.

Chapter 6

Multi-Winner Contests for Strategic Network Diffusion

Strategic diffusion encourages participants to take active roles in promoting stakeholders' agendas by rewarding successful referrals. As social media continues to transform the way people communicate, strategic diffusion has become a powerful tool for stakeholders to influence people's decisions or behaviors for desired objectives. Existing reward mechanisms for strategic diffusion are usually either vulnerable to false-name attacks or not individually rational for participants that have made successful referrals. In this chapter, we introduce a novel *multi-winner contests* (MWC) mechanism for strategic diffusion in social networks. The MWC mechanism satisfies several desirable properties, including false-name-proofness, individual rationality, budget constraint, monotonicity, and subgraph constraint. Numerical experiments on four real-world social network datasets demonstrate that stakeholders can significantly boost participants' aggregated efforts with proper design of competitions. Our work sheds light on how to design manipulation-resistant mechanisms with appropriate contests. This work is based on our published work, [Shen et al., 2019]

6.1 Background

Strategic diffusion is the process of spreading information among social media users to promote desired private or social outcomes [Galeotti and Goyal, 2009]. As the impact of social media on people's daily lives continues to grow, strategic diffusion has become a prominent tool for stakeholders (e.g., individuals, companies, governments, and NGOs) to influence people's preferences, decisions or behaviors [Jackson and Yariv, 2011, Chaffey, 2016]. The vast popularity of strategic diffusion in social networks is primarily because it encourages participants to take active roles in promoting stakeholders' agendas in a word-of-mouth fashion (i.e., in the forms of referrals). This viral marketing strategy can reach a broader audience at a faster pace and are usually more economically efficient than traditional advertising such as newspapers, radios or televisions [Leskovec et al., 2007, Galeotti and Goyal, 2009].

To unleash the power of crowds, stakeholders usually reward both direct and indirect referrals to encourage potential participants to perform the tasks early and invite influential players to participate. For example, if Alice refers Bob and Bob then refers Cathy, both Alice and Bob are rewarded for Cathy's purchases. However, Bob is typically given more rewards than Alice for his direct referral. Such referral mechanisms are often called the *geometric reward mechanisms*, or the *incentive tree mechanisms* [Pickard et al., 2011]. Geometric reward mechanisms are usually effective and easy to implement [Tang et al., 2011]. As a result, they have witnessed a growing range of serious applications, including product promotion [Drucker and Fleischer, 2012], disaster rescue [Rutherford et al., 2013], global manhunt [Rahwan et al., 2013], participatory sensing [Gao et al., 2015] and crowdfunding [Naroditskiy et al., 2014b].

Despite the promising prospects of the geometric reward mechanisms, they are usually vulnerable to manipulations [Drucker and Fleischer, 2012]. Under many geometric reward mechanisms, indirect referrals are rewarded [Pickard et al., 2011]. A strategic player may create multiple fake accounts or identities on his behalf with one referring another to increase his rewards. Players' such malicious behaviors are often called *false-name attacks*. A false-name attack refers to a strategy that a profit-maximizing agent utilizes to gain benefits by creating multiple false identities (i.e., replicas). False identities can be analogous to "free riders" in the sense that they pay less than what they should have paid. This free rider problem arises when genuine identities are unfeasible or difficult to be recognized. Unfortunately, many social network services lack effective methods to fully eliminate false identities [Ferrara et al., 2016].

False-name attacks severely impede stakeholders from implementing desired individual or societal outcomes. On the one hand, false-name attacks are undesirable because they not only diminish stakeholders' revenues but also reduce other truthful players' payoffs [Drucker and Fleischer, 2012]. On the other hand, prior research indicates that false-name attacks are pervasive in social networks since players may create false accounts with no or minimal efforts if no interventions are given [Lorenz et al., 2011, Naroditskiy et al., 2014a, Ferrara et al., 2016].

Despite that much research has been devoted to tackling the false-name-attack problems in social networks [Conitzer et al., 2010, Emek et al., 2011, Todo et al., 2011, Brill et al., 2016], our work is most closely related to the mechanism design problem for multi-level marketing [Drucker and Fleischer, 2012]. In their work, Drucker and Fleischer introduce a class of mechanisms by capping the rewards a player can get from indirect referrals. They show that under their mechanisms false-name attacks are unprofitable. While illuminating, their methods indicate that some of the less influential players in the referral networks will receive no rewards for successful indirect referrals. For instance, players with ordinary or low capabilities may have little or no incentives to participate in the referral mechanisms. Ignoring these players is problematic for stakeholders because the population of many online platforms typically consist of a large portion of participants with low or common capabilities [Ipeirotis, 2010, Dow et al., 2012]. Besides, the implementation of their mechanisms is not based on graphs but rather on trees, it remains unknown whether their mechanisms can be applied to large-scale social networks consisting of thousands of participants with different abilities.

To address these issues, we introduce a novel mechanism called the *Multi-Winner Contests* (MWC) mechanism for strategic diffusion in social networks. The MWC mechanism distinguishes itself from existing methods in two aspects. First, it allocates virtual credits for each successful referrals by taking both diffusion contributions and verifiable task efforts into account. Second, it determines the diffusion rewards for each player according to the results of contests that compare the virtual credits earned by the qualified players. We show that the mechanism is false-name-proof, individually rational, budget-constrained, monotonic, subgraph-constrained, and computationally efficient. We conducted extensive experiments with four real-world social network datasets. Experimental results demonstrate that false-name-attacks are unprofitable under the MWC mechanism. Stakeholders can significantly boost the aggregated efforts of players when they select parameters of the MWC mechanism appropriately. Our work casts light on how to integrate competitions into the design of novel mechanisms to counter manipulations.

6.2 Strategic Diffusion in Social Networks

This section first describes notations used for modeling strategic diffusion in social networks. It then introduces the mechanism design problem for strategic diffusion. After defining the concept of false-name attacks, it presents the solution concepts for the reward mechanism design problem.

6.2.1 Preliminaries

Strategic diffusion processes or referral networks are usually modeled with *directed acyclic graphs* (DAGs). We consider a referral DAG $G = (V, E)$ (See Figure 6.1) where V denotes the set of players that may contribute (i.e., nodes) and E denotes the set of referral relationships (i.e., edges). For any nodes $v, u \in G$ ($v \neq u$), if there is a directed edge from v to u, then it means that u's decision to contribute is partially a result of v. In this case, we say u is a *direct successor* of v and v is a *direct predecessor* of u. For each node v, the number of direct predecessors v has is its *indegree* $deg^-(v)$. The number of direct successors it has is its *outdegree*, denoted by $deg^+(v)$. A source node (i.e., a seed node) has a indegree of 0, while a sink node has a outdegree of 0.

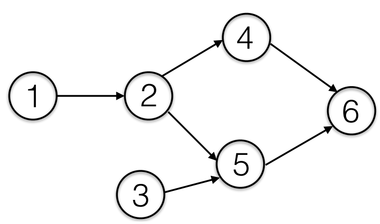


Figure 6.1: A typical directed acyclic graph.

For any nodes $v, u \in G$ ($v \neq u$), if a path leads from v to u, then u is said to be a *successor* of v and reachable from u, and v is said to be a *predecessor* of v. We write the distance between v and u by $dist(u, v)$. If there is no path from v to u, then the distance between the two nodes is infinity; that is $dist(v, u) = \infty$. The length from v to v is 0, i.e., $dist(v, v) = 0$. Let κ_v^+ be the set of all successors of v, we have $\kappa_v^+ = \{u \in V : 0 < dist(v, u) < \infty\}$. Similarly, the set of v's predecessors $\kappa_v^- = \{u \in V : 0 < dist(u, v) < \infty\}$. Let G_v be the subgraph rooted at v, then $G_v = (V', E')$, where $V' = \{v\} \cup \kappa_v^+$ and $E' \subseteq E$.

In a DAG $G = (V, E)$, if a player $v \in V$ makes a contribution to the designated tasks (e.g., making purchases, answering questions, reporting software bugs), we say player v exerts *task efforts* $t_v \in \mathbb{R}_{\geq 0}$. Player v may either directly or indirectly spread the information of the tasks to his successors to maximize his profits. If one of his successors u also contributes to the tasks, then we say v makes *diffusion contributions* $d_v \in \mathbb{R}_{\geq 0}$. In our model, we assume that the task efforts are verifiable and available for the stakeholder. For instance, a seller usually has the legitimate transactions information about the products she sells to each buyer.

Each edge $e = (v_i, v_j) \in E$ has a weight $\omega_{v_i v_j}$ (i.e., ω_e). The weight $\omega_{v_i v_j}$ represents the proportion of the credits assigned to v_i when v_j contributes to the designated tasks with efforts t_{v_j}. Here, $\omega_{v_i v_j} = 1/deg^-_{v_j}$ if $v_i \neq v_j$; otherwise, $\omega_{v_i v_j} = 1$. Let $P_{v_i v_j}$ be the set of all paths from v_i to v_j and p be a path in the set $P_{v_i v_j}$. Path p's weight $\omega(p)$ is the product of the weights of all edges along the path. That is,

$$\omega(p) = \prod_{(v_i, v_j) \in p} \omega_{v_i v_j} \ , \tag{6.1}$$

where the length of the path $|p| = dist(v_i, v_j)$.

6.2.2 Mechanism Design for Strategic Diffusion

We consider a principal (e.g., a seller, a task owner) employs strategic diffusion to maximize the aggregated efforts on the tasks she designates. Initially, the principal selects some participants (i.e., seeds) $S \subseteq V$ in a social network $G = (V, E)$ to perform the tasks that she specifies. The principal may select the seed nodes randomly if she has no prior knowledge of the social network. Alternatively, she may select the seed nodes using the influence maximization approach if the network structure and diffusion method are known a priori [Kempe et al., 2003] .

Some of the seed participants then perform the tasks and invite their neighbors in the social network to participate. If a participant v exerts task efforts t_v, he will receive a task reward $\pi_t(v)$ from the principal. Participant v may spread the information to his successors $u \in \kappa^+_v$ to maximize his profits. If a successor u exerts task efforts t_u, the referrer v will receive a diffusion reward $\pi_d(v,u)$. The total diffusion rewards for v: $\pi_d(v) = \sum_{u\in\kappa_v^+}\pi_d(v,u)$. Thus, the total rewards for v are determined by: $\pi(v) = \pi_t(v) + \pi_d(v)$.

The total contributions of v include both the task efforts and the diffusion contributions. In strategic diffusion, the task efforts are verifiable while the diffusion contributions are difficult to verify because strategic players can generate fake referrals with no or minimal efforts by creating multiple false identities with one being referred by another. We thus treat them separately. Let $c(v)$ be the total contributions for v, we have: $c(v) = (t_v, d_v)$. Let Θ_v be the space of v's total contributions $c(v)$, and $\Theta = (\Theta_v)_{v\in V}$.

In our model, we assume that the valuation of a player is linear in the rewards he receives and the cost he pays is linear in the task efforts he has contributed. That is, there are constants α_v, β_v, γ_v and ς_v with $\beta_v > 0$ and $\varsigma_v > 0$, such that player v's utility is $U(v) := \alpha_v + \beta_v \cdot \pi(v) - (\gamma_v + \varsigma_v \cdot t_v)$. Without loss of generality, let $\alpha_v = \gamma_v$ and $\delta_v = \frac{\varsigma_v}{\beta_v}$. We have player v's utility:

$$U(v) = \pi(v) - \delta_v \cdot t_v \ , \tag{6.2}$$

where $\delta_v > 0$ is a private coefficient that determines the player's marginal cost for exerting extra unit effort. The higher ability a player owns, the lower δ_v he has, and vice versa. That is, there is a negative correlation between a player's ability and his cost coefficient.

We note that a player may incur costs on spreading the information to his neighbors. We avoid explicitly including players' diffusion costs into the utility function for two concerns. First, diffusion efforts are not directly verifiable for the principal. The exact correlations between the costs on task efforts and the diffusion costs may vary from player to player, which are unknown to the principal. Second, a player v can integrate his diffusion costs into the task efforts by setting a higher cost coefficient δ_v. Similar techniques were also used in literature [Shen et al., 2018b].

In strategic diffusion, the principal is interested in a *reward mechanism* π that determines the reward for each player that has exerted efforts.

Definition 6.1. A reward mechanism π is a tuple of payments for each player $v \in V$, where $G = (V, E)$. That is, $\pi = (\pi(v))_{v \in V}$, where $\pi(v) : \Theta \to \mathbb{R}$.

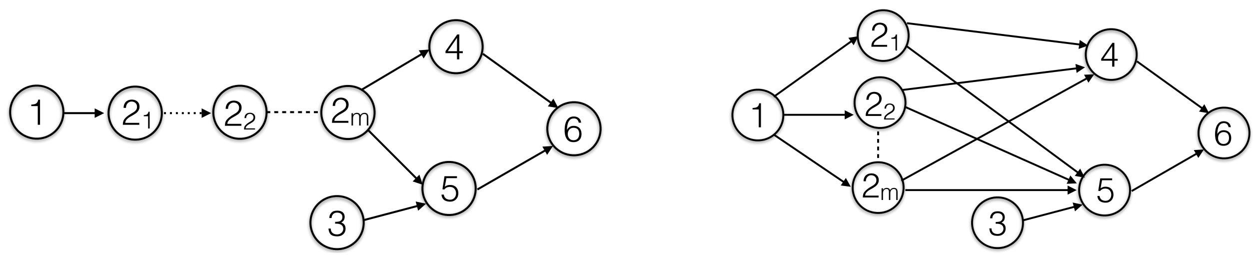
6.2.3 False-Name Attacks

In a graph $G = (V, E)$, we say a graph $G' = (V'', E'')$ and a set of replicas $R \subseteq V'$ are a *false-name attack* by v in G if when we collapse R into the single node with label v in G' we get the graph G.

Definition 6.2 (False-Name Attack)**.** Given a referral graph $G = (V, E)$, for any $v \in V$, let DS_v, DP_v be the sets of node v's direct successors and direct predecessors, respectively. $G' = (V'', E'')$ is obtained from G by a false-name attack at v if:

- $V'' = V \setminus \{v\} \cup \{r_1, ..., r_m\}$, $m > 1$. The set of nodes $R = \{r_1, ..., r_m\}$ is the set of replicas of v.
- The sum of the task efforts of v's replicas is equal to v's task efforts. That is, $t_v = \sum_{r \in R} t_r$, where $t_r > 0$.
- All replicas of v have at least one direct successor; that is $deg^+(r) \geq 1$ for all $r \in R$.
- For all $u \in DS_v$, $G_u = G'_u$, where G'_u is a subgraph of G' rooted at u.
- The direct predecessors of a replica of v are either replicas of v, or the direct predecessors of $v \in V$, or null. That is, $DP_r = R \setminus \{r\} \cup DP_v \cup \emptyset$, where $r \in R$.
- $\forall r \in R, \exists$ a replica r' of v such that $r \in \kappa^+_{r'}$.

In general, there are three types of false-name attacks: type 1 (See Figure 6.2a), type 2 (See Figure 6.2b), and hybrid. Type 1 false-name attacks occur in the form of a long referral chain that consists of the replicas, while in type 2 attacks the replicas operate in parallel. Hybrid attacks are combinations of type 1 and type 2 attacks.



(a) Type 1 (b) Type 2

Figure 6.2: False-name attacks at node 2: nodes $2_1, 2_2, ..., 2_m$ are replicas of node 2.

A false-name attack is profitable if the sum of the rewards received by the replicas $r \in R$ of $v \in V$ are higher than the reward v receives. That is, $\sum_{r \in R} \pi(r) > \pi(v)$.

6.2.4 Solution Concepts

Since false-name attacks are harmful to both the stakeholders and other truthful players, it is desirable that the reward mechanism satisfies the *false-name-proofness* property. A reward mechanism is false-name proof if false-name attacks are unprofitable for every player in a social network. That is, the total rewards that $v \in G$ receives are at least the same as the sum of the rewards received by the replicas of v in a graph obtained from a split of G at node v.

Definition 6.3 (False-Name-Proofness)**.** A reward mechanism π is false-name-proof if for all $v \in G$: $\pi(v) \geq \sum_{r \in R} \pi(r)$, where R is the set of replicas due to a false-name attack at node v.

According to Equations 6.2 and 6.5, if player v's costs on task efforts (i.e., $\delta_v \cdot t_v$) are equal to or higher than the rewards $\pi_t(v)$, then player v will have no incentives to participate in the strategic diffusion without positive diffusion rewards. To encourage potential players to participate, a reward mechanism needs to ensure that a player who has made successful referrals receives positive (expected) rewards. Otherwise, it would be not appealing to players with high marginal costs (i.e., high δ_v). A reward mechanism satisfies *individual rationality* if its (expected) utility is positive and it allocates positive (expected) diffusion rewards to each player that have made successful referrals.

Definition 6.4 (Individual Rationality)**.** A reward mechanism π is individually rational if for each player $v \in V$ with $d_v > 0$, we have: $U(v) > 0$, and $\pi_d(v, u) > 0$, where $u \in \kappa_v^+$, and $U(v)$ is determined by Equation 6.2.

In practice, a reward mechanism should be budget constrained. If not, the mechanism will be economically unfeasible for deployment. In our setting, the total rewards distributed to the players should not exceed a fixed portion ϑ of the total aggregated task efforts.

Definition 6.5 (Budget Constraint)**.** A reward mechanism π is budget constrained if: $\sum_{v \in G} \pi(v) \leq \vartheta \cdot \sum_{v \in G} t_v$, where ϑ is a positive constant.

Another important constraint is that a reward mechanism should limit indirect referrals to restrict the formation of long referral chains. This property is usually called the *monotonicity* constraint. A reward mechanism is monotonic if direct referrals receive higher diffusion rewards than or at least the same rewards as indirect referrals. The monotonicity property encourages participants to form short diffusion chains by offering more rewards. As a result, it limits the scope of indirect rewards, which is desirable in practice because many successful referral chains are usually short [Leskovec et al., 2007].

Definition 6.6 (Monotonicity)**.** A reward mechanism π is monotonic If v_2 is a successor of v_1, adding a direct successor v_i to v_2 increases v_1's diffusion rewards $\pi_d(v_1^{'})$ at least as much as the diffusion rewards $\pi_d(v_1^{''})$ by adding a direct successor v_j to a successor of v_2 , where $t_{v_i} = t_{v_j}$, i.e., $\pi_d(v_1^{'}) \geq \pi_d(v_1^{''})$.

In strategic diffusion, players should not have incentives to delay performing the tasks to wait for a referral with a more rewarding position in a social network G. To satisfy this constraint, a reward mechanism should determine the rewards for each player $v \in V$ based on the subgraph G_v rooted at v.

Definition 6.7 (Subgraph Constraint)**.** A reward mechanism π is subgraph-constrained if $\pi(v)$ only depends on the rooted subgraph G_v.

6.3 Multi-Winner Contests Mechanism

We present a novel reward mechanism called the *Multi-Winner Contests* (MWC) mechanism. It has two key ingredients. The mechanism first calculates the virtual credits for the diffusion contributions of each player with successful referrals. It then determines the diffusion rewards by holding a contest among players that are in his rooted subgraph. The MWC mechanism is computationally efficient and satisfies several desirable properties, including false-name-proofness, individual rationality, budget constraint, monotonicity, and subgraph constraint.

6.3.1 Virtual Credits for Diffusion Contributions

For each newly joined player v that has exerted task efforts t_v, the MWC mechanism pays $\pi_t(v) = \mu \cdot t_v$ ($\mu > 0$) for his task contributions and allocates virtual credits $\eta \cdot (t_v)^2$ for his diffusion contributions. If v has either directly or indirectly referred player $u \in \kappa_v^+$ to participate ($t_u > 0$), his virtual credits b_v are computed by:

$$b_v = \eta \cdot (t_v)^2 + t_v \cdot \sum_{u \in \kappa_v^+} \sum_{p \in P_{vu}} t_u \cdot \omega(p) \cdot \lambda^{|p|} \, , \tag{6.3}$$

where $0 < \lambda < 1$, $\eta \geq \lambda/2$, and P_{vu} is the set of paths from v to u.

The initial allowance of virtual credits $\eta \cdot (t_v)^2$ serves dual purposes. First, it ensures that each player has a positive amount of virtual credits to enter into the contests and receives a positive share of diffusion rewards. Second, it allows the mechanism to significantly reduce a player's virtual credits if he splits his task efforts. This dedicated design is an essential step that makes false-name attacks unprofitable. Note that $t_u \cdot \lambda^{|p|}$ is a typical *tree incentive mechanism* (i.e., geometric reward mechanism). The MWC mechanism extends it to a reward mechanism that applies to social networks by introducing the weights $w(p)$. To incentivize players to exert higher task efforts, the MWC mechanism amplifies the virtual credits for diffusion contributions by multiplying the player's task efforts t_v.

6.3.2 Diffusion Rewards

The MWC mechanism performs multi-winner contests to decide the diffusion rewards. In the contests, the winners are chosen simultaneously by a *contest success function* (CSF) that compares the virtual credits of each player. A CSF determines each player's probability of winning the contest in terms of all players' efforts [Skaperdas, 1996].

In general, there are two types of contest success functions: the *ratio* form and the *difference* form [Skaperdas, 1996]. In the ratio form, the winning probabilities depend on the ratio of efforts exerted by each player. In the difference form, they are determined by the difference in efforts that each player has exerted. The ratio form can be naturally applied to large contests that consist of many players, while extending the difference form to contests with more than two players is usually non-trivial and difficult [Jia et al., 2013]. Strategic diffusion in social networks typically involves a large number of players. It is thus desirable to use the ratio form CSFs for analytical convenience.

The ratio form CSFs typically predict contest outcomes from the ratio of the efforts that each player has devoted. For each node $v \in G$, the MWC mechanism holds a contest for each subgraph $G_v = (V', E')$ rooted at v. If player v's virtual credits $b_v \geq \eta \cdot (t_v)^2$, he will be allowed to enter into the contests. Otherwise, player v makes no contributions and receives 0 rewards. If v is allowed to enter into the contest, his probability of winning is determined by a contest success function: $prob(v) = \frac{(b_v)^\sigma}{\sum_{u \in V'} (b_u)^\sigma}$, where $0 < \sigma \leq 1$, and $b_v \geq \eta \cdot (t_v)^2$.

The parameter σ can be interpreted as the "noise" of a contest. It captures the marginal increase in the probability of winning caused by a higher effort and is crucial to the outcomes of the contest [Jia et al., 2013]. Contests with low σ can be regarded as poorly discriminating or "noisy" contests. That is, players with different efforts may have a similar level of chance to win. Contests with high σ can be regarded as highly discriminating in the sense that players with higher efforts have a greater chance to win. The contest success functions are imperfectly discriminating in the sense that with all of them, the prize at stake is awarded probabilistically to one of the players with higher virtual credits leading to a higher probability of winning the prize.

In the contest success function, each player's probability of success does not depend on his identity or the identities of his opponents, but just on the efforts (i.e., virtual credits) of the players. This property indicates that if in some cases two players have identical efforts then their probabilities of success must be equal and if all players were to exert identical efforts, then each one of them would have a probability of success equal to $1/n$. Besides, a player's probability of success is independent of agents who have not exerted efforts.

The total rewards for diffusion contributions are $\pi_d = \phi \cdot \sum_{u \in V'} t_u \frac{deg^-_{G_v}(u)}{deg^-(u)}$, where $\phi > 0$, and $deg^-_{G_v}(u)$ denotes the number of direct predecessors of u in $G_v = (V', E')$. Here, $deg^-_{G_v}(u) \leq deg^-(u)$. The diffusion rewards of v are thus determined as:

$$\pi_d(v) = \frac{(b_v)^\sigma}{\sum_{u \in V'} (b_u)^\sigma} \cdot \phi \cdot \sum_{u \in V'} t_u \cdot \frac{deg^-_{G_v}(u)}{deg^-(u)} \ . \tag{6.4}$$

The MWC mechanism utilizes the probability of winning to determine the proportion of the total diffusion rewards for each player. This is because empirical evidence shows that proportional-prize contests exert higher aggregated efforts than winner-take-all contests [Cason et al., 2010, Sheremeta, 2011]. Besides, proportional-prize contests also limit the degree of biases that discourage low-ability players without altering the performance of stronger players [Cason et al., 2010, 2018].

6.3.3 The MWC Mechanism

The MWC mechanism $\pi = (\pi(v))_{v \in G}$ uses a post-price mechanism $\pi_t(v) = \mu \cdot t_v$ to reward a player that exerts t_v task efforts. The total reward for player v includes both the task rewards $\pi_t(v)$ and the diffusion rewards $\pi_d(v)$. Thus, we have the total reward for player v:

$$\pi(v) = \begin{cases} \mu \cdot t_v + \pi_d(v) & \text{if } b_v \geq \eta \cdot (t_v)^2 \\ 0 & \text{otherwise ,} \end{cases} \tag{6.5}$$

where $\mu \in \mathbb{R}_{>0}$ is the reward parameter that characterizes to what extent the principal values agents' efforts, and b_v, $\pi_d(v)$ are defined by Equations 6.3 and 6.4, respectively.

The MWC mechanism π (See Algorithm 6.1 for an implementation) is computationally efficient. The social network $G = (V, E)$ can be implemented using adjacency lists, where each node maintains a list of all its adjacent edges. It takes $O(|V|^2 + |V| \cdot |E|)$ time to compute the virtual credits (See Lines 1–6). Computing the total rewards (See Lines 7–9) also takes $O(|V|^2 + |V| \cdot |E|)$ time. Therefore, the time complexity for the MWC mechanism is $O(|V|^2 + |V| \cdot |E|)$.

The MWC mechanism achieves false-name-proofness by introducing an initial allowance $\eta_v \cdot (t_v)^2$ of virtual credits for each player v that has exerted non-zero task efforts. If the task efforts reduce, the allowance will decrease quadratically. False-name-attacks are not

Algorithm 6.1 Multi-Winner Contests Mechanism

Input: G- a social network; S-seed nodes
Output: π-payment for each node $v \in G$
Initialize: $\pi(v) = 0$ for all $v \in G$
for each newly joined player $v \in G$ **and** $t_v > 0$ **do**
    $b_v \leftarrow \eta \cdot (t_v)^2$
    for each $v_i \in \kappa_v^-$ **and** $t_{v_i} > 0$ **do**
        $b_{v_i} \leftarrow b_{v_i} + t_{v_i} \cdot t_v \cdot \sum_{p \in P_{v_i v}} w(p) \cdot \lambda^{|p|}$
    end for
end for
for each v_j in G_v **do**
    Compute the rewards $\pi(v_j)$ by Equation 6.5
end for
return π

profitable because splitting task efforts into smaller pieces causes a larger degree of reduction than the degree of increase in virtual credits. This elegant design of the virtual credits enables the MWC mechanism to be manipulation-resistant to all the three types of false-name attacks. We postpone all the proofs to the Appendix.

Theorem 6.1 (False-Name-Proofness)**.** *The MWC mechanism π is false-name-proof.*

The MWC mechanism employs a post-price mechanism π_t to determine the rewards for players' task efforts. For the diffusion rewards, each player that has made successful referrals is guaranteed to receive a positive amount of rewards. Thus, it follows that the MWC mechanism satisfies the individual rationality property.

Theorem 6.2 (Individual Rationality)**.** *The MWC mechanism π is individually rational.*

The MWC mechanism is budget-constrained because it allocates a fixed portion of rewards (i.e., $\mu \cdot \sum_{v \in G} t_v$) for the task efforts and a fixed number of rewards (i.e., $\phi \cdot \sum_{v \in G} t_v$) for diffusion rewards.

Theorem 6.3 (Budget Constraint)**.** *The MWC mechanism π is budget-constrained.*

Under the MWC mechanism, players with direct referrals are given more virtual credits than if they have indirect referrals. This is achieved by introducing the discounting factor $\lambda < 1$. For the same level of task efforts, the virtual credits for diffusion rewards decrease as the distance of path between the referrer and the successor increases.

Theorem 6.4 (Monotonicity)**.** *The MWC mechanism π is monotonic.*

Since the diffusion rewards are determined by a contest among players in a subgraph rooted at player v, it follows that the MWC mechanism is subgraph constrained.

Theorem 6.5 (Subgraph Constraint)**.** *The MWC mechanism π satisfies the subgraph constraint.*

6.4 Experiments

Before describing the results, we introduce the experimental settings.

6.4.1 Experimental Settings

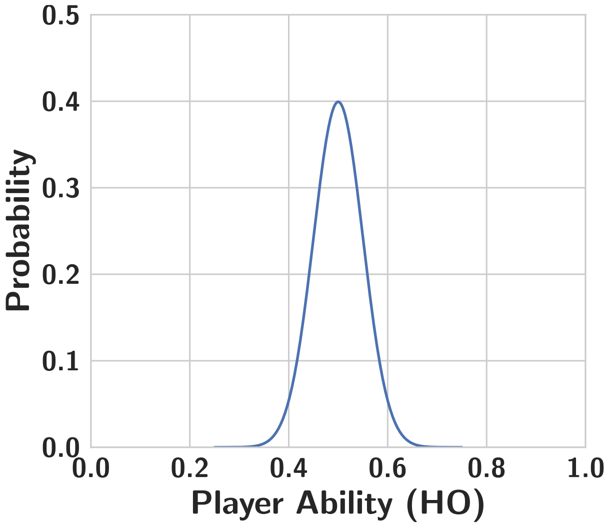
We used four public datasets: Twitter [Hodas and Lerman, 2014], Flickr [Cha et al., 2009], Flixster [Goyal et al., 2011], and Digg [Hogg and Lerman, 2012]. These datasets included anonymized timestamps that could be used to estimate the influence probabilities needed for simulating the diffusion process of social networks [Goyal et al., 2010]. For each dataset, we first estimated the influence diffusion probabilities for each node using the learning algorithms by Goyal et al. (2010) with the Bernoulli distribution under the static model. We then simulated the influence diffusion process with the general threshold model [Kempe et al., 2003] based on the estimated diffusion probabilities. After preprocessing, each dataset produced a largest weakly connected component. See Table 6.1 for dataset configuration.

Table 6.1: Dataset configuration (M.D.–maximum degree, A.D.–average degree)

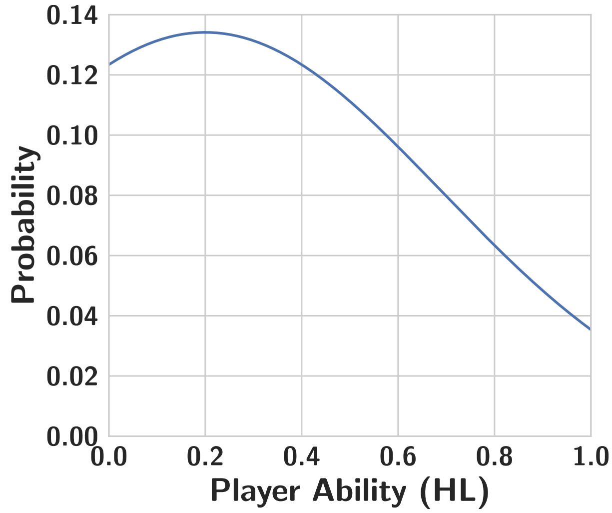
Dataset	#Nodes	#Edges	#Seeds	M.D.	A.D.
Twitter	323,185	2,148,717	1,715	8,822	52
Flickr	145,305	2,149,882	768	6,731	34
Flixster	95,969	484,865	502	3,109	27
Digg	17,817	128,587	107	1,375	20

We modeled players' abilities with the simulation method by Burnap et al. (2013). We considered four groups of players: *homogeneous (HO)*-players with similar levels of abilities; *heterogeneously low (HL)*-players with different level of abilities, and the average abilities are low; *heterogeneously high (HH)*-players with different level of abilities, and the average abilities are high; and *distinct (DI)*- a portion of players with low average abilities, the other portion with high average abilities. Each player's ability was generated according to a Gaussian distribution with means that follow the probability density function (See Figure 6.3). Let ρ_v be player v's ability, then his cost coefficient $\delta_v = 1 - \rho_v$.

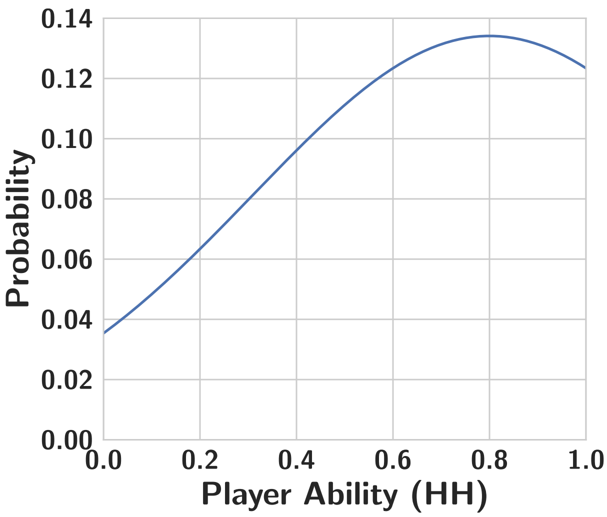
In our experiments, we set $\lambda = 0.5$ as it was standard in many geometric reward mechanisms. In practice, a stakeholder usually sets $\varphi \leq 1$ to make profits, but φ should be as close to 1 as possible to encourage players to participate. We let $\varphi = 1$. To encourage players to join, we set $\mu = 0.9$, and $\phi = \varphi - \mu = 0.1$. Note that $\eta \geq \lambda/2 = 0.25$, we set $\eta = 0.25$. For each group of players in each dataset, we varied the noise factors from 0 to 1 with an increment of 0.05. For each result (i.e., a data point) obtained, we ran the respective experiment 20 times. We ran all the experiments on the same 3.7GHz 6-core Linux machine with 32GB RAM.



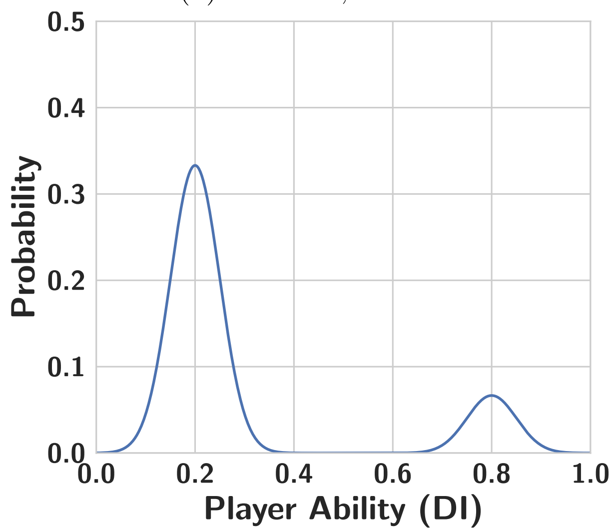
(a) mean=.5, std=.05



(b) mean=.2, std=.7



(c) mean=.8, std=.7

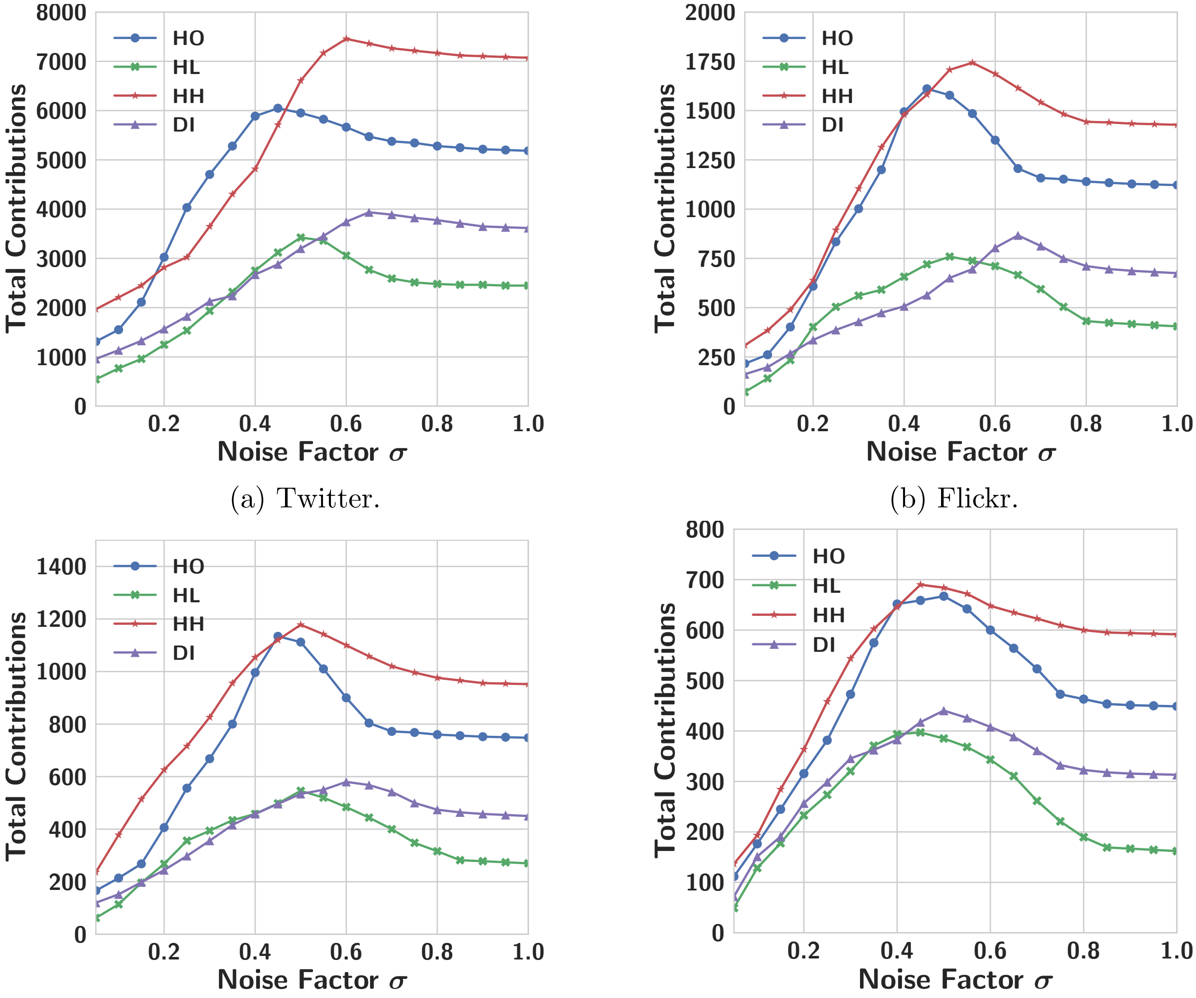


(d) means=.2, .8, std=.05

Figure 6.3: Probability density function of the means of players' abilities by four different groups: HO, HL, HH and DI.

6.4.2 Results

For each dataset, we compared the total aggregated task efforts by each group of players as the noise factor σ varied (See Figure 6.4). Figure 6.4a shows that the total contributions were quite low ($\leq 2,000$) when the noise factor was zero. The main reason is that players had no incentives to make extra diffusion contributions (See Figure 6.5a) since the diffusion rewards were determined by a random lottery with no dependence upon the efforts of players when $\sigma = 0$.

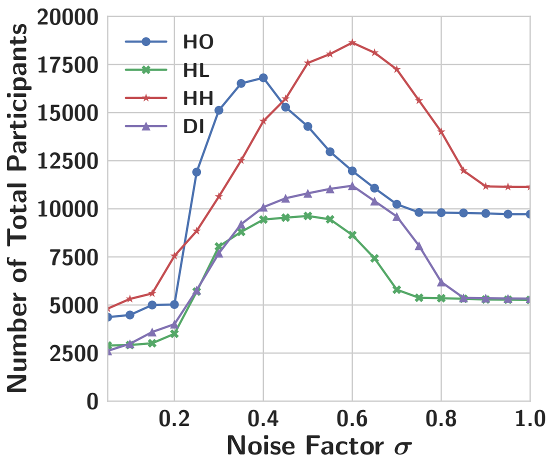


(a) Twitter. (b) Flickr.

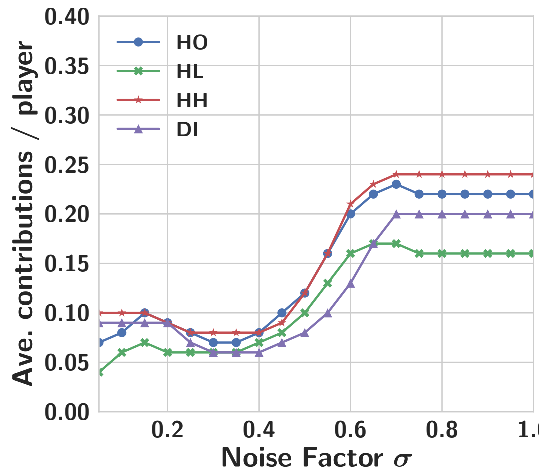
(c) Flixster. (d) Digg.

Figure 6.4: A comparison of total contributions with different bias factors.

As σ began to increase, the total contributions experienced a significant growth, which was largely due to the increase of players (See Figure 6.5a). The growth continued until it reached the peak where the total contributions began to shrink gradually (See Figure 6.4). The decline continued until it started to level off when the noise factor exceeded 0.8 (See Figure 6.4). An explanation for this trend is that when σ increased, the MWC mechanism values higher diffusion efforts more. When σ is low, the reward mechanism favors players



(a) Num. of Players.



(b) Ave. Contributions.

Figure 6.5: A comparison of number of participants and average contributions per player on Twitter.

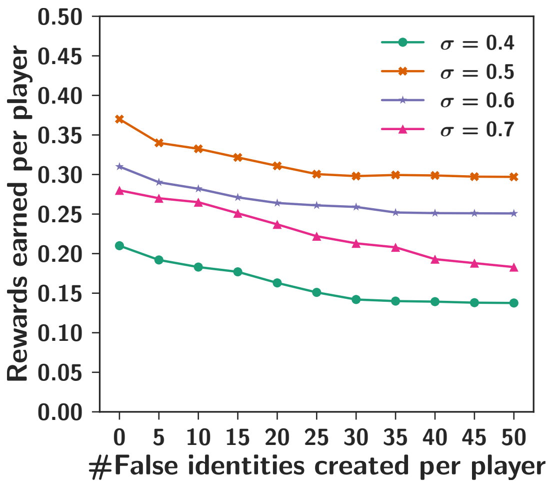
with average abilities. These players usually take up a large portion of the population. When σ reached a point, the mechanism no longer leaned toward these players. It became more appealing to high-ability players than the low-ability or ordinary-ability players (See Figure 6.5b). However, all groups except the HH group included a small portion of high-ability players. That also explains why the HH group experienced a slighter decline than the other groups. Similar observations were found in social networks.

Players in the HH and HO groups typically performed better than the other two groups. As the noise factor σ increased, the HO groups experienced the highest degree of growth in total task efforts. Two factors contributed to this phenomenon. First, HO groups mainly consisted of players with normal abilities (See Figure 6.3a). Second, when σ increased to a point, the contests favored players with common abilities. Figure 6.4 shows that the total contributions of the DI groups were the most invariant to the noise factors. This is because the DI groups lacked a population mass of players with ordinary abilities (See Figure 6.3d). The observations applied to all the three datasets (See Figure 6.4). Despite that the optimal noise factors differed in populations and network structures, there were "sweet spots" for stakeholders to maximize the total efforts. In our experiments, all the optimal noise factors fell within the range between 0.45 and 0.65 (See Figure 6.4), which suggested that medium noise factors were typically superior than the others.

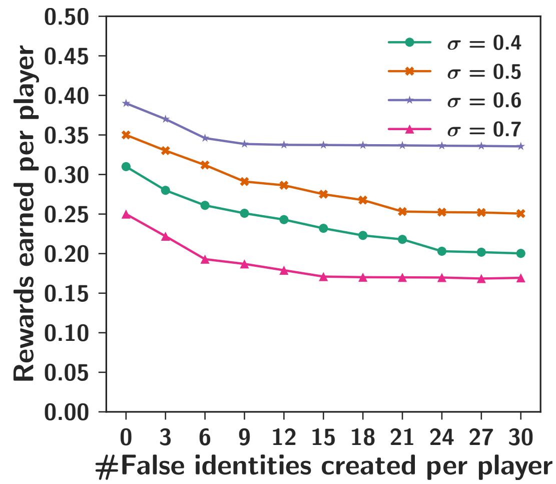
To study how false-name attacks affect players' rewards, we measured the total rewards earned by a player that had created different numbers of false identities. For each network, we considered a population consisting of the same percentage (25%) of players from each of the four groups: HO, HL, HH, and DI. We compared four noise factors: 0.4, 0.5, 0.6, and 0.7 with all the four networks.

Results show that the rewards earned by each player reduced significantly as the number of false identities increased from zero. The decline then slowed down until the rewards remained almost steady (See Figure 6.6). These trends were observed on all the groups of experiments that were given different noise factors and different network structures. An explanation is that when a player created a small number of false identities, his task efforts would be diluted to each false identity. This substantially reduced the player's virtual credits and his diffusion rewards. When the number of false identities continued to increase, there was little room to reduce. This consistent results further echoed our theoretical analysis that the MWC mechanism is false-name-proof.

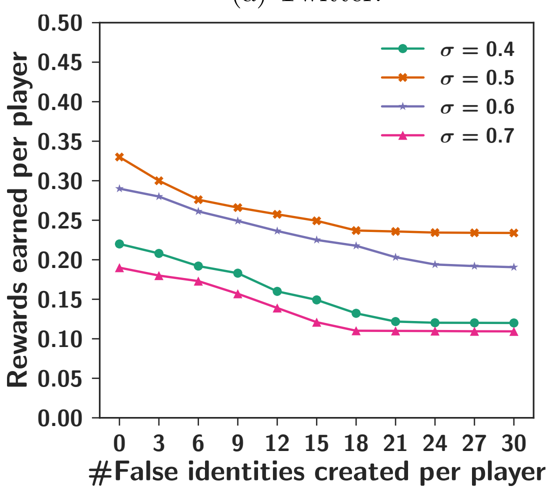
In summary, experimental results indicate that stakeholders can maximize the total task contributions by selecting appropriate (typically moderate) noise factors for the contests. It further demonstrates that players will not gain by creating false identities. The results also show that the MWC mechanism can scale to large social networks with hundreds of thousands of nodes.



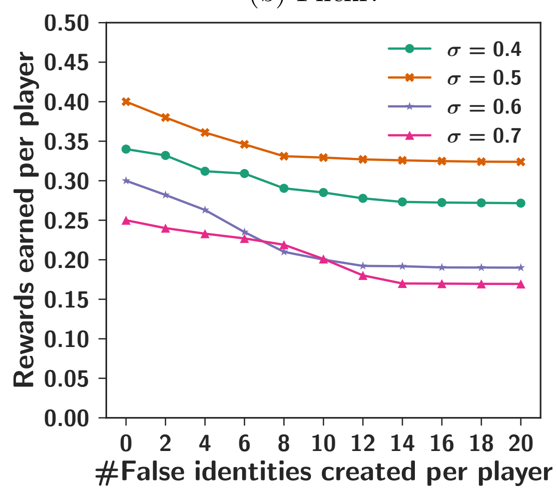
(a) Twitter.



(b) Flickr.



(c) Flixster.



(d) Digg.

Figure 6.6: Rewards (normalized) obtained by a player that creates different number of false identities for groups with different noise factors.

6.5 Summary

In this paper, we introduce a novel multi-winner contests mechanism for strategic diffusion in social networks. The mechanism is false-name-proof and individual rational for players with successful referrals. It is computationally efficient, budget-constrained, monotonic, and subgraph-constrained. Experiments on four real-world social networking datasets show that stakeholders can boost the performance of players by selecting proper noise factors. They further indicate that the MWC mechanism is both false-name-proof and scalable. Our work demonstrates the promising prospects of bringing contests to mechanism design.

Our work opens several exciting avenues for future research. In our model, agents do not discount the future and do not have uncertainties about the delivery of the rewards. In many real-world activities (e.g., crowdfunding, and investments), however, agents need to consider future risks when they make decisions. Another fertile area is to develop novel methods to automatically select optimal noise factors of MWC mechanisms for different concerns (e.g., profit maximization). We also find it very rewarding to integrate contests into the design of truthful mechanisms [Shen et al., 2016, Zhao et al., 2018].

Chapter 7

A Generalized Contest Mechanism for Strategic Network Diffusion

Strategic network diffusion, a marketing strategy that allows stakeholders to promote their desired outcomes by offering incentives to successful referrals, plays an important role in modern society. Unfortunately, existing incentive mechanisms for for strategic network diffusion are often vulnerable to manipulations such as false-name attacks and collusion. In this chapter, we introduce a new class of mechanisms called the *Generalized Contest Mechanism* (GCM). We show that the GCM are resistant to both false-name attacks and collusion. It also satisfies other desirable properties such as budget constraints, individual rationality, monotonicity, and subgraph constraints. Experiments on real-world social network datasets show that the GCM not only can beat the state-of-the-art mechanisms in terms of total efforts but is also more robust to false-name attacks and collusive manipulations. Our work brings new insights into how to counter manipulations with proper design of competitions.

7.1 Background

Strategic network diffusion refers to the marketing strategy in which a stakeholder (e.g., a seller, a task owner) incentivizes participants to spread the designated information across their social networks to achieve her desired outcomes [Galeotti and Goyal, 2009, Shen et al., 2019]. Such viral marketing strategy is usually more efficient and effective than traditional advertisement because referral marketing is usually more credible to potential customers [Berman, 2016]. It can also have access to new customers that traditional marketing methods may not reach, and provide better matching of potential users' needs to a product or service [Berman, 2016]. Moreover, previous studies show that the value and contribution of participants acquired through referrals are higher than those of non-referred customers [Schmitt et al., 2011]. Due to these benefits, strategic network diffusion has become a powerful tool for stakeholders to promote their agendas and has broad applications in many domains such as crowdsourcing [Pickard et al., 2011], disaster management [Gao et al., 2011, Besaleva and Weaver, 2013], health care [De Choudhury et al., 2013], e-commerce [Constantinides et al., 2008, Tiago and Veríssimo, 2014, Zhao et al., 2018], economic diversification [Alshamsi et al., 2018] and political campaigns [Enli, 2017].

To encourage early adoption and attract participation from influential players, stakeholders usually provide rewards (e.g., discounts, free products, event tickets or other monetary incentives) for both direct and indirect referral [Emek et al., 2011, Pickard et al., 2011]. For instance, if Alice has referred Bob and Bob has invited Dana, both Alice and Bob are rewarded for Dana's contributions (e.g., purchases, labor work) toward the stakeholders' goals. Mechanisms that provide incentives to the successful referral chains are often called *geometric mechanisms* [Emek et al., 2011, Drucker and Fleischer, 2012].

Geometric mechanisms have many important applications ranging from disaster response to product promotion [Besaleva and Weaver, 2013, Shen et al., 2019]. However, they are often vulnerable to manipulations such as false-name-attacks and collusion [Drucker and Fleischer, 2012]. In strategic network diffusion, a false-name attack refers to an agent's strategy to create multiple false identities in order to increase his diffusion rewards, while collusion is a manipulative strategy that a group of agents communicate to jointly limit the competition (e.g., reducing the task efforts). False-name attacks and collusion are pervasive in social networks since players may create multiple accounts with no or minimal cost, and may communicate with their social ties to reduce the competition if they can make profits [Drucker and Fleischer, 2012, Naroditskiy et al., 2014a, Ferrara et al., 2016]. Nevertheless, Both manipulations are undesirable because they not only diminish stakeholders' ability to implement their goals but also reduce other honest players' payoffs [Drucker and Fleischer, 2012, Shen et al., 2019]. Besides, agents' manipulative behaviors may crash the markets [Tang et al., 2011].

Our contributions: To counter false-name attacks and collusion in strategic network diffusion, we introduce a new mechanism called the *Generalized Contest Mechanism* (GCM). In doing so, we first characterize a necessary and sufficient condition for a reward mechanism to be false-name-proof: the reward function should be superadditive in terms of the diffusion contributions. We then identify a sufficient condition for a mechanism to be collusion-proof: the total diffusion rewards for the referrers of a player should be increasing and bounded above in terms of the total task efforts of the referrers. Finally, we integrate both conditions into a single mechanism. We demonstrate that the resulting mechanism is both false-name-proof and collusion-proof. It also satisfies other desirable properties such as budget constraints, individual rationality, monotonicity, and subgraph constraints. Numerical results demonstrate that the GCM can outperform the state–of-the-art multi-winner contests (MWC) mechanism [Shen et al., 2019] in terms of total task efforts with a variety of supperadditive functions and increasing and bounded functions. The GCM is also more robust to false-name attacks and collusive manipulations than the MWC mechanism. Our work provides a systematic approach to manipulation-resistant mechanism design.

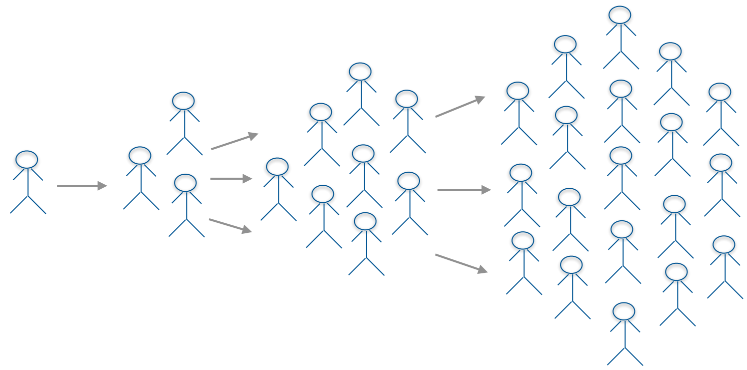


Figure 7.1: A simple illustration of strategic network diffusion.

7.2 Strategic Network Diffusion

Before introducing the mechanism design problem, we first describe key notations.

7.2.1 Notations

We model referral processes with a *directed acyclic graph* (DAG) $G = (V, E)$. In a DAG G, nodes V represent the players that may voluntarily participate in the strategic diffusion to increase their utilities. To maximize their profits, the players need to both complete the principal's designated tasks (e.g., buying a product, collecting data, crowdfunding, manhunt, and software development) and invite their neighbors to participate. Edges E correspond to the referral relationships.

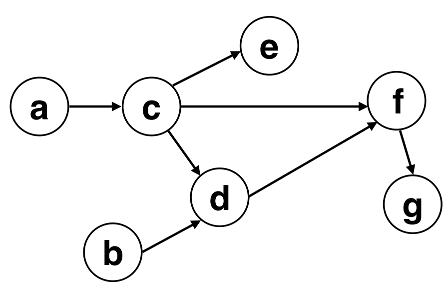


Figure 7.2: A typical directed acyclic graph for modeling strategic network diffusion.

For any two nodes $u, v \in V (v \neq u)$, if a path from u to v exists, then u is a *predecessor* of v, and v is a *successor* of u. In this case, player v's decision to contribute to the principal's tasks is (partially) due to u's referral. We write the distance between u and v by $dist(u, v) \in \mathbb{Z}_{>0}$. The distance between v and itself is 0 (i.e., $dist(v, v) = 0$). If there is no path from u to v, then $dist(u, v) = +\infty$. If $dist(u, v) = 1$, we say u is a *direct predecessor* of v, and v is a *direct successor* of u. For each node $v \in V$, the number of its direct predecessors is its *indegreee* $deg^-(v)$, and the number of its direct successors is its *outdegree* $deg^+(v)$. The set of v's all predecessors $I_v^- = \{u \in V : 0 < dist(u, v) < +\infty\}$. Similarly, the set of v's all successors $I_v^+ = \{u \in V : 0 < dist(v, u) < +\infty\}$. The subgraph rooted at v includes node v and all its successors. That is, $G_v = (V_v, E_v)$, where $V_v = \{v\} \cup I_v^+$, and $E_v \subseteq E$.

In a referral DAG $G = (V, E)$, if a player v makes irreversible contributions to the principal's designated tasks (e.g., purchasing products without returns, collecting data, manhunt, and software development), then player v exerts task efforts $t_v \in \mathbb{R}_{\geq 0}$. Player v can choose to spread the information of the tasks to his successors and invite them to participate to maximize his profits. If one of his successors $u \in I_v^+$ also exert *task efforts* $t_u \in \mathbb{R}_{\geq 0}$, then we say player v makes *diffusion contributions* $d_v \in \mathbb{R}_{\geq 0}$. In strategic network diffusion, a player's decision to exert positive task efforts may be due to either a single referral chain or multiple referrals. This setting is more general than the tree setting where a player's decision to make contributions can be only traced from a single referral chain. For example, it is likely that a customer will use a mobile application only if more than one of his friends have actively used it and have sent invitations to him. Similar to the tree setting, a player in a referral network is also permitted to invite multiple neighbors to increase his diffusion contributions.

In practice, players' task efforts are usually measurable and verifiable to the principal. For instance, the principal can quantify a player's task efforts according to the pieces of the tasks or the amount of time spent on the tasks. We thus assume that players' task efforts are observable in our model. However, the diffusion contributions are often difficult or even unfeasible to verify because strategic players can generate fake referrals with no or minimal efforts by creating multiple false accounts or identities to form extra referral chains.

7.2.2 The Mechanism Design Problem

We consider a principal aims to maximize the aggregated task efforts via strategic network diffusion. At the beginning, the principal selects a number of participants $S \subseteq V$ in a social network $G = (V, E)$ as the seed nodes. The optimal selection of seed nodes in a social network falls into the broad categories of influence maximization problems [Kempe et al., 2003]. The influence maximization problem is proven to be NP-hard even the diffusion model is known as a priori, but one can address it with approximation algorithms [Chen et al., 2009]. Alternatively, the principal can choose the seed nodes arbitrarily if she has no prior knowledge of the social network.

The seed participants may voluntarily perform the tasks and invite their neighbors in the social network to participate. If a player $v \in G = (V, E)$ exerts task efforts t_v, he will receives a task reward $\pi_t(v)$ from the principal. If v also spread the information to his successors and one of his successors $u \in I_v^+$ also exerts task efforts, then the referrer v will also receive a diffusion reward $\pi_d(v, u)$. Let $\pi_d(v)$ denote the total diffusion rewards that v receives, we have $\pi_d(v) = \sum_{u \in I_v^+} \pi_d(v, u)$. Player v's total rewards $\pi(v) \in \mathbb{R}_{\geq 0}$ include both the task rewards and the diffusion rewards, i.e., $\pi(v) = \pi_t(v) + \pi_d(v)$.

In our model, we assume that the valuation of a player is linear in the rewards he receives and the cost he pays is linear in the efforts he has contributed. Let $c(v) \in \mathbb{R}_{\geq 0}$ denote the total cost for player v to exert task efforts t_v and diffusion contributions d_v, player v's total cost $c_v = t_v + d_v$. Let $\rho_v \in \mathbb{R}_{\geq 0}$ denote the parameter that measures the relative cost for player v to make a unit diffusion contribution compared with the cost of the unit task effort, we have $d_v = \rho_v \cdot t_v$. Hence, $c(v) = (1 + \rho_v) \cdot t_v$. Let Θ_v be player v's type space for the total cost c_v, then the type space for all the players in a social network $G = (V, E)$ is $\Theta = (\Theta_v)_{v \in V}$. In practice, the cost for information diffusion is relatively insignificant compared with the task efforts. That is, $0 \leq \rho_v << 1$. For example, a player share the information via mass messages to all the target neighbors.

Player v's utility $U(v) = \pi(v) - \zeta_v \cdot c_v$, where $\zeta_v \in \mathbb{R}_{\geq 0}$ is a private coefficient that quantifies player v's marginal cost for exerting extra unit effort. Let $\delta_v = \zeta_v \cdot (1 + \rho_v)$, we have player v's utility

$$U(v) = \pi(v) - \delta_v \cdot t_v \ , \tag{7.1}$$

where $\delta_v \in \mathbb{R}_{\geq 0}$ is a private parameter that quantifies player v's marginal cost for exerting extra unit effort. In practice, performing manipulations also incur costs. However, such cost can be incorporated into the private cost coefficient that is privately known to the player himself [Shen et al., 2019].

In strategic network diffusion, the principal is interested in an incentive mechanism π that determines the rewards for each player that has exerted efforts (See Def. 7.1). In practice, an incentive mechanism must satisfy several constraints (e.g., budget feasibility, and voluntary participation) to be implementable. In this paper, we are particularly interested in a mechanism that is resistant to manipulations such as false-name attacks and collusion.

Definition 7.1 (Incentive Mechanism)**.** An incentive mechanism π for strategic network diffusion in a social network $G = (V, E)$ is a payment rule $\pi = (\pi(v))_{v \in V}$, where $\pi(v) : \Theta \to \mathbb{R}$ is the payment function for agent v.

7.3 Design Constraints

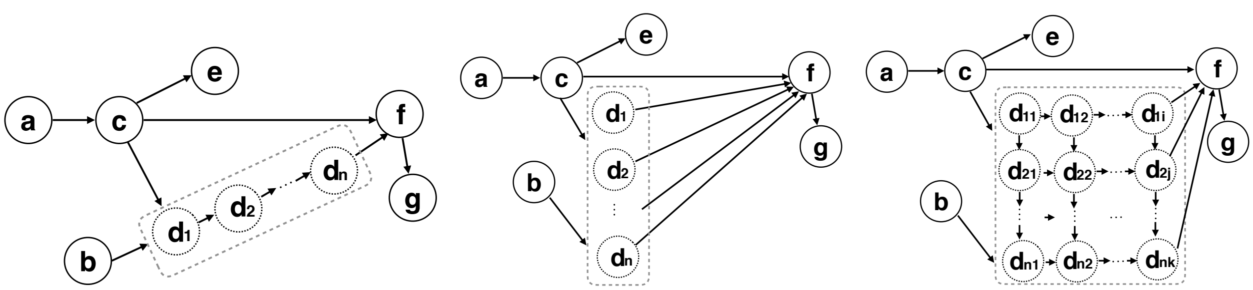
In this section, we formally introduce the two primary constraints for manipulation-resistant mechanism design: false-name proofness and collusion-proofness. We also discuss other constraints such as individual rationality, budget constraint, monotonicity and subgraph constraint.

7.3.1 False-Name Proofness

In strategic network diffusion, a strategic agent may create false identities (i.e., replicas) to form fake referrals to maximize his profits. In a referral graph $G = (V, E)$, a false-name attack for player $v \in G$ is a split at node v that results in a new graph $G' = (V', E')$ and a set of replicas $R \subseteq V'$. For the resulting graph G', if we collapse R into the single node v in G' we get the graph G.

Definition 7.2 (False-Name Attack)**.** In a DAG $G = (V, E)$, we say a graph $G' = (V', E')$ and a set of replicas $R \subseteq V'$ are a *false-name attack* for player v in the graph G if $v \in V'$ and when we collapse R into the single node v in G' we get the graph G.

In strategic network diffusion, there are generally three types of false-name attacks: *serial split*, *parallel split* and *hybrid split*. Serial split forms a long referral chain with one replica referring to another (See Fig. 7.3a). Parallel split duplicates the replicas in parallel to increase the count of referral relationships (See Fig. 7.3b). Hybrid split consists of both serial attacks and parallel attacks (See Fig. 7.3c). False-name attacks are detrimental to both the principal and other honest players. They can even crash the markets if most of the participants participate in the false-name attacks. Therefore, it is desirable that the incentive mechanism satisfies the *false-name-proofness* property.



(a) Serial split (b) Parallel split (c) Hybrid split

Figure 7.3: Three types of false-name attacks at node d.

An incentive mechanism is false-name-proof if under the mechanism false-name attacks are unprofitable for each player in the referral graph. Specifically, the total rewards that $v \in G = (V, E)$ receives are greater than or the same as the sum of the rewards obtained by the replicas of v due to a split of G at node v (See Def. 7.3).

Definition 7.3 (False-Name-Proofness)**.** An incentive mechanism π is false-name-proof for all $v \in G : \pi(v) \geq \sum_{r \in R} \pi(r)$, where R is the set of replicas due to a false-name attack at node v.

7.3.2 Collusion-Proofness

Different from false-name attacks, a collusion in a referral graph $G = (V, E)$ is a coalition strategy that makes at least one of the weakly connected players better off by changing the internal network structure (i.e., adding or removing the referral relationships) or reducing the task efforts of the coalition (See Def. 7.4).

Definition 7.4 (Collusion)**.** A collusion is a coalition strategy of weakly connected players that makes at least one player in the coalition better off by changing the internal network structure or reducing the task efforts of the coalition. Formally, in a DAG $G = (V, E)$, we say a graph $G' = (V', E')$ and a set of players $P \subseteq V'$ are a collusive coalition if under the incentive mechanism π, we have

- for each pair $u, v \in P$, there exists either a directed path from u to v in G, or a directed path from v to u in G.
- for each $v \in V'$, we have $v \in V$.
- for each $u, v \in V \setminus P$, if $uv \in E$, then $uv \in E'$;
- for each $v \in V \setminus P$, the task efforts t_v remain unchanged.

- for each $v \in P$, the task efforts t_v do not increase.
- There exist at least one $v \in P$ such that we have $U(v) < U'(v)$, where $U(v)$ is player v's utility when there is no collusion, and $U'(v)$ is player v's utility when v is in a collusive coalition. Other members' utilities do not change.

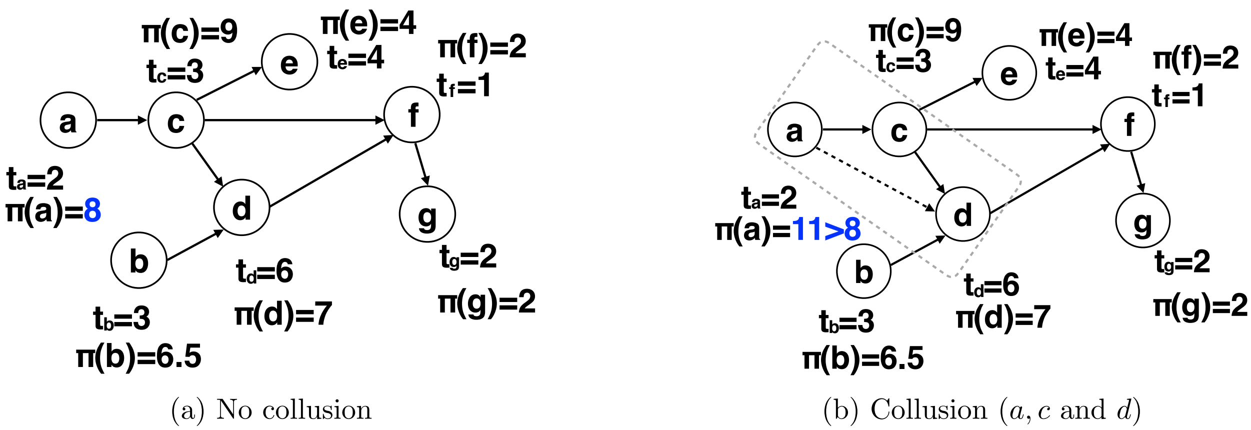


Figure 7.4: An example of strategic network diffusion with no collusion and with collusion under a typical geometric mechanism.

A group of players may reduce the task efforts, or change the internal network structure, or use combination of these two strategies to form a collusive coalition. Note that there may be coalitions that can increase the total efforts, which are usually beneficial to the principal. Such behaviors are typically not categorized as collusion.

Collusion is harmful to both the principal and other honest players. The collusive behavior depletes the principal's budgets with less task efforts. When the budget is fixed, under many mechanisms, the budget allocated to other honest players become less. Therefore, the incentive mechanism should satisfy the *collusion-proofness* property. An incentive mechanism is collusion-proof if under the mechanism no group of weakly connected players can collude to make at least one member of the group better off by changing the internal network structure or reducing the task efforts without making the remaining members worse off (See Def. 7.5).

Definition 7.5 (Collusion-Proofness)**.** An incentive mechanism π is collusion-proof if no group of weakly connected players $P \subseteq V$ can increase the utilities for at least one player in the coalition without reducing the utilities for any other member in the group through changing the internal network structure or reducing the task efforts of the coalition.

7.3.3 Other Constraints

To be implementable in practice, an incentive mechanism should also satisfy other desirable constraints, such as *voluntary participation*, *budget constraint*, *monotonicity*, and *subgraph constraint*.

A player should not be forced to participate in a mechanism. That is, a mechanism needs to incentivize or at least not penalize participation. An incentive mechanism satisfies the *voluntary participation* property if each player $v \in G = (V, E)$ receives positive (expected) diffusion rewards for each successful referral.

Definition 7.6 (Voluntary Participation)**.** An incentive mechanism π satisfies voluntary participation if for each player $v \in G = (V, E)$ with $d_v \geq 0$, we have $U(v) \geq 0$, and $\pi_d(v, u) > 0$, where $u \in I_v^+$, and $U(v)$ is determined by Equation 7.1.

To be economically feasible, an incentive mechanism should be budget constrained. In strategic network diffusion, the total rewards allocated to the players should not exceed a fixed portion ϑ of the total aggregated task efforts.

Definition 7.7 (Budget Constraint)**.** A reward mechanism is budget-constrained if we have $\sum_{v \in G} \pi(v) \leq \vartheta \cdot \sum_{v \in G} t_v$, where $\vartheta \in \mathbb{R}_{>0}$ is a predefined constant.

A mechanism for strategic network diffusion should limit indirect referrals to prevent agents from forming unnecessarily long referral chains. That is, the mechanism should satisfy the *monotonicity* constraint to limit the scope of indirect referrals and facilitate the task completion.

Definition 7.8 (Monotonicity)**.** An incentive mechanism π is monotonic if v_2 is a successor of v_1, adding a direct successor v_i to v_2 increases v_1's diffusion rewards $\pi_d^{'}(v_1)$ at least as much as the diffusion rewards $\pi_d^{''}(v_1)$ by adding a direct successor v_j to a successor of v_2, where $t_{v_i} = t_{v_j}$.

In strategic network diffusion, players should have no incentives to delay performing the tasks to wait for a referral with a more rewarding position. That is, an incentive mechanism should determine the diffusion rewards for each player $v \in V$ based on the reverse subgraph G_v rooted at v.

Definition 7.9 (Subgraph Constraint)**.** An incentive mechanism π is subgraph-constrained if $\pi(v)$ only depends on the rooted subgraph G_v.

It is challenging for stakeholders to design a mechanism that satisfies all the constraints, especially false-name-proofness and collusion-proofness. A main reason is that in practice agents' identities are often costly to verify and the collusive communications are usually difficult or unfeasible to capture [Jin et al., 2011, Yang et al., 2014, Feldman and Chuang, 2005]. To tackle this challenge, a careful treatment is needed.

7.4 Generalized Contest Mechanism

In this section, we take a decomposition approach to manipulation-resistant mechanism design for strategic network diffusion. To do so, we first characterize the conditions for an incentive mechanism to satisfy both false-name-proofness and collusion-proofness. We then present a class of Generalized Contest Mechanism (GCM) that satisfy all the design constraints.

7.4.1 Superadditivity

We decompose the incentive mechanism into two payment rules: one for the task efforts and the other for the diffusion contributions. For task efforts, a posted price mechanism is used . Posted price mechanisms are both false-name-proof and collusion-proof because they determine the rewards for each agent based solely on the verifiable task efforts.

We now focus on the diffusion incentive mechanism $\pi_d(v)$. In order to satisfy the false-name-proofness requirement, the incentive mechanism should be supperadditive in the diffusion contributions d_v. That is, for any two diffusion contributions $d(v_1), d(v_2) \in \mathbb{R}_{\geq 0}$, we have $\pi_d(d(v_1) + d(v_2)) \geq \pi_d(d(v_1)) + \pi_d(d(v_2))$. The intuition is inspired from the superadditivity in functional analysis. The supperadditivity requirement of the mechanism brings a monotonicity in terms of the diffusion rewards that create extra incentives for the agent not to split.

Definition 7.10 (Superadditivity)**.** Let $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$, a function f is *supperadditive* if $f(x + y) \geq f(x) + f(y)$ for all x, $y \in \mathbb{R}_{\geq 0}$.

Theorem 7.1. *An incentive mechanism is false-name-proof if and only if it is superadditive in terms of the diffusion contributions.*

Theorem 7.1 gives us a sufficient and necessary condition for an incentive mechanism to be false-name-proof when players' valuation is linear in the rewards and the costs he pays are linear in the task efforts. To design a false-name-proof mechanism, the principal needs to make the diffusion mechanism be superadditive in the diffusion efforts.

7.4.2 Increasing and Bounded Functions

In strategic network diffusion, a group of strategic players may collude to change the internal network structure or reduce the task efforts to benefit at least one member of the group without hurting the remaining members. To counter agents' such behavior, a mechanism should bound the total diffusion rewards for a referral. However, referrers will have no incentives to provide extra efforts if the number of the total diffusion rewards is fixed. To encourage the referrers to exert efforts, the mechanism needs to be monotonically increasing in the sum of total task efforts. Combining the two required conditions, the mechanism should be a monotonically increasing and bounded function in the sum of the referrers' task efforts.

Inspired by the monotone convergence theorem [Bibby, 1974] in real analysis, we demonstrate that if an incentive mechanism is monotonically increasing and bounded above in terms of the sum of the predecessors' task efforts, then the mechanism is collusion-proof.

Suppose a player $v \in G = (V, E)$ exerts task efforts $t(v)$ due to the referrals of his predecessors I_v^-. If the sum of the diffusion rewards due to the referral of player v is an increasing and bounded above function with respect to the sum of the task efforts of all the predecessors, the mechanism is collusion-proof.

Definition 7.11 (Increasing and bounded function)**.** A function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$ is monotonically increasing and bounded above if the following two conditions hold: (1) for all $x, y \in \mathbb{R}_{\geq 0}$ such that $x \leq y$ one has $f(x) \leq f(y)$; (2) for all $x \in \mathbb{R}_{\geq 0}$ there exists $L \in \mathbb{R}$ where $L = \lim_{x \to +\infty} f(x)$.

The diffusion rewards should be increasing in the sum of the task efforts because the task efforts of the players are verifiable. A group of players can only increase their diffusion efforts by increasing the sum of their total task efforts once the referral chains leading to v have established. The function should be bounded above due to the budget constraint requirement.

Theorem 7.2. *An incentive mechanism is collusion-proof if the mechanism is an increasing and bounded function in terms of the sum of the predecessors' task efforts* $\sum_{u \in I_v^-} t(u)$*.*

The increasing and bounded function provide incentives for referrers to increase their task efforts instead of colluding to reduce the task efforts. Meanwhile, it bounds the diffusion rewards. Note that the increasing and bounded function is not an necessary condition. To see this, we give an example. If the mechanism follows a uniform allocation rule, it is also collusion-proof. However, this rule violates the false-name-proofness constraint.

7.4.3 The GCM

Based on the Theorems 7.1 and 7.2, we introduce the generalized contest mechanism for strategic diffusion in social networks. The mechanism first constructs a superadditive function to determine the diffusion contributions for each player in the successful referral chain. It then uses an increasing and upper bounded function to determine the diffusion rewards.

For each player $v \in G = (V, E)$, if v exerts task efforts $t(v) > 0$ due to the referrals of his predecessors I_v^-. The task rewards for player $u \in G_v$ are computed by

$$\pi_t(u) = \mu \cdot t(u) \ , \tag{7.2}$$

where $\mu \in \mathbb{R}_{>0}$ is the reward parameter that characterizes to what extent the principal values agents' efforts.

The total diffusion rewards for the players in the subgraph G_v rooted at v is $\phi \cdot t(v)$ where $\phi \in \mathbb{R}_{>0}$. Let $D_v = \{u \in I_v^- : dist(u, v) = 1\}$ be the set of player v's direct predecessors. The virtual credits for player v are $b(v, v) = \eta \cdot f(t(v))$, where $f(\cdot)$ is a superadditive function in task efforts t. For each $u \in I_v^-$, the virtual credits for u's diffusion contributions due to v are determined by

$$b(u, v) = \eta \cdot f(t(u)) + t(u) \cdot \frac{\sum_{uu' \in p'} t(u')}{\sum_{uu' \in p} t(u')} \cdot t(v) \cdot (1 - \lambda) \tag{7.3}$$

The proportion of diffusion rewards allocated to each player $u \in G_v$ are calculated according to a ratio-form contest success function

$$h(b(u,v)) = \frac{(b(u,v))^{\rho}}{\sum_{u' \in G_v} (b(u',v))^{\rho}} \ , \tag{7.4}$$

where $\rho \in \mathbb{R}_{>0}$ is the noise factor.

The diffusion rewards for player u due to referring v are calculated by

$$\pi_d(u,v) = \begin{cases} h(b) \cdot t(v) \cdot \phi \cdot g(y) & u \neq v \\ h(b) \cdot t(v) \cdot \phi \cdot g(y) + t_v \cdot \phi(1 - g(y)) & u = v \ , \end{cases} \tag{7.5}$$

where $\lambda \in \mathbb{R}_{>0}$ is the proportion of diffusion credits given to the task worker v, and $0 \leq g(y) \leq 1$ is an increasing and bounded above in terms of y. Here, $y = \sum_{u \in I_v^-} t(u)$ are the total task efforts of player v's predcessors I_v^-.

The total rewards for player $u \in G_v$ are determined by

$$\pi(u,v) = \pi_t(u) + \pi_d(u,v) \ , \tag{7.6}$$

where $\pi_t(u)$ and $\pi_d(u,v)$ are computed by Eq. 7.2 and Eq. 7.5, respectively.

The GCM updates the diffusion rewards for each player in an anytime fashion, making it suitable for large-scale social networks. The social network $G = (V, E)$ can be implemented with adjacency lists. In the data structure, each node maintains a list of all its adjacent edges. It takes $O(|V|)$ time to allocate the diffusion credits, compute the task efforts, and update the rewards for all the newly joined players (See line 3-5 in Algo. 7.1). Computing diffusion credits for each player in the predecessors requires a depth-first search that takes $O(|V| + |E|)$ time (See line 7). To compute diffusion credits for all the players, it takes

Algorithm 7.1 Generalized Contest Mechanism

Input: S-seed nodes in a social network G.
Output: π-payment for each node $v \in G$
Initialize: $\pi(v) = 0$ for all $v \in G$
1: **while** network propagation is **true do**
2:     **for** each newly joined player $v \in G$ **and** $t(v) > 0$ **do**
3:         Assign diffusion credits $b(v, v) \leftarrow \eta \cdot (t(v))^2$
4:         Compute task efforts $\pi_t(u)$ by Eq. 7.2
5:         Update rewards $\pi(u) \leftarrow \pi(u) + \pi_t(u)$
6:         **for** each $u \in I_v^-$ **and** $t(u) > 0$ **do**
7:             Compute diffusion credits $b(u, v)$ by Eq. 7.3
8:         **end for**
9:         **for** each $u \in G_v$ **and** $b(u, v) \geq \eta \cdot f(t(u))$ **do**
10:             Compute diffusion rewards $\pi_d(u, v)$ by Eq. 7.5
11:             Update rewards $\pi(u) \leftarrow \pi(u) + \pi_d(u, v)$
12:         **end for**
13:     **end for**
14: **end while**
15: **return** π

$O(|V| \cdot (|V| + |E|))$ time (See line 6-8). Similarly, it requires $O(|V| \cdot (|V| + |E|))$ time to compute the diffusion rewards and update the total rewards for each player in the social network (See line 9-12). Therefore, the total time complexity for updating the total rewards for all the players in Algo. 7.1 is $O(|V|^2 + |V||E|)$.

The GCM achieves false-name-proofness and collusion-proofness simultaneously by introducing a superadditive function to determine the diffusion contributions and a ratio-form contest success function to compute the proportion of diffusion rewards. The total diffusion rewards for all the referrers and the task worker are capped with an increasing and bounded above function in the total task efforts exerted by the referrers. For each player that has made successful referrals, he also receives a positive diffusion reward. Thus, the mechanism satisfies the voluntary participation constraints. The GCM is budget-constrained because it allocates a fixed portion of rewards for both task efforts and diffusion contributions. Under the GCM, direct referrals are given more diffusion rewards than indirect referrals through decayed diffusion credits. Therefore, the GCM is monotonic. For each newly joined player

v, the GCM updates all the rewards of the players in the same subgraph rooted at v. Hence, the GCM satisfies the subgraph constraint.

Theorem 7.3. *The generalized contest mechanism satisfies false-name-proofness, collusion-proofness, individual rationality, budget constraint, monotonicity, and subgraph constraint.*

Apart from false-name attacks and collusion, a mechanism may be vulnerable to *collusion false-name attacks*: false-name attacks that are results of coordinated action by a group of players.

Corollary 7.1. *The generalized contest mechanism is resistant to collusion false-name attacks.*

When the superadditive function $f(t(u)) = (t(u))^2$ and $g(y) = 1$, the generalized contest mechanism corresponds to the multi-winner contests (MWC) mechanism [Shen et al., 2019]. Note that the MWC mechanism is not collusion-proof as when referrers have the same cost coefficients, they could reduce their task efforts while still receiving the same level of diffusion rewards.

7.5 Experiments

There are different ways to construct the superadditive functions and the increasing and bounded functions. GCM variants with different functions may perform differently. It is desirable to identify the variants that have the best performance and compare their performance with the state-of-the-art mechanisms for strategic network diffusion. In our work, we conducted numerical simulations to show the promise of the generalized contest mechanism on four real-world social network datasets.

Table 7.1: A comparison of the total task efforts for different groups

	GCM - Log			GCM - Lin			GCM-Quad			GCM-Exp			MWC
	Quadratic	Cubic	Quartic	Quadratic	Cubic	Quartic	Quadratic	Cubic	Quartic	Quadratic	Cubic	Quartic	
Twitter	1,992	2,966	3,320	4,119	4,471	5,780	5,041	5,888	6,673	7,142	8,963	9,327	4,080
Flickr	261	303	377	499	581	709	976	1,190	1,483	1,752	1,985	2,324	491
Flixster	120	178	268	391	434	499	650	732	806	945	1,069	1,177	379
Digg	113	170	196	211	242	275	389	418	472	513	603	691	207

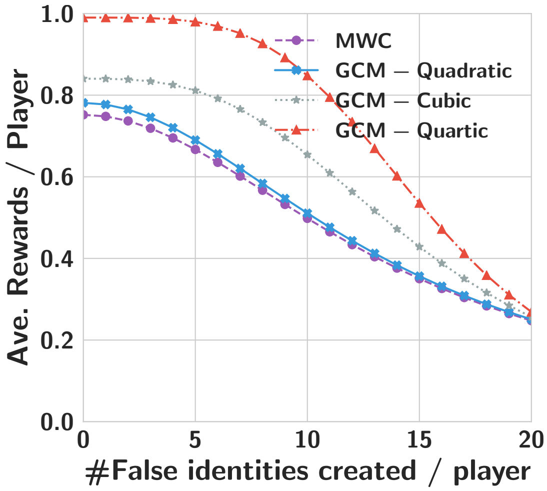
7.5.1 Experimental Settings

Datasets In our experiments, we used four real-world social network datasets: Twitter [Hodas and Lerman, 2014], Flickr [Cha et al., 2009], Flixster [Goyal et al., 2011], and Digg [Hogg and Lerman, 2012]. We estimated the influence probabilities for simulating the diffusion process using the learning algorithms with the Bernoulli distribution under the static model [Goyal et al., 2010]. We next used the general threshold model [Kempe et al., 2003] to simulate the influence diffusion process with the estimated probabilities. After that, each dataset generated a largest weakly connected component of the entire social networks.

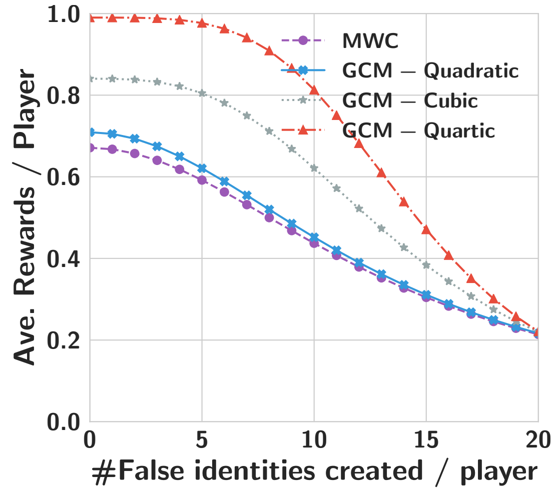
We estimated the efforts according to a "S"-shaped sigmoid curve $1/(1+e^{-(deg^-(v)-0.8*MD)})$ and the cost coefficient is calculated by $\delta_v = -0.1 * 1/(1+e^{-(deg^-(v)-0.8*MD)}) + 0.2$.

Method We measured the total efforts, the average individual efforts, the number of participates, and the social welfare. We used the multi-winner contests mechanism [Shen et al., 2019] as the benchmark. We ran all the experiments 20 times on the same 3.7GHz 6-core machine.

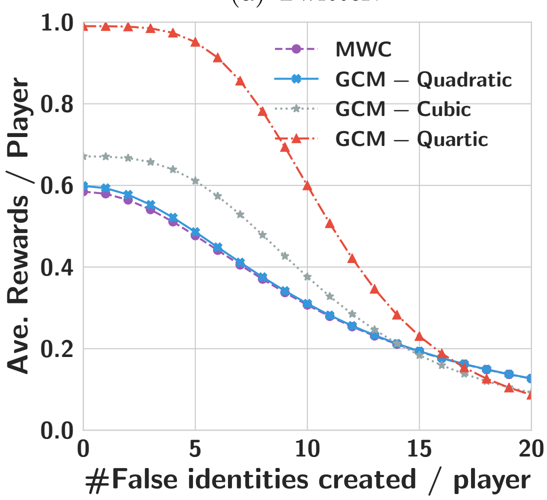
Parameters In our experiments, we set the noise factor as $\rho = 0.55$ because moderate noise factors typically can expect the highest efforts [Shen et al., 2019]. As standard in many geometric mechanisms, we set $\lambda = 0.5$. For fair comparison, we choose $\phi = 0.1$, $\mu = 0.9$, and $\eta = 0.25$.



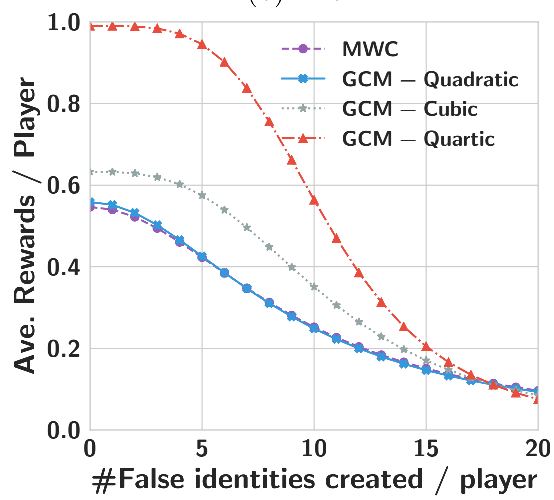
(a) Twitter.



(b) Flickr.



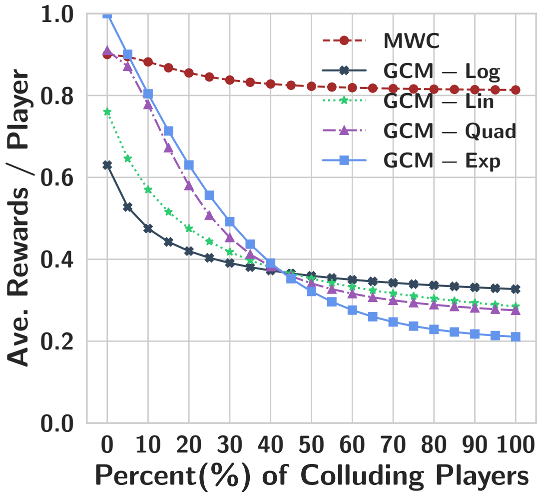
(c) Flixster.



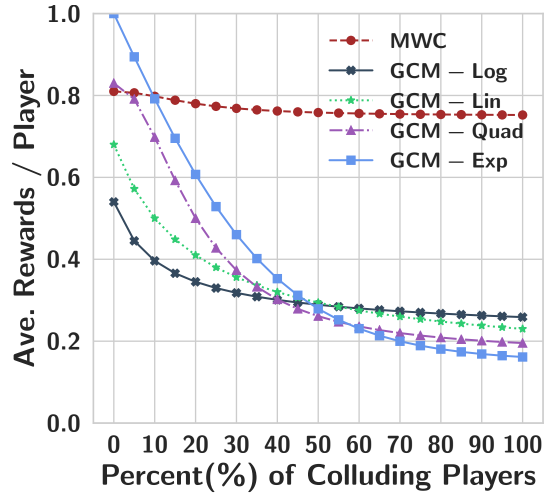
(d) Digg.

Figure 7.5: Average rewards (normalized by the highest number in each dataset) for each player that creates different false identities.

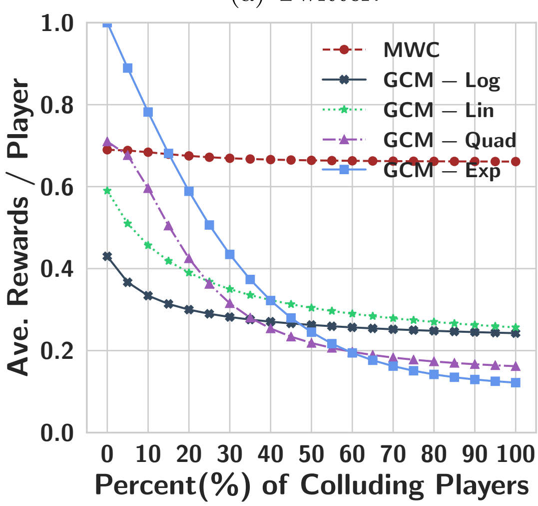
We selected the superadditive functions. We considered the following functions: (1) quadratic: $(t(u))^2$, (2) cubic: $(t(u))^3$, and (3) quartic: $(t(u))^4$. We varied the increasing and bounded above functions. We considered the following functions: (1) linear: $\frac{g(y)}{1+g(y)}$; (2) quadratic: $\frac{(g(y))^2}{1+(g(y))^2}$; (3) exponential: $\frac{e^{g(y)}-1}{e^{g(y)}+1}$; (4) logarithmic: $\frac{\log(g(y)+1)}{1+\log(g(y)+1)}$. This resulted in 12 GCM variants. We compared their performance with the MWC mechanism.



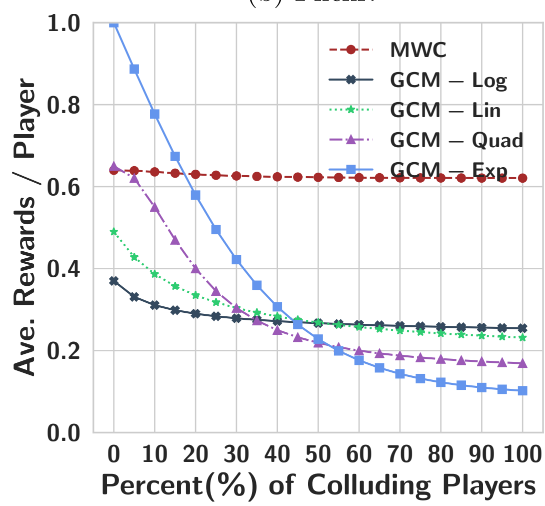
(a) Twitter.



(b) Flickr.



(c) Flixster.



(d) Digg.

Figure 7.6: Average rewards (normalized by the highest number in each dataset) for each player that is in different level of collusion.

7.5.2 Results

We compared the social welfare, the average individual efforts, and the total efforts. Our results indicate that the general contest mechanisms outperform the multi-winner contests mechanism consistently.

Performance without Manipulations When there were no manipulations, all the GCM vairants except the GCM-Log groups outperformed the MWC mechanism. Among them, the

GCM-Exp groups performed the best, followed by the GCM-Quad groups and the GCM-Lin groups (See Table 7.1). The underline reason is that the generalized contest mechanisms with exponential and quadratic form of increasing and bounded functions increased greater than the linear form. The MWC used the linear form. It had a higher increase rate than the logarithmic form. GCMs with quartic functions performed better than the cubic and quadratic form because the quartic forms offered higher incentives for the players to exert task efforts.

Robustness to False-Name Attacks With the same increasing and bounded function (linear), we studied how the selection of superadditive function affects the performance in presence of different levels of false-name attacks. Results show that the degree of polynomials increase the resistance false-name attacks (See Fig. 7.5). This is because higher degree of polynomials provides higher diffusion credits and rewards. Players who have performed false-name attacks suffer from a higher loss of rewards.

Robustness to Collusion With the same superadditive function (quadratic), we compared how the selection of increasing and bounded function affects the performance in presence of different levels of collusion. Results show that MWC experienced the smallest degree of reduction in the rewards, while the GCM-Exp group had the biggest reduction (See Fig. 7.6). This indicates that GCM variants are more resistant to collusion because players would receive substantially less rewards if they colluded under the GCM variants.

Discussion In summary, the generalized contest mechanism can achieve performance on par with the state-of-the-art mechanisms that is resistant to false-name attacks. It provides a higher level of flexibility for stakeholders to tune the parameters to tailor for their specific needs.

7.6 Summary

To counter false-name attacks and collusion, we introduce the generalized contest mechanism. We take a decomposition approach to design the manipulation-resistant mechanisms. The GCM satisfies desirable properties such as individual rationality, budget constraint, monotonicity and subgraph constraint. We compare variants of GCM with the state-of-the-art mechanism for strategic network diffusion empirically. Results show that the GCMs are more robust to manipulations and outperform the MWC mechanism in terms of performance as well. Our work provides a new approach to manipulation-resistant mechanisms.

In our work, we assumed that the budget grows linear in task efforts. This requires an rising market. It will be interesting to study if the budget is fixed or shrinks. Another fruitful problem is to study how to address the challenges when players learn (e.g., their marginal cost coefficients adapt). Another important problem is design manipulation-resistant mechanisms to maximize individual efforts.

Chapter 8

Countering Free Riding in Utility Sharing with Sequential Contests

Free riding is a common problem that arises in many online communities such as peer-to-peer file sharing systems, user-generated content platforms, and open-source software development. Agents' free riding behaviors severely reduce the production of society and can often lead to the tragedy of the commons. In this paper, we introduce a sequential contest mechanism to counter agents' free-riding behavior. The mechanism allocates a number of tokens to each agent that has made contributions to the utility production before a private deadline, and then conducts a ratio-form contest among the qualified agents. The sequential contest mechanism is resistant to free riding and also satisfies the budget constraints. We conducted experiments with real-world dataset to evaluate its performance and robustness. Experimental results demonstrate that stakeholders can achieve optimal results by tuning the parameters of the sequential contest mechanism. Our method brings new insights into how to design free riding resistant mechanisms for utility sharing.

8.1 Background

The free rider problem arises in many real-world scenarios, including crowdfunding [Agrawal et al., 2014], open-source software development [Baldwin and Clark, 2006], user-generated content platforms [Lao, 2010], strategic network diffusion [Shen et al., 2019], and peer-to-peer file sharing systems [Ramaswamy and Liu, 2003]. A free rider usually gain benefits from the contributions made by others without contributing to the development of public resources. For instance, millions of people use Wikipedia — a free encyclopedia every day but only a small fraction of the users pay to support its operation or put efforts into increasing its contents or improving its quality. Most of the users receive benefits from Wikipedia without contributing to its development [Antin and Cheshire, 2010]. These users are often referred as the *free riders*.

Free riding is a key challenge in cooperation among organizations [Hogg and Huberman, 2008]. Agents' free riding behavior is usually harmful to the growth of the public resources. It often makes the public goods unsustainable [Grossman and Hart, 1980]. For example, the Wikipedia would not be able to sustain its operation if no one donates or no one spend time helping improve it. A main reason for the free rider problem is that there is a lack of mechanisms that exclude non-cooperators (e.g., Wikipedia do not prevent users from having access to it) [Antin and Cheshire, 2010].

Another problem in utility sharing is that even some agents choose to contribute, they may not be willing to contribute early [Shen et al., 2018b]. The agents usually have uncertainty about the prospect of the public goods. For instance, if users believe that the Wikipedia would not sustain in the near future, they are unlikely to contribute to its operation. Therefore, stakeholders of the public goods must not only provide incentives to agents that have contributed but also need to incentivize those who have contributed early.

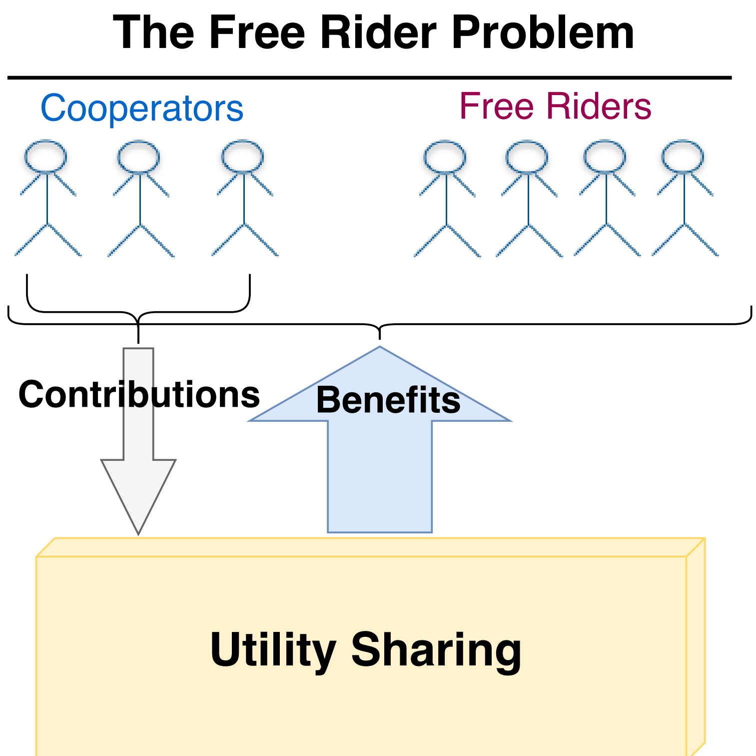


Figure 8.1: An illustration of the free rider problem.

To counter the free riding behavior in utility sharing, we introduce a novel mechanism called the *Sequential Contest Mechanism* (SCM). The sequential contest mechanism initially assign the same number of tokens to each of the newly joined agents. It updates the tokens according to normalized weight of the agent's contributions to the utility sharing. If an agent chooses not to contribute, his weight of contributions will be diluted to a small amount that disqualify him from gaining the benefits. In this way, the mechanism encourages agents to contribute. Another benefits of using sequential contests is that agents have extra incentives to contribute early because early movers have an advantage in the final outcomes in sequential contests. The efforts and payoffs of earlier movers are strictly higher than later players. The earlier players may impact the followers and they do not have direct competitors.

We conducted numerical simulations with real-world data to evaluate the performance of the sequential contest mechanism. Results show that the proposed mechanism can achieve optimal performance through parameter tuning and is resistant to free riding. Our work demonstrates the promising prospective of proper competition in countering free riding in utility sharing.

8.2 The Free Rider Problem in Utility Sharing

This section introduces the free rider problem in utility sharing and formulates the mechanism design problem. Before describing the agents' decision models, we first introduce the notations used.

8.2.1 Notations

We consider discrete time $T = \{1, 2, ...\}$ with a set of players $I = \{1, 2, ...\}$ arrive sequentially in a utility sharing scenario. At each time $t \in T$, a player $i \in I$ that has not exerted his effort previously decides whether to exert his effort $e_{i,t} \in \mathbb{R}_{\geq 0}$ or not. Without loss of generality, we assume that $e_{i,t} \in [0, 1]$ for all $i \in I, t \in T$. For simplicity's sake, we further assume that the total efforts that a player has invested in the utility sharing should be less than or equal to 1. That is, $\sum_{t=1}^{|T|} e_{i,t} \in [0, 1]$. Let $n = |T|$, and $m = |I|$.

The effort investment of player i during the n periods is represented by a vector $\mathbf{e}_i = (e_{i,1}, e_{i,2}, ..., e_{i,n})$. For all $i \in I, t \in T$, we have $e_{i,t} \in [0, 1]$. Thus, $\mathbf{e}_i \in \Theta_i = \{(x_1, x_2, ..., x_n) \in [0, 1]^n | \sum_{t=1}^{n} x_t \leq 1\}$. Therefore, the space for all players' invested efforts $\mathbf{e} \in Theta$ where $\Theta = (\Theta_i)_{i \in I}$. Without loss of generality, we assume that a player only invests his effort in one of the time $t \in T$. That is, if player i invests a total effort of b in the time periods of $|T|$, then we can assume that player i only invests his efforts at time t (i.e., $e_{i,t} = b$), and invests no effort in other times (i.e., $e_{i,t} = 0$ for other times). Let I_t denote the set of players that invest their efforts in time t (i.e., $I_t = \{i \in I | b_{i,t} > 0\}$), and e_t denote the total effort invested from time 1 to t, we have

$$e_t = \sum_{i \in I} \sum_{j \leq t} e_{i,j} \ . \tag{8.1}$$

In utility sharing, the principal is interested in solicits as many efforts as possible. The minimum total invested efforts is 0, while the maximum total invested efforts possible is $m \cdot 1 = m$. Thus, the total invested efforts to the principal $e_t \in [0, m]$. Without loss of generality, the utility of total invested efforts to the principal can be presented as a function $g(b_t)$ where $g : \{0, m\} \to [0, 1]$ such that $g(0) = 0$ and $g(m) = 1$.

In practice, it is often difficult or even unfeasible to solicit the maximum amount of efforts (i.e., $b_t = m$). The principal often has a private threshold $\theta \in (0, 1]$ for her utility. That is, the principal will be interested in soliciting the efforts such that her utility reaches her threshold θ. Assume that the principal's utility g reaches the threshold θ at time $t(\theta, \mathbf{e})$. We have

$$t(\theta, \mathbf{e}) = \begin{cases} \min\{t \in I | g(e_t) \geq \theta\} & \text{if } u(e_t) \geq \theta \text{ for some } 1 \leq t \leq n; \\ -+\infty & \text{otherwise.} \end{cases} \tag{8.2}$$

8.2.2 The Free Rider Problem

In utility sharing, players typically want to maximize their payoffs. At time t, player i receives a utility u_t that is determined by the principal. Note that, the utility u_t is the same for all the players $i \in I$ at time t. Let $t' = t(\theta, \mathbf{e})$, player i's utility at time t is calculated by

$$u(i, t') = u_{t'} \cdot s_{i,t'} - \sum_{t=0}^{t'} e_{i,t} \ , \tag{8.3}$$

where $s_{i,t'} \in [0, 1]$ is player i's estimate of the probability that the principal's utility exceeds her private threshold θ at time t'. In large-scale utility sharing that consists of thousands of players, it is without loss of generality to assume that player i's estimate of the probability $s_{i,t'}$ are independent and identically distributed according to the normal distribution $\mathcal{N}(0,1)$. Thus, the expected value for $s_{i,t'}$ is $\mathbb{E}(s_{i,t'}) = 0.5$.

Without extra incentives, players tend to exert no efforts at all, which results in free riding (e.g., $\sum_{t=0}^{t'} e_{i,t} = 0$). The players receive positive utilities, but contribute nothing to the utility sharing. Eq 8.3 can be simplified as

$$u(i,t') = u_{t'} \cdot s_{i,t'} \ . \tag{8.4}$$

In order to counter the free riding behavior, the principal may provide rewards to the players that have invested efforts into the utility sharing. Let $\pi(i) \in \mathbb{R}_{\geq 0}$ denote the rewards that player i receives in the utility sharing. When the principal receives the rewards, player i's utility is

$$u(i,t') = (u_{t'} + \mathbb{E}(\pi(i))) \cdot s_{i,t'} - \sum_{t=0}^{t'} e_{i,t} \ . \tag{8.5}$$

By Eq. 8.4 and Eq. 8.5, the relative utility for player i not performing free riding is

$$\Delta u(i,t') = (u_{t'} + \mathbb{E}(\pi(i))) \cdot s_{i,t'} - \sum_{t=0}^{t'} e_{i,t} - u_{t'} \cdot s_{i,t'} \tag{8.6}$$

$$= \mathbb{E}(\pi(i)) \cdot s_{i,t'} - \sum_{t=0}^{t'} e_{i,t} \ . \tag{8.7}$$

In our model, we assume that player i's utility is his relative utility $\Delta u(i,t')$. Thus, we have player i's utility in the utility sharing for contributing a effort in one of the time periods is

$$\mathbb{E}(u(i)) = \mathbb{E}(\pi(i)) \cdot s_{i,t'} - \sum_{t=0}^{t'} e_{i,t} \ . \tag{8.8}$$

In order to counter players' free riding behavior, the principal must provide incentives that is increasing in terms players' efforts before the terminating time $t' = t(\theta, \mathbf{e})$.

8.2.3 Mechanism Design in Utility Sharing

In utility sharing, a principal is interested in obtaining the desirable utility production as early as possible by providing incentives to some players that have exerted positive contributions to the utility production. That is, the principal is interested in a utility sharing mechanism π that determines the rewards for each agent that has exert efforts.

Definition 8.1. A utility sharing mechanism π is a tuple of payments for each agent $i \in I$. That is, $\pi = (\pi(i)_{i \in I})$, where $\pi(i) : \Theta \to \mathbb{R}$.

The utility sharing mechanism should be able to counter the free riding behavior. In other words, players would be better off to participate in the investment of their efforts than exerting no efforts at all.

Definition 8.2 (Free-Riding-Proofness)**.** A utility sharing mechanism π is free-riding-proof if for all the players it is better off to participate in the sequential production than not participating. That is, $u(i) \geq 0$ for all $i \in I$ that has invested efforts before $t' = t(\theta, \mathbf{e})$.

To be economically feasible, a utility sharing mechanism should be budget constrained. That is, the total amount of the rewards should not exceed a fixed portion $\vartheta \in \mathbb{R}_{>0}$ of the total aggregated efforts.

Definition 8.3 (Budget Constraint)**.** A utility sharing mechanism π is budget constrained if

$$\sum_{i \in I} \pi(i) \leq \vartheta \cdot e_{t'} \ .$$

8.3 Sequential Contest Mechanism

We introduce the *sequential contest mechanism* (SCM) to encourage agents to contribute to the utility production. The intuition of the SCM is that it performs a sequential contest among the players that have exerted positive efforts before the terminating time $t' = t(\theta, \mathbf{e})$.

At time t, the mechanism first assign the tokens $c_{i,t}$ to a player i that exerts an effort of $e_{i,t} > 0$. At time $t+1$, the mechanism first checks if $t+1 \geq t(\theta, \mathbf{e})$. If it is not, the mechanism updates the tokens for all players by $c_{i,t+1} = \lambda \cdot c_{i,t}$ where $\lambda \in \mathbb{R}_{\geq 1}$. Here λ serves as a discounting factor, it encourages the players to exert efforts as early as possible because early contributions weigh more. If $t+1 \geq t(\theta, \mathbf{e})$, the mechanism conduct a ratio-form contest among all the players that have positive number of tokens. Player i's probability of winning is

$$Pr(c_{i,t'}) = \frac{(c_{i,t'})^{\rho}}{\sum_{j \in I} (c_{j,t'})^{\rho}} \ , \tag{8.9}$$

where $\rho \in (0, 1]$ is the noise factor of the contest, and $t' = t(\theta, \mathbb{e})$. Here, $Pr(c_{i,t'}) \in [\beta, 1]$, where $\beta \in (0, 1]$ is the minimum possibility that the principal set for a player to receive the rewards.

Given the the budget constraint, the mechanism distributes a total amount of $\alpha \cdot e_{t'}$. Therefore, the expected rewards for player i that has exerted efforts before the terminating time t' are

$$\mathbb{E}(\pi(i)) = Pr(c_{i,t'}) \cdot \alpha \cdot e_{t'} \ , \tag{8.10}$$

where $Pr(c_{i,t'})$ and $e_{t'}$ are determined by Eq. 8.10 and Eq. 8.1, respectively.

To satisfy the free-riding-proofness, player i's expected utility should be always non-negative. To achieve this requirement, the parameter α should be carefully chosen. Since $\mathbb{E}(s_{i,t'}) = 0.5$, by Eq. 8.8 Eq. 8.1 and Eq. 8.10, we have

$$
\begin{aligned}
\mathbb{E}(u(i)) &= \frac{1}{2}\mathbb{E}(\pi) - \sum_{t=0}^{t'} e_{i,t} \\
&= \frac{1}{2}\alpha \cdot Pr(c_{i,t'}) \cdot e_{t'} - \sum_{t=0}^{t'} e_{i,t} \\
&= \frac{1}{2}\alpha \cdot Pr(c_{i,t'}) \cdot (\sum_{k\in I}\sum_{j\leq t'} e_{k,j}) - \sum_{t=0}^{t'} e_{i,t} \\
&= \frac{1}{2}\alpha \cdot Pr(c_{i,t'}) \cdot (\sum_{t=0}^{t'} e_{i,t} + \sum_{k\in I/\{i\}}\sum_{j\leq t'} e_{k,j}) - \sum_{t=0}^{t'} e_{i,t} \\
&= (\frac{1}{2}\alpha \cdot Pr(c_{i,t'}) - 1)\sum_{t=0}^{t'} e_{i,t} + \frac{1}{2}\alpha \cdot Pr(c_{i,t'})\sum_{k\in I/\{i\}}\sum_{j\leq t'} e_{k,j} \\
&\geq (\frac{1}{2}\alpha\beta - 1)\sum_{t=0}^{t'} e_{i,t} + \frac{1}{2}\alpha \cdot Pr(c_{i,t'})\sum_{k\in I/\{i\}}\sum_{j\leq t'} e_{k,j} \\
&> (\frac{1}{2}\alpha\beta - 1)\sum_{t=0}^{t'} e_{i,t}
\end{aligned}
\tag{8.11}
$$

Let $\alpha = \frac{2}{\beta}$, we have $\frac{1}{2}\alpha\beta - 1 \geq 0$. Thus, we have $(\frac{1}{2}\alpha\beta - 1)\sum_{t=0}^{t'} e_{i,t} \geq 0$. Therefore, we have $\mathbb{E}(u(i)) > 0$.

The mechanism π only reward the agents that have made positive contributions before the terminating time t' and have a probability of winning that is no less than the minimum value β. The SCM thus satisfies the free-riding-proofness.

The sequential contest mechanism (See Algo. 8.1) operates in a online fashion. Each time, the mechanism checks if the principal's utility of the aggregated efforts exceed the threshold value θ (See line 1, Algo. 8.1). If so, it first updates the tokens of the players that have invested efforts in previous times by multiplying a discounting factor λ (See lines 2–4, Algo. 8.1). It then assign tokens to the newly joined players that have exerted positive efforts (See lines 5–7, Algo. 8.1).

Algorithm 8.1 Sequential Contest Mechanism

Input: θ-the threshold of the utility production.
Output: π-payment for each player $i \in I$
Initialize: $\pi(i) = 0, c_{i,0} = 0$ for all $i \in I$
1: **while** time t is less than $t(\theta, \mathbf{e})$ **do**
2:     **for** existing player $i \in I$ **and** $c_{i,t} > 0$ **do**
3:         Update tokens $c_{i,t+1} \leftarrow \lambda \cdot c_{i,t}$
4:     **end for**
5:     **for** each newly joined player $i \in I$ **and** $e_{i,t} > 0$ **do**
6:         Assign tokens $c_{i,t} \leftarrow e_{i,t}$
7:     **end for**
8: **end while**
9: **for** each player $i \in I$ **and** $c_{i,t'} > 0$ **do**
10:     Calculate player i's $Pr(c_{i,t'})$ by Eq. 8.9.
11:     **if** Player i's $Pr(c_{i,t'}) \geq \beta$ **then**
12:         Update rewards $\pi(i) \leftarrow \pi(i) + Pr(c_{i,t'}) \cdot \alpha \cdot e_{t'}$
13:     **end if**
14: **end for**
15: **return** π

After the production utility reaches the threshold θ, the mechanism conduct a contest among the players that have positive number of tokens (See Lines 9–14, Algo. 8.1). In doing so, the mechanism first calculates the probability of winning according to a ratio-form contest function that is defined by 8.9 (See Line 10, Algo. 8.1). It then updates the rewards for each player *pi* that has a probability of winning no less than the minimum possibility β by adding the rewards $Pr(c_{i,t'}) \cdot \alpha \cdot e_{t'}$ (See line 12, Algo. 8.1).

The SCM π is computationally efficient. It takes $O(|T||I|)$ time to compute the tokens for all the players (See Lines 1–8, Algo. 8.1). Calculating the rewards takes $O(|I|)$ time (See Lines 9–14, Algo. 8.1). Therefore, the time complexity for the SCM is $O(|T||I|)$.

Theorem 8.1 (Free-Riding-Proofness). *The utility sharing mechanism π is free-riding-proof.*

Proof. Let $\beta = \min\{Pr(c_{i,t'})\}$, and $\alpha = \frac{2}{\beta}$. By Eq. 8.11, it follows that $\mathbb{E}(u(i)) > 0$ for all $i \in I$. □

The utility sharing mechanism π satisfies the budget constraint by construction.

Theorem 8.2 (Budget Constraint). *The utility sharing mechanism π is budget constrained.*

Proof. The mechanism π satisfies the budget constraint by construction because the total rewards are $\alpha \cdot e^{t'}$. Let $\vartheta \leq alpha$, we have the total rewards $\sum_{i \in I} \pi(i) = \alpha \cdot e^{t'} \leq \vartheta$. This completes the proof. □

The SCM depends on several parameters such as the noise factor ρ, the discounting factor λ, and the principal's threshold θ. Thus, it is desirable to understand how its performance is influenced by these parameters in practice.

8.4 Experiments

This section describes the numerical experiments we conducted to evaluate the performance and the robustness of the sequential contest mechanism. Before presenting the results, we first introduce the dataset and the method.

8.4.1 Experimental Settings

Dataset In our experiments, we used the Kaggle ranked user data Felipe Salvatore [2018] as the initial dataset. The dataset reflects 4,767 individual users' efforts in Kaggle competitions. The dataset contains 4,767 rows with each row representing a ranked player. In each row, there are six data fields: the register date of the user, the current points earned by the user, the current ranking of the user, the highest ranking of the user, the country that the user is located at, and the continent that the user belongs to. In our work, we removed two irrelevant data fields: the country and the continent.

The original dataset does not contains all the necessary data fields for the experiments. Before conducting the experiments, we performed data processing. We first normalized the players' data points by dividing each value by the maximum value observed in the data samples. We then selected the normalized data points as lower bound of the efforts that minimum efforts that the players would receive non-negative expected utilities (i.e., $E(u(i)) \geq 0$). Let e'_i denote player i's lower bound of the effort, we estimated the maximum efforts that player i could invest by

$$\hat{e}_i = e'_i \cdot \frac{\hat{r}_i}{r_i} , \tag{8.12}$$

where r_i, and $\hat{r_i}$ denote the current ranking and the highest ranking of player i, respectively.

To estimate the expected utility, an agent needs to have her cost coefficient. Unfortunately, the original dataset does not include this information. We estimated each agent's cost efficient δ_i by assuming her utility was zero. This is without of loss generality because agent i's utility is positively correlated to the cost coefficient according to Eq. 8.10. Thus, we have

$$\delta_i = \frac{R - r_i}{R \cdot e_i} , \tag{8.13}$$

where $R = 4767$, and e_i is player i's normalized points number.

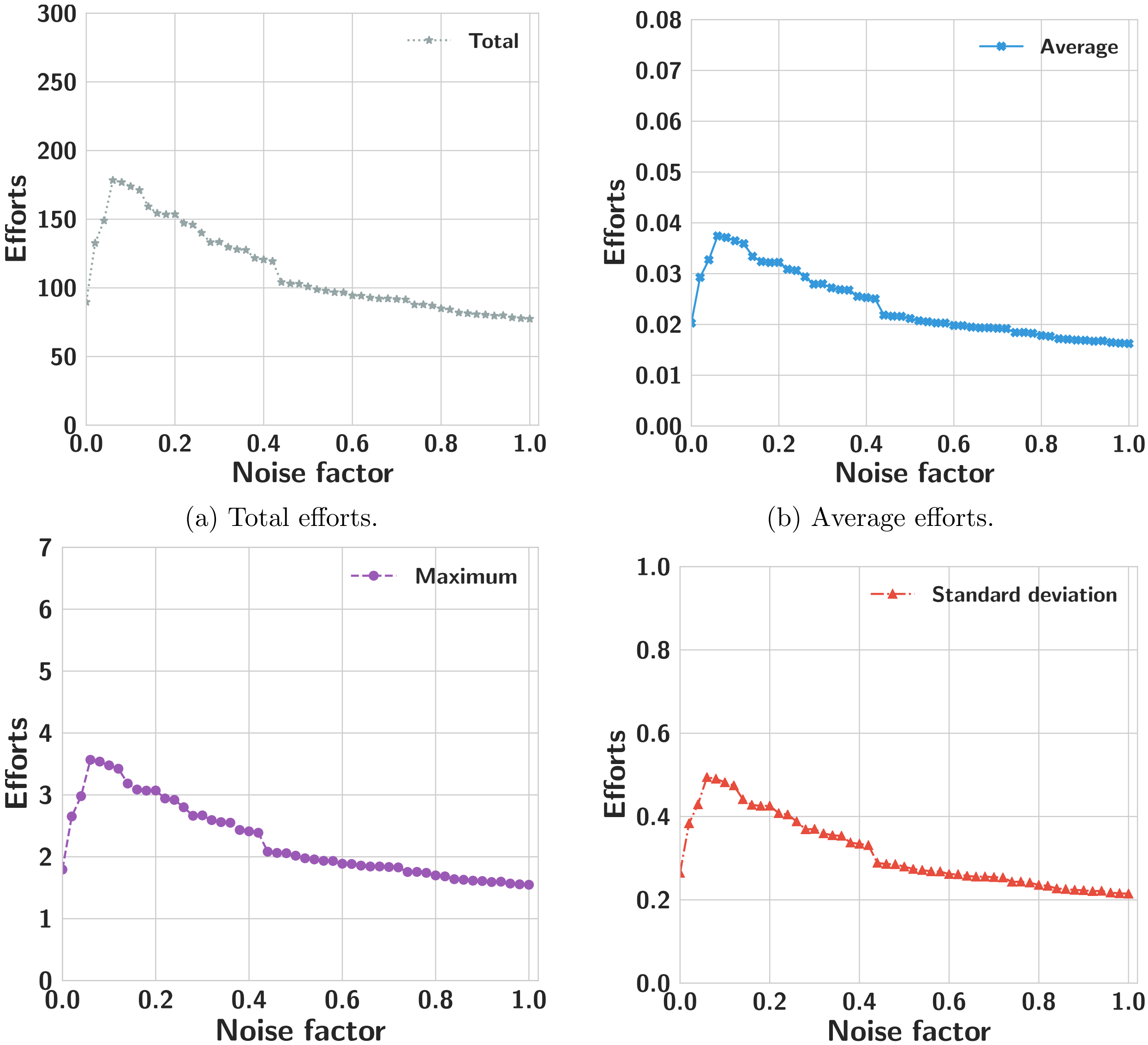
Method We selected the total efforts and the total utilities as the main metrics. The principal was interested in the total efforts while the players are interested in the total utilities. We also compared the average efforts, standard deviation of the efforts, the average utilities, and the standard deviation of the utilities for the first group of experiments. We performed four groups of experiments. In the first group of experiments, we compared the performance of the sequential contest mechanisms with different noise factors. In the second group, we varied the discounting factor to study how different discounting factors influenced the performance of the sequential contest mechanism. In the third one, we evaluated the performance of the sequential contest mechanism when we changed the threshold of the utility production. In the last group, we measured the robustness of the mechanism as the percentage of free riders increased.

We ran each group of experiments for 30 times and reported the average numbers of each metrics. All the simulations were conducted on the same 3.7 GHz 6-core linux machine.

8.4.2 Results

Noise Factor We varied the noise factor from 0.0 to 1.0 with an increment of 0.02. Fig. 8.2 shows that all the four effort metrics – the total efforts, the average efforts, the maximum efforts, and the standard deviation of individual efforts initially increased to the peaks and then fall gradually as the noise factor increased. The trends indicated that the generated contest mechanism performed the best when the noise factor is low (e.g., noise factor $\rho = 0.06$). Fig. 8.3 demonstrates that the players' utilities first increased when the noise factors were low (e.g., noise factor $\rho \leq 0.1$) and then decreased with fluctuations as the noise factor increased. From Figs. 8.2 and 8.3, we can see that a low noise factor (e.g., $\rho = 0.06$) is the most beneficial to both the principal and the players. Two factors contributed to this phenomenon: a majority of the agents were players with common or low ability (i.e., they had high cost coefficients); contests with a low noise factor favor players with average abilities Jia et al. [2013], Shen et al. [2019]. Thus, a low noise factor is typically optimal for the principal to solicit efforts while also providing incentives for agents to contribute.

Discounting Factor We evaluated the total efforts and the total utility when we changed the discounting factor λ from 1.00 to 1.10 with an increment of 0.0005. Fig. 8.4 demonstrates that both the total efforts and the total utility increased gradually to the peaks when the discounting factor was ranging from 1.05 to 1.07. They then dropped slightly. The sequential contest mechanism received the highest performance for both metrics when the discounting



(a) Total efforts.

(b) Average efforts.

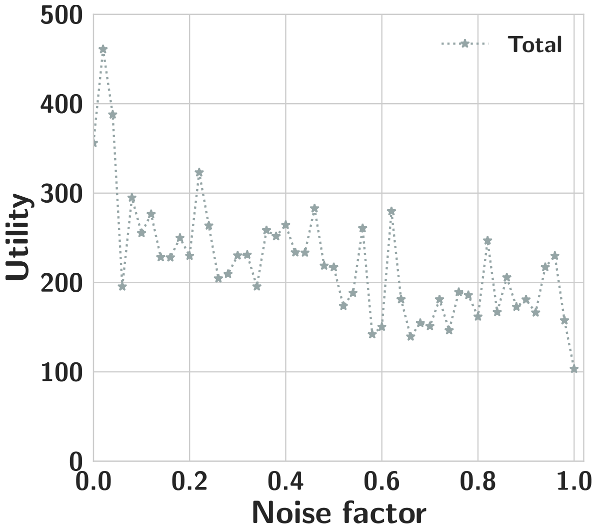
(c) Maximum efforts.

(d) S.D. efforts.

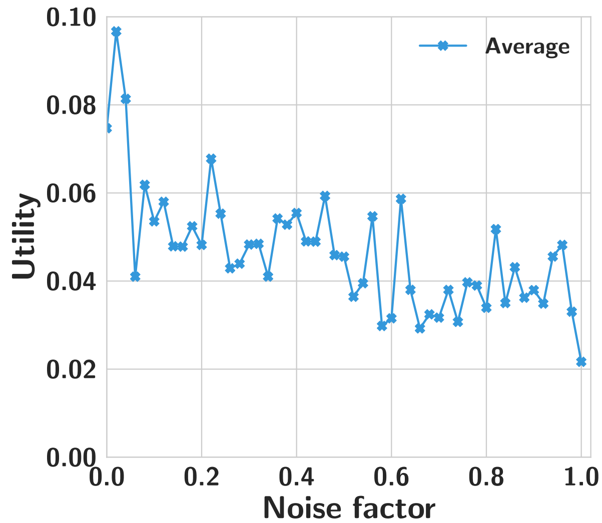
Figure 8.2: A comparison of four effort metrics by varying the noise factor from 0.0 to 1.0.

factor was moderately high (e.g., discounting factor $\lambda = 1.06$). An explanation for this trend was that a relatively high discounting factor encouraged players to contribute early. This would enable to receive more efforts before the deadline (i.e., the time when the utility production exceeded the threshold).

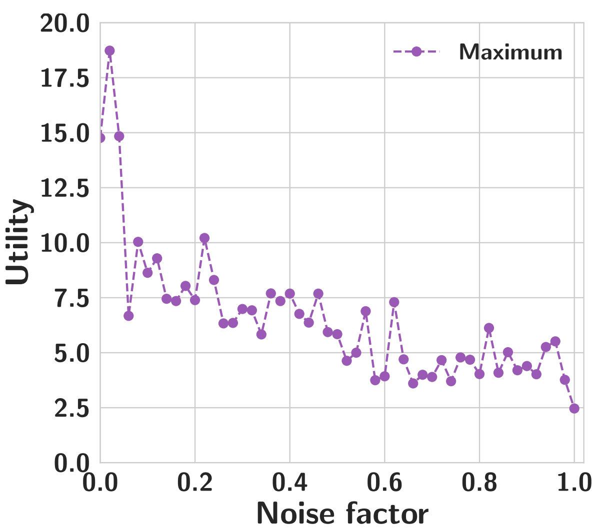
Threshold of Production We measured the performance of the sequential contest mechanism when we changed the threshold of utility production from 0.0 to 1.0 with an increment



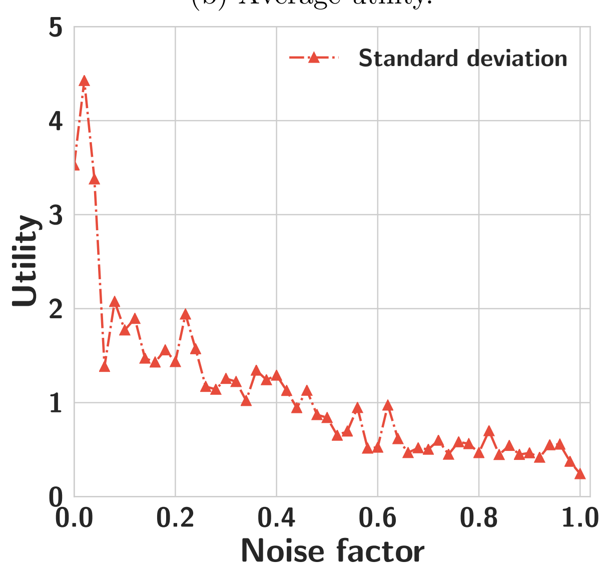
(a) Total utility.



(b) Average utility.



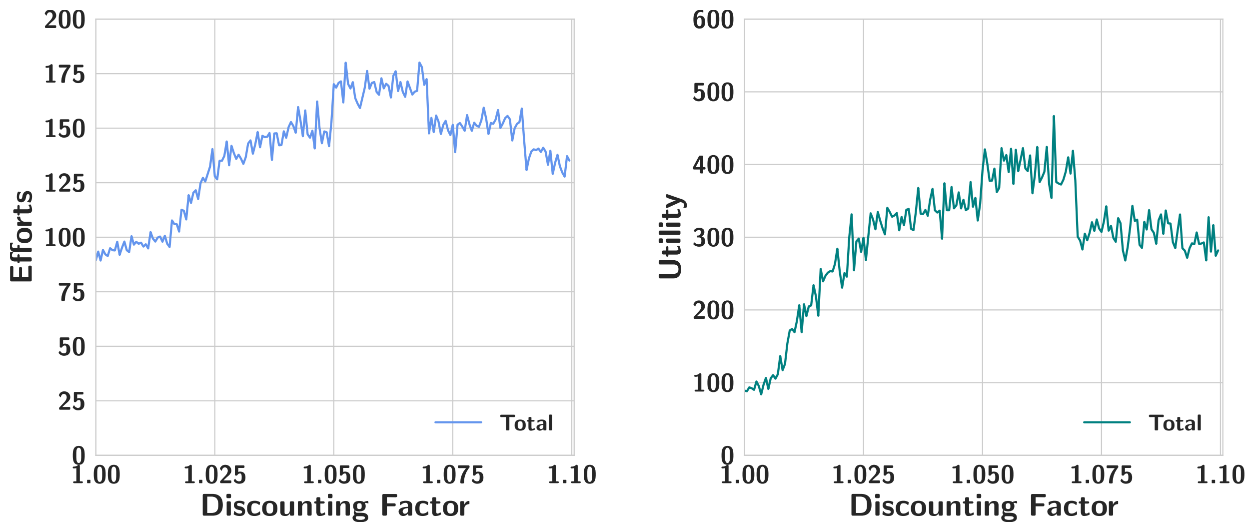
(c) Maximum utility.



(d) S.D. utility.

Figure 8.3: A comparison of four utility metrics by varying the noise factor from 0.0 to 1.0.

of 0.005. As the threshold increased, the total efforts and the total utility first increased significantly until $\theta = 0.05$ and then fluctuated around the same level (See. Fig. 8.5). The reason is that initially the players needed to exert more efforts to receive a higher utility as the threshold of utility production increased. When the threshold reached a level, the players did not exert more efforts because otherwise they would receive fewer utilities or the required efforts were beyond their capacities.



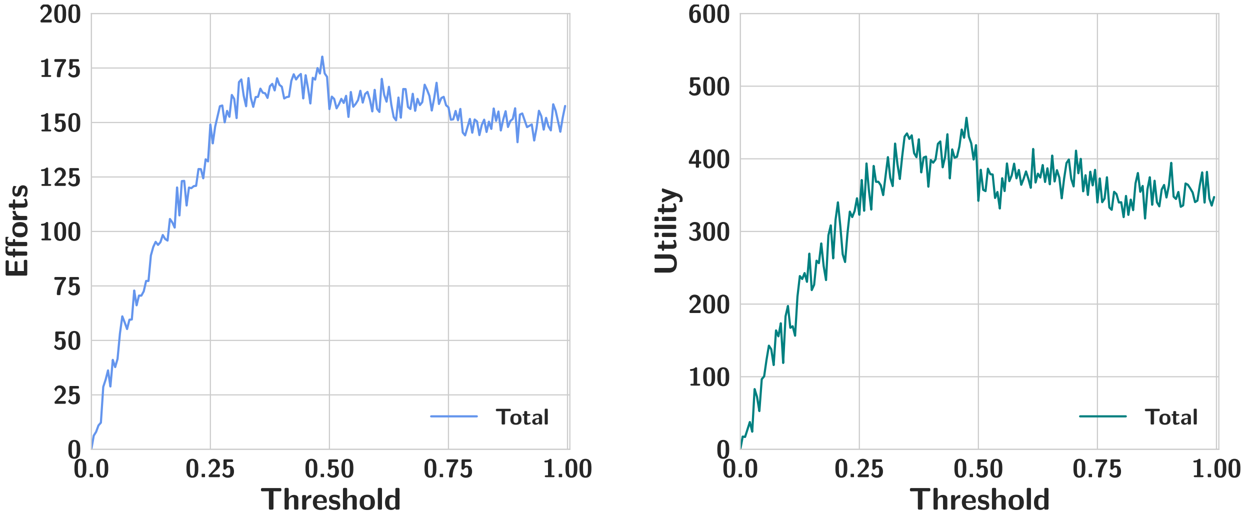
(a) Total efforts

(b) Total utility.

Figure 8.4: A comparison of total efforts and total utility by varying the discounting factor from 1.00 to 1.10 with threshold $\theta = 0.50$, noise factor $\rho = 0.06$.

Robustness to Free Riding To evaluate the robustness of the sequential contest mechanism, we evaluated the performance of the mechanism by varying the percentage of the free riders from 0% to 100% with an increment of 0.5%. Fig. 8.6 demonstrates that both the total efforts and the total utility first dropped sharply and then decreased gradually as the percentage of free riders increased. This results further indicate that the sequential contest mechanism is free-riding-proof.

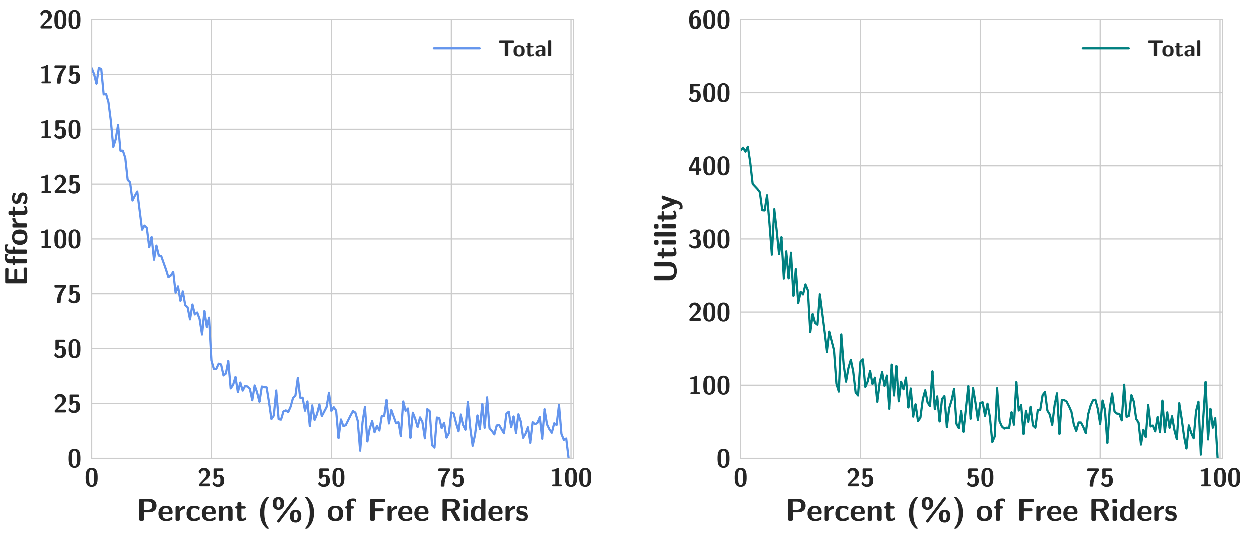
Discussion Via extensive experiments, we demonstrate that stakeholders can both optimize the total efforts and counter agents' free riding behavior when they use the sequential contest mechanisms with proper (optimal) configurations (e.g., noise factor $\rho = 0.06$, threshold $\theta = 0.05$, and discounting factor $\lambda = 1.06$ in our experiments).



(a) Total efforts

(b) Total utility.

Figure 8.5: A comparison of total efforts and total utility by varying the threshold from 0.0 to 1.0 with noise factor $\rho = 0.06$ and discounting factor $\lambda = 1.06$.



(a) Total efforts.

(b) Total utility

Figure 8.6: A comparison of total efforts and total utility by percentage of free riders from 0% to 100% with threshold $\theta = 0.50$, noise factor $\rho = 0.06$ and discounting factor $\lambda = 1.06$.

8.5 Related Work

Free riding reduces the efficiency of the utility sharing in the sense that it results in the under-production or over-consumption of the utility (or the good). It is harmful to both the stakeholders and the individuals that have made contributions [Grossman and Hart, 1980, Kim and Walker, 1984].

Because of its significance, many attempts have been made to address the free rider problem. For instance, Groves et al. [1977] introduced government provision or taxation to achieve the Pareto-optimal allocation of public goods. This methods imposed taxes on individuals that receive benefits from the utility sharing. It is convenient the the stakeholders have the authority but not feasible for non-government entities or individuals. Another method to counter the free riding behavior is through assurance contracts [Runge, 1984, Tabarrok, 1998]. This approach is suitable to the scenarios (e.g., crowdfunding) when the stakeholders can provide guarantees or refunds when the utility production fails to be delivered as promised. When the stakeholders cannot provide such assurance, however, this method is not implementable in practice. If the stakeholders can exclude individual players, they can established privileged groups such as clubs [Hampton, 1987]. Alternatively, the stakeholders may use the Coasian solution to pool the resources until a goal of production has been met [Coase, 1960]. Despite the insightfulness of these methods, they require strong assumption on the rationality of the agents usually does not hold in practice. This is particularly true in scenarios that consist of a large number of agents.

Our work distinguishes itself from previous work in three aspects. First, it does not impose strong rationality assumption on agents. The agents have private thresholds (i.e., cost coefficient) to trigger their actions. Second, we take a different approach to counter agents' free riding behavior: introducing a sequential contest to increase the competitions among the agents. In order to receive rewards, the agents needs to compete with each other by investing efforts in utility production. This practice enables the stakeholders to attract contributions from the players that otherwise would not exert their efforts. Third, with appropriate configurations of the sequential contest mechanism, both the stakeholders and the players can receive the reasonably good utilities.

8.6 Summary

To counter agents' free riding behavior in utility sharing, we introduce a sequential contest mechanism. The mechanism rewards players that have made contributions before the realization of the utility production. It distributes the rewards through a ratio-form contest among the qualified players. We demonstrates that the mechanism is resistant to free riding and also satisfies the budget constraints. Through extensive experiments on real-world data, we demonstrate that the sequential contest mechanism can produce good performance when the parameters (e.g., the noise factor, the threshold, and the discounting factor) of contest are selected appropriately. The experimental results also confirmed the mechanism's robustness to free riders. Our work advances the state-of-the-art by offering a new approach to countering agents' free riding behavior in utility sharing.

Our work opens several exciting revenues for further research. In our model, we assumed that the agents have fixed cost coefficient. It would be interesting to study how to counter agents' free riding behavior when they have adaptive cost coefficient. A possible solution is through online learning of agents' preferences [Fürnkranz and Hüllermeier, 2010]. Another fruitful area is to develop novel methods to automatically select optimal parameters of the sequential contest mechanism. To do so, the stakeholders need to conduct inference and perform realistic simulations with real-world data. We also find it very rewarding to bring contests into other domains such as cybersecurity and disaster response.

Chapter 9

Conclusions and Future Directions

In this dissertation, we introduced a unified framework called *mechanism design with thresholding agents* to relax some of the unrealistic assumptions (i.e., full rationality, direct preference revelation, and no group manipulations) in mechanism design. We demonstrated the promising prospects by applying the framework to addressing real-world challenges in crowdfunding, transportation systems, information diffusion, and utility sharing.

9.1 Our Contributions

Chapters 2 – 3 discussed the background of traditional mechanism design and the theory of mechanism design with thresholding agents.

Chapter 4 proposed to use information design as a tool to help the entrepreneur to increase revenue by influencing backers' beliefs of a campaign's probability of success. In this work, we introduced a heuristic algorithm to dynamically compute information-disclosure policies for the entrepreneur and conducted numerical simulations to demonstrate its competitiveness over the widely-adopted immediate-disclosure policy. The experimental results demonstrate that the immediate-disclosure policy is not optimal when backers use cutoff policies. With appropriate heuristics, an entrepreneur can benefit from dynamic information disclosure.

Our work presented the first study on information design where a sender interacts with multiple receivers that follow thresholding policies.

Chapter 5 introduced a post-price online mechanism called the *Integrated Online RideSharing* (IORS) mechanism to promote ridesharing in autonomous mobility-on-demand systems. We showed that the IORS mechanism is ex-post incentive compatible. Simulation results demonstrated that the IORS outperformed the auction-based mechanism and was comparable to the optimal offline algorithm.

Chapter 6 proposed a novel multi-winner contests mechanism to counter false-name attacks in strategic information diffusion. The mechanism is false-name-proof, individually rational, budget-constrained, monotonic, subgraph-constrained, and computationally efficient. Experimental results demonstrated that false-name attacks are unprofitable under the MWC mechanism. Stakeholders can significantly boost the aggregated efforts of players when they slect parameters of the MWC mechanism appropriately.

Chapter 7 presented a generalized contest mechanism to counter both false-name attacks and collusion. We identified conditions for a diffusion mechanism to be false-name-proof and collusion-proof. The generalized contest mechanism integrates both conditions into a single mechanism. It is both false-name-proof and collusion-proof. Experimental results showed that the generalized contest mechanism not only can beat the state-of-the art mechanism in terms of total efforts but is also more robust to false-name attacks and collusive manipulations.

Chapter 8 introduced a sequential contest mechanism to counter agents' free riding behavior by providing rewards to the players that have made contributions before the realization of the utility production. The mechanism distributes the rewards through a ratio-form contest among qualified players. We demonstrated that the sequential contest mechanism is resistant to free riding and can produce good performance when the parameters of the contest are selected appropriately. Our work provides a new approach to countering agents' free riding behavior in utility sharing.

9.2 Future Directions

This dissertation opens several avenues for future research. In our model, we assumed that the agents have fixed thresholds. It would be rewarding to study how to design mechanisms when agents have dynamic thresholds. A possible solution is through online learning of agents' preferences.

Another fruitful area is to develop algorithms to automatically select the parameters of the mechanisms. To do so, the mechanism designer needs to conduct inference and perform realistic simulations from real-world data.

It is also interesting to bring the framework of mechanism design with thresholding agents to other domains. For example, in cybersecurity, a defender can use the framework to design mechanisms that have different level of security given economical or physical constraints. To promote open source software development, stakeholders can use contests to provide high incentives to contribute.

Bibliography

A. Abdulkadiroğlu and T. Sönmez. School choice: A mechanism design approach. *American economic review*, 93(3):729–747, 2003.

A. Agrawal, C. Catalini, and A. Goldfarb. Some simple economics of crowdfunding. *Innovation Policy and the Economy*, 14(1):63–97, 2014.

A. K. Agrawal, C. Catalini, and A. Goldfarb. The geography of crowdfunding. Technical report, National bureau of economic research, 2011.

S. Alaei, A. Malekian, and M. Mostagir. A dynamic model of crowdfunding. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 363–363. ACM, 2016.

R. Alonso and O. Camara. Bayesian persuasion with heterogeneous priors. *Journal of Economic Theory*, 165:672–706, 2016.

A. Alshamsi, F. L. Pinheiro, and C. A. Hidalgo. Optimal diversification strategies in the networks of related products and of related research areas. *Nature communications*, 9(1): 1328, 2018.

J. Antin and C. Cheshire. Readers are not free-riders: reading as a form of participation on wikipedia. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 127–130. ACM, 2010.

W. B. Arthur and D. A. Lane. Information contagion. *Structural Change and Economic Dynamics*, 4(1):81–104, 1993.

P. H. Au. Dynamic information disclosure. *The RAND Journal of Economics*, 46(4):791–823, 2015.

M. Babaioff, S. Dughmi, R. Kleinberg, and A. Slivkins. Dynamic pricing with limited supply. *ACM Transactions on Economics and Computation*, 3(1):4, 2015.

C. Y. Baldwin and K. B. Clark. The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science*, 52(7): 1116–1127, 2006.

A. Bandura. Human agency in social cognitive theory. *American psychologist*, 44(9):1175, 1989.

D. Bergemann and S. Morris. Bayes correlated equilibrium and the comparison of information structures in games. *Theoretical Economics*, 11(2):487–522, 2016.

D. Bergemann and M. Pesendorfer. Information structures in optimal auctions. *Journal of Economic Theory*, 137(1):580–609, 2007.

D. Bergemann and A. Wambach. Sequential information disclosure in auctions. *Journal of Economic Theory*, 159:1074–1095, 2015.

M. Berkelaar, K. Eikland, P. Notebaert, et al. lpsolve: Open source (mixed-integer) linear programming system. *Eindhoven U. of Technology*, 2004.

B. Berman. Referral marketing: Harnessing the power of your customers. *Business Horizons*, 59(1):19–28, 2016.

L. I. Besaleva and A. C. Weaver. Applications of social networks and crowdsourcing for disaster management improvement. In *2013 International Conference on Social Computing*, pages 213–219. IEEE, 2013.

J. Bibby. Axiomatisations of the average and a further generalisation of monotonic sequences. *Glasgow Mathematical Journal*, 15(1):63–65, 1974.

D. Blackwell et al. Equivalent comparisons of experiments. *The Annals of Mathematical Statistics*, 24(2):265–272, 1953.

T. Börgers. *An introduction to the theory of mechanism design*. Oxford University Press, USA, 2015.

M. Brill, V. Conitzer, R. Freeman, and N. Shah. False-name-proof recommendations in social networks. In *AAMAS 2016*, pages 332–340, 2016.

A. Burnap, Y. Ren, P. Y. Papalambros, R. Gonzalez, and R. Gerth. A simulation based estimation of crowd ability and its influence on crowdsourced evaluation of design concepts. In *the ASME 2013 International Design Engineering Technical Conferences*, 2013.

T. N. Cason, W. A. Masters, and R. M. Sheremeta. Entry into winner-take-all and proportional-prize contests: An experimental study. *Journal of Public Economics*, 94 (9-10):604–611, 2010.

T. N. Cason, W. Masters, and R. Sheremeta. Winner-take-all and proportional-prize contests: theory and experimental results. *Journal of Economic Behavior and Organization*, 2018.

B. Caulfield. Estimating the environmental benefits of ride-sharing: A case study of dublin. *Transportation Research Part D: Transport and Environment*, 14(7):527–531, 2009.

M. Cha, A. Mislove, and K. P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *Proc. of WWW 2009*, pages 721–730, 2009.

D. Chaffey. Global social media research summary 2016. *Smart Insights: Social Media Marketing*, 2016.

C. P. Chambers and F. Echenique. *Revealed preference theory*, volume 56. Cambridge University Press, 2016.

W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 199–208. ACM, 2009.

S.-F. Cheng, D. T. Nguyen, and H. C. Lau. Mechanisms for arranging ride sharing and fare splitting for last-mile travel demands. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, pages 1505–1506. International Foundation for Autonomous Agents and Multiagent Systems, 2014.

Z. Chong, B. Qin, T. Bandyopadhyay, T. Wongpiromsarn, B. Rebsamen, P. Dai, E. Rankin, and M. H. Ang Jr. Autonomy for mobility on demand. In *Intelligent Autonomous Systems 12*, pages 671–682. Springer, 2013.

E. H. Clarke. Multipart pricing of public goods. *Public choice*, 11(1):17–33, 1971.

R. H. Coase. The problem of social cost. In *Classic papers in natural resource economics*, pages 87–137. Springer, 1960.

M. G. Colombo, C. Franzoni, and C. Rossi-Lamastra. Internal social capital and the attraction of early contributions in crowdfunding. *Entrepreneurship Theory and Practice*, 39(1): 75–100, 2015.

B. J. Coltin and M. Veloso. Towards ridesharing with passenger transfers. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, pages 1299–1300. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

V. Conitzer and T. Sandholm. Revenue failures and collusion in combinatorial auctions and exchanges with vcg payments. In *International Workshop on Agent-Mediated Electronic Commerce*, pages 1–14. Springer, 2004.

V. Conitzer and T. Sandholm. Failures of the vcg mechanism in combinatorial auctions and exchanges. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 521–528. ACM, 2006.

V. Conitzer, N. Immorlica, J. Letchford, K. Munagala, and L. Wagman. False-name-proofness in social networks. In *International Workshop on Internet and Network Economics*, pages 209–221. Springer, 2010.

E. Constantinides, C. L. Romero, and M. A. G. Boria. Social media: a new frontier for retailers? In *European Retail Research*, pages 1–28. Springer, 2008.

J. W. Crandall. Towards minimizing disappointment in repeated games. *Journal of Artificial Intelligence Research*, 49:111–142, 2014.

M. De Choudhury, M. Gamon, S. Counts, and E. Horvitz. Predicting depression via social media. *ICWSM*, 13:1–10, 2013.

N. R. Devanur and J. D. Hartline. Limited and online supply and the bayesian foundations of prior-free mechanism design. In *Proceedings of the 10th ACM conference on Electronic commerce*, pages 41–50. ACM, 2009.

S. Dow, A. Kulkarni, S. Klemmer, and B. Hartmann. Shepherding the crowd yields better work. In *CSCW 2012*, pages 1013–1022, 2012.

F. A. Drucker and L. K. Fleischer. Simpler sybil-proof mechanisms for multi-level marketing. In *EC 2012*, pages 441–458, 2012.

D. Easley and A. Ghosh. Behavioral mechanism design: Optimal crowdsourcing contracts and prospect theory. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 679–696. ACM, 2015.

B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American economic review*, 97 (1):242–259, 2007.

M. Ellman and S. Hurkens. Optimal crowdfunding design. *Available at SSRN 2709617*, 2015.

Y. Emek, R. Karidi, M. Tennenholtz, and A. Zohar. Mechanisms for multi-level marketing. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 209–218. ACM, 2011.

G. Enli. Twitter as arena for the authentic outsider: exploring the social media campaigns of trump and clinton in the 2016 us presidential election. *European journal of communication*, 32(1):50–61, 2017.

M. Feldman and J. Chuang. Overcoming free-riding behavior in peer-to-peer systems. *ACM sigecom exchanges*, 5(4):41–50, 2005.

Felipe Salvatore. Ranked users kaggle data. https://www.kaggle.com/felsal/ranked-users-kaggle-data/, 2018. accessed 03rd January 2019.

E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 2016.

J. Fürnkranz and E. Hüllermeier. *Preference learning*. Springer, 2010.

M. Furuhata, M. Dessouky, F. Ordóñez, M.-E. Brunet, X. Wang, and S. Koenig. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57:28–46, 2013.

A. Galeotti and S. Goyal. Influencing the influencers: a theory of strategic diffusion. *The RAND Journal of Economics*, 40(3):509–532, 2009.

J. Gallien. Dynamic mechanism design for online commerce. *Operations Research*, 54(2): 291–310, 2006.

H. Gao, G. Barbier, R. Goolsby, and D. Zeng. Harnessing the crowdsourcing power of social media for disaster relief. Technical report, Arizona State Univ Tempe, 2011.

H. Gao, C. H. Liu, W. Wang, J. R. Zhao, Z. Song, X. Su, J. Crowcroft, and K. K. Leung. A survey of incentive mechanisms for participatory sensing. *IEEE Communications Surveys and Tutorials*, 17(2):918–943, 2015.

A. Ghosh and R. Kleinberg. Optimal contest design for simple agents. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 913–930. ACM, 2014.

G. Gigerenzer and D. G. Goldstein. Reasoning the fast and frugal way: models of bounded rationality. *Psychological review*, 103(4):650, 1996.

A. V. Goldberg and J. D. Hartline. Competitiveness via consensus. In *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 215–222. Society for Industrial and Applied Mathematics, 2003.

A. Goyal, F. Bonchi, and L. V. Lakshmanan. Learning influence probabilities in social networks. In *WSDM 2010*, pages 241–250, 2010.

A. Goyal, F. Bonchi, and L. V. Lakshmanan. A data-based approach to social influence maximization. *Proc. of the VLDB Endowment*, 5(1):73–84, 2011.

S. J. Grossman and O. D. Hart. Takeover bids, the free-rider problem, and the theory of the corporation. *The Bell Journal of Economics*, pages 42–64, 1980.

T. Groves, J. Ledyard, et al. Optimal allocation of public goods: A solution to the free rider problem. *Econometrica*, 45(4):783–809, 1977.

B. Q. Ha and J. D. Hartline. Mechanism design via consensus estimates, cross checking, and profit extraction. *ACM Transactions on Economics and Computation*, 1(2):8, 2013.

J. Hampton. Free-rider problems in the production of collective goods. *Economics & Philosophy*, 3(2):245–273, 1987.

E. A. Hansen and R. Zhou. Anytime heuristic search. *Journal of Artificial Intelligence Research*, 28:267–297, 2007.

G. Hardin. The tragedy of the commons. *science*, 162(3859):1243–1248, 1968.

P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4(2): 100–107, 1968.

K. Hendricks and R. H. Porter. Collusion in auctions. *Annales d'Economie et de Statistique*, pages 217–230, 1989.

E. v. Hippel and G. v. Krogh. Open source software and the "private-collective" innovation model: Issues for organization science. *Organization science*, 14(2):209–223, 2003.

C. A. Hoare. Quicksort. *The Computer Journal*, 5(1):10–16, 1962.

N. O. Hodas and K. Lerman. The simple rules of social contagion. *Scientific Reports*, 4: 4343, 2014.

T. Hogg and B. A. Huberman. Solving the organizational free riding problem with social networks. In *AAAI Spring Symposium: Social Information Processing*, pages 24–29, 2008.

T. Hogg and K. Lerman. Social dynamics of digg. *EPJ Data Science*, 1(1):5, 2012.

J. E. Hopcroft and R. M. Karp. An nˆ5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2(4):225–231, 1973.

M. E. Horn. Fleet scheduling and dispatching for demand-responsive passenger services. *Transportation Research Part C: Emerging Technologies*, 10(1):35–63, 2002.

J. Huang, R. A. Berry, and M. L. Honig. Auction-based spectrum sharing. *Mobile Networks and Applications*, 11(3):405–418, 2006.

L. Hurwicz and S. Reiter. *Designing economic mechanisms*. Cambridge University Press, 2006.

P. G. Ipeirotis. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students*, 17(2):16–21, 2010.

M. O. Jackson. Mechanism theory. *Available at SSRN 2542983*, 2014.

M. O. Jackson and L. Yariv. Diffusion, strategic interaction, and social structure. In *Handbook of Social Economics*, volume 1, pages 645–678. Elsevier, 2011.

H. Jia, S. Skaperdas, and S. Vaidya. Contest functions: Theoretical foundations and issues in estimation. *International Journal of Industrial Organization*, 31(3):211–222, 2013.

L. Jin, H. Takabi, and J. B. Joshi. Towards active detection of identity clone attacks on online social networks. In *Proceedings of the first ACM conference on Data and application security and privacy*, pages 27–38. ACM, 2011.

D. Kahneman. A perspective on judgment and choice: mapping bounded rationality. *American Psychologist*, 58(9):697, 2003.

E. Kamar and E. Horvitz. Collaboration and shared plans in the open world: studies of ridesharing. In *Proceedings of the 21st International Joint Conference on Artifical Intelligence*, pages 187–194. Morgan Kaufmann Publishers Inc., 2009.

E. Kamenica and M. Gentzkow. Bayesian persuasion. *The American Economic Review*, 101 (6):2590–2615, 2011.

F. Kamyab, M. Amini, S. Sheykhha, M. Hasanpour, and M. M. Jalali. Demand response program in smart grid using supply function bidding mechanism. *IEEE Trans. Smart Grid*, 7(3):1277–1284, 2016.

P. Kau and L. Hill. A threshold model of purchasing decisions. *Journal of Marketing Research*, pages 264–270, 1972.

D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.

Kickstarter. Kickstarter Stats. https://www.kickstarter.com/help/stats, 2017. Online; accessed November 10th, 2017.

O. Kim and M. Walker. The free rider problem: Experimental evidence. *Public Choice*, 43 (1):3–24, 1984.

A. Kleiner, B. Nebel, and V. A. Ziparo. A mechanism for dynamic ride sharing based on parallel auctions. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

V. Krishna and M. Perry. Efficient mechanism design. *Available at SSRN 64934*, 1998.

L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003.

V. Kuppuswamy and B. L. Bayus. Crowdfunding creative ideas: The dynamics of project backers in kickstarter. 2015.

V. Kuppuswamy and B. L. Bayus. Does my contribution to your crowdfunding project matter? *Journal of Business Venturing*, 32(1):72–89, 2017.

J.-J. Laffont and D. Martimort. Mechanism design with collusion and correlation. *Econometrica*, 68(2):309–342, 2000.

M. Lao. Resale price maintenance: The internet phenomenon and free rider issues. *The Antitrust Bulletin*, 55(2):473–512, 2010.

K. Larson and T. Sandholm. Costly valuation computation in auctions. In *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 169–182. Morgan Kaufmann Publishers Inc., 2001.

G.-G. Lee and H.-F. Lin. Customer perceptions of e-service quality in online shopping. *International Journal of Retail & Distribution Management*, 33(2):161–176, 2005.

J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, 1(1):5, 2007.

A. Levofsky and A. Greenberg. Organized dynamic ride sharing: The potential environmental benefits and the opportunity for advancing the concept. In *Transportation Research Board 2001 Annual Meeting*, 2001.

J. Lorenz, H. Rauhut, F. Schweitzer, and D. Helbing. How social influence can undermine the wisdom of crowd effect. *Proc. Natl. Acad. Sci. USA*, 108(22):9020–9025, 2011.

A. Marelli and A. Ordanini. What makes crowdfunding projects successful before and during the campaign? In *Crowdfunding in europe*, pages 175–192. Springer, 2016.

N. Marwell. Competing fundraising models in crowdfunding markets. *Available at SSRN 2777020*, 2015.

W. J. Mitchell. *Reinventing the automobile: Personal urban mobility for the 21st century*. MIT Press, 2010.

E. Mollick. The dynamics of crowdfunding: An exploratory study. *Journal of Business Venturing*, 29(1):1–16, 2014.

R. B. Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

R. B. Myerson. Optimal coordination mechanisms in generalized principal–agent problems. *Journal of mathematical economics*, 10(1):67–81, 1982.

R. B. Myerson and M. A. Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of economic theory*, 29(2):265–281, 1983.

M. Naor, B. Pinkas, and R. Sumner. Privacy preserving auctions and mechanism design. In *Proceedings of the 1st ACM Conference on Electronic Commerce*, pages 129–139. ACM, 1999.

V. Naroditskiy, N. R. Jennings, P. Van Hentenryck, and M. Cebrian. Crowdsourcing contest dilemma. *J R Soc Interface*, 11(99):20140532, 2014a.

V. Naroditskiy, S. Stein, M. Tonin, L. Tran-Thanh, M. Vlassopoulos, and N. R. Jennings. Referral incentives in crowdfunding. In *HCOMP 2014*, pages 171–183, 2014b.

S. Nath, P. Dayama, D. Garg, Y. Narahari, and J. Y. Zou. Threats and trade-offs in resource critical crowdsourcing tasks over networks. In *AAAI*, 2012.

N. Nisan and A. Ronen. Algorithmic mechanism design. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 129–140. ACM, 1999.

N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic behavior*, 35 (1-2):166–196, 2001.

N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic game theory*. Cambridge University Press, 2007.

M. I. Ohannessian, M. Roozbehani, D. Materassi, and M. A. Dahleh. Dynamic estimation of the price-response of deadline-constrained electric loads under threshold policies. In *2014 American Control Conference*, pages 2798–2803. IEEE, 2014.

D. C. Parkes and S. P. Singh. An mdp-based approach to online mechanism design. In *Advances in Neural Information Processing Systems*, pages 791–798, 2004.

S. G. Pauker and J. P. Kassirer. The threshold approach to clinical decision making. *New England Journal of Medicine*, 302(20):1109–1117, 1980.

G. Pickard, W. Pan, I. Rahwan, M. Cebrian, R. Crane, A. Madan, and A. Pentland. Time-critical social mobilization. *Science*, 334(6055):509–512, 2011.

P. Poudenx. The effect of transportation policies on energy consumption and greenhouse gas emission from urban passenger transportation. *Transportation Research Part A: Policy and Practice*, 42(6):901–909, 2008.

I. Rahwan, S. Dsouza, A. Rutherford, V. Naroditskiy, J. McInerney, M. Venanzi, N. R. Jennings, and M. Cebrian. Global manhunt pushes the limits of social mobilization. *Computer*, 46(4):68–75, 2013.

L. Ramaswamy and L. Liu. Free riding: A new challenge to peer-to-peer file sharing systems. In *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*, pages 10–pp. IEEE, 2003.

L. Rayo and I. Segal. Optimal information disclosure. *Journal of Political Economy*, 118(5): 949–987, 2010.

S. M. Ross. *Stochastic processes*, volume 2. John Wiley & Sons New York, 1996.

A. E. Roth, T. Sönmez, and M. U. Ünver. Kidney exchange. *The Quarterly Journal of Economics*, 119(2):457–488, 2004.

C. F. Runge. Institutions and the free rider: The assurance problem in collective action. *The Journal of Politics*, 46(1):154–181, 1984.

A. Rutherford, M. Cebrian, I. Rahwan, S. Dsouza, J. McInerney, V. Naroditskiy, M. Venanzi, N. R. Jennings, E. Wahlstedt, S. U. Miller, et al. Targeted social mobilization in a global manhunt. *PLoS ONE*, 8(9):e74628, 2013.

P. Samadi, A. H. M. Rad, R. Schober, and V. W. Wong. Advanced demand side management for the future smart grid using mechanism design. *IEEE Trans. Smart Grid*, 3(3):1170–1180, 2012.

P. Schmitt, B. Skiera, and C. Van den Bulte. Referral programs and customer value. *Journal of marketing*, 75(1):46–59, 2011.

W. Shen and C. Lopes. Managing autonomous mobility on demand systems for better passenger experience. In *Proceedings of the 18th Conference on Principles and Practice of Multi-Agent Systems*, pages 20 –35. Springer, 2015.

W. Shen, V. Babushkin, Z. Aung, and W. L. Woon. An ensemble model for day-ahead electricity demand time series forecasting. In *Proceedings of the fourth international conference on Future energy systems*, pages 51–62. ACM, 2013.

W. Shen, C. V. Lopes, and J. W. Crandall. An online mechanism for ridesharing in autonomous mobility-on-demand systems. In *IJCAI 2016*, pages 475–481, 2016.

W. Shen, A. Al Khemeiri, A. Almehrzi, W. Al Enezi, I. Rahwan, and J. W. Crandall. Regulating highly automated robot ecologies: Insights from three user studies. In *Proceedings of the 5th International Conference on Human Agent Interaction*, pages 111–120. ACM, 2017.

W. Shen, R. Achar, and C. V. Lopes. Toward understanding the impact of user participation in autonomous ridesharing systems. In *2018 Winter Simulation Conference (WSC)*, pages 845–856. IEEE, 2018a.

W. Shen, J. W. Crandall, K. Yan, and C. V. Lopes. Information design in crowdfunding under thresholding policies. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 632–640. International Foundation for Autonomous Agents and Multiagent Systems, 2018b.

W. Shen, Y. Feng, and C. V. Lopes. Multi-winner contests for strategic diffusion in social networks. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI-19)*. AAAI Press, 2019.

R. M. Sheremeta. Contest design: An experimental investigation. *Economic Inquiry*, 49(2): 573–590, 2011.

J. C. Short, D. J. Ketchen, A. F. McKenny, T. H. Allison, and R. D. Ireland. Research on crowdfunding: Reviewing the (very recent) past and celebrating the present. *Entrepreneurship Theory and Practice*, 41(2):149–160, 2017.

S. Skaperdas. Contest success functions. *Economic theory*, 7(2):283–290, 1996.

V. Skirnevskiy, D. Bendig, and M. Brettel. The influence of internal social capital on serial creators' success in crowdfunding. *Entrepreneurship Theory and Practice*, 41(2):209–236, 2017.

J. Solomon, W. Ma, and R. Wash. Don't wait!: How timing affects coordination of crowdfunding donations. In *Proceedings of the 18th ACM Conference on CSCW*, pages 547–556. ACM, 2015.

K. Spieser, K. Treleaven, R. Zhang, E. Frazzoli, D. Morton, and M. Pavone. Toward a systematic approach to the design and evaluation of automated mobility-on-demand systems: A case study in singapore. In *Road Vehicle Automation*, pages 229–245. Springer, 2014.

R. Strausz. A theory of crowdfunding-a mechanism design approach with demand uncertainty and moral hazard. *CEPR Discussion Paper No. DP11222*, 2016.

R. Strausz. A theory of crowdfunding: A mechanism design approach with demand uncertainty and moral hazard. *American Economic Review*, 107(6):1430–76, 2017.

X. Su. Intertemporal pricing with strategic customer behavior. *Management Science*, 53(5): 726–741, 2007.

A. Tabarrok. The private provision of public goods via dominant assurance contracts. *Public Choice*, 96(3-4):345–362, 1998.

I. A. Taneva. Information design. *University of Edinburgh*, 2015.

J. C. Tang, M. Cebrian, N. A. Giacobe, H.-W. Kim, T. Kim, and D. B. Wickert. Reflecting on the darpa red balloon challenge. *Communications of the ACM*, 54(4):78–85, 2011.

M. T. P. M. B. Tiago and J. M. C. Veríssimo. Digital marketing and social media: Why bother? *Business Horizons*, 57(6):703–708, 2014.

T. Todo, A. Iwasaki, and M. Yokoo. False-name-proof mechanism design without money. In *AAMAS 2011*, pages 651–658, 2011.

H. R. Varian and C. Harris. The vcg auction in theory and practice. *American Economic Review*, 104(5):442–45, 2014.

X. Wang, A. Montgomery, and K. Srinivasan. When auction meets fixed price: A theoretical and empirical examination of buy-it-now auctions. *Quantitative Marketing and Economics*, 6(4):339, 2008.

P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of applied probability*, pages 287–298, 1988.

Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai. Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1): 2, 2014.

H. Youn, M. T. Gastner, and H. Jeong. Price of anarchy in transportation networks: efficiency and optimality control. *Physical Review Letters*, 101(12):128701, 2008.

S. Yu, S. Johnson, C. Lai, A. Cricelli, and L. Fleming. Crowdfunding and regional entrepreneurial investment: an application of the crowdberkeley database. *Research Policy*, 46(10):1723–1737, 2017.

D. Zhao, S. D. Ramchurn, and N. R. Jennings. Incentive design for ridesharing with uncertainty. *arXiv preprint arXiv:1505.01617*, 2015.

D. Zhao, B. Li, J. Xu, D. Hao, and N. R. Jennings. Selling multiple items via social networks. In *AAMAS 2018*, pages 68–76, 2018.

Y.-P. Zhou, M. Fan, and M. Cho. On the threshold purchasing behavior of customers facing dynamically priced perishable products. *University of Washington*, 2005.

Appendix A

Derivations and Proofs for Chapter 4

A.1 Proofs

Proof of Proposition 1

By condition, $|s(k_1)| = |s(k_2)|$. That means the revenue does not increase between time k_1 to time k_2. Since $k_1 < k_2$, we have $T - k_2 < T - k_1$, which means less time is left to achieve the fundraising goal G. Thus, the campaign's PoS decreases or stays the same, regardless of backers' arrivals from time k_1 to k_2. That is, $r(t, (s(k_1), t)) \geq r(t, (s(k_2), t))$. By Equation 4.1, the utility of all the subsequently arriving backers weakly decreases, i.e, $\forall t \geq k_2, \forall i \in \mathcal{I}(t) : u_i((s(k_1), t)) \geq u_i((s(k_2), t))$. By Definition 4.2, we have $(s(k_1), t) \succsim (s(k_2), t)$ for all $t \geq k_2$, and for all $i \in \mathcal{I}(t)$.

Proof of Proposition 2

By assumption $\forall t \in \mathcal{T}, b(t) \in \{0, 1\}$, we have that the number of arrivals from time k_1 to k_2 is: $\sum_{k_1}^{k_2} b(t') \leq k_2 - k_1$. Since $|s(k_2)| - |s(k_1)| \geq (k_2 - k_1) \cdot P/G$, we have that at least $(k_2 - k_1)$ backers have contributed from time k_1 to k_2. Without loss of generality, one can assume that each time from k_1 to k_2, at least one backer arrives at the campaign and makes a contribution. In other words, the revenue of the campaign grows faster than or equal to the arrival of backers between k_1 and k_2. Thus, the campaign's PoS increases or stays the same. That is, $r(t,(s(k_2),t)) \geq r(t,(s(k_1),t))$. By Equation 4.1, the utility of all the subsequently arriving backers weakly increases, i.e., $\forall t \geq k_2, \forall i \in \mathcal{I}(t) : u_i(s(k_2),t_2) \geq u_i((s(k_1),t_1))$. By Definition 4.2, we have $\forall t \geq k_2, \forall i \in \mathcal{I}(t) : \varepsilon_2 \succsim \varepsilon_1$.

Proof of Lemma 1

By condition, we have $\varepsilon_2 \succsim \varepsilon_1$. By relation of preferences and utility [Chambers and Echenique, 2016], $\varepsilon_2 \succsim \varepsilon_1 \iff u_i(t,\varepsilon_2) \geq u_i(t,\varepsilon_1)$. According to backers' utility function (Equation 4.1), $r_i(t,\varepsilon_2) \geq r_i(t,\varepsilon_1)$. Depending on the order of the backer i's threshold ϕ_i, his belief $r_i(t,\varepsilon_1)$ when given report ε_1 and the belief $r_i(t,\varepsilon_2)$ given report ε_2, there are the following three cases.

- $r_i(t,\varepsilon_1) \leq r_i(t,\varepsilon_2) < \phi_i$: in this case, backer i will not contribute to the campaign given either ε_2 or $(\varepsilon_1,\varepsilon_2)$. That is, $E(\alpha_i = 1|\varepsilon_2) = E(\alpha_i = 1|(\varepsilon_1,\varepsilon_2)) = 0$.
- $r_i(t,\varepsilon_1) \leq \phi_i \leq r_i(t,\varepsilon_2)$: given report ε_2, backer i will contribute to the campaign and leave the system. An additional signal about the project status cannot improve his possibility of pledging, i.e, $E(\alpha_i = 1|\varepsilon_2) \geq E(\alpha_i = 1|(\varepsilon_1,\varepsilon_2))$.
- $\phi_i < r_i(t,\varepsilon_1) \leq r_i(t,\varepsilon_2)$: under this condition, backer i will contribute to the campaign and leave the system given either of the two reports. That is, $E(\alpha_i = 1|\varepsilon_2) = E(\alpha_i = 1|(\varepsilon_1,\varepsilon_2)) = 1$.

Proof of Lemma 2

We prove it by contradiction. Assume, to the contrary, that $\exists t \geq \max\{k_1,k_2\}$, $j \in \mathcal{I}(t)$:

$r_j(t,(\varepsilon_1,\varepsilon_2)) > \max\{r_j(t,\varepsilon_1), r_j(t,\varepsilon_2)\}$. Depending on the order of backer j's threshold ϕ_j, $r_j(t,(\varepsilon_1,\varepsilon_2))$, and $\max\{r_j(t,\varepsilon_1), r_j(t,\varepsilon_2)\}$, we have the following three cases:

- $r_j(t,(\varepsilon_1,\varepsilon_2)) > \max\{r_j(t,\varepsilon_1), r_j(t,\varepsilon_2)\} \geq \phi_j$: if $r_j(t,\varepsilon_1) > r_j(t,\varepsilon_2)$, we have

$$r_j(t,(\varepsilon_1,\varepsilon_2)) > r_j(t,\varepsilon_1) \geq \phi_j \; . \tag{A.1}$$

  By utility function (Equation 4.1),

$$u_j((\varepsilon_1,\varepsilon_2)) = u_j(\varepsilon_1) = c_j \cdot \alpha_j(t) \; . \tag{A.2}$$

  By the relation of preferences and utility [Chambers and Echenique, 2016], $(\varepsilon_1,\varepsilon_2) \sim \varepsilon_1$, where $\sim$ denotes indifferent to. This contradicts that

$$r_j(t,(\varepsilon_1,\varepsilon_2)) > r_j(t,\varepsilon_1) \; . \tag{A.3}$$

  A similar contradiction occurs if $r_j(t,\varepsilon_2) > r_j(t,\varepsilon_1)$.

- $\phi_j > r_j(t,(\varepsilon_1,\varepsilon_2)) > \max\{r_j(t,\varepsilon_1), r_j(t,\varepsilon_2)\}$: in this case, by Equation 4.1, $u_j((\varepsilon_1,\varepsilon_2)) = u_j(\varepsilon_1) = u_j(\varepsilon_2) = 0$. By the relation of preferences and utility [Chambers and Echenique, 2016], $(\varepsilon_1,\varepsilon_2) \sim \varepsilon_1 \sim \varepsilon_2$. This contradicts that

$$r_j(t,(\varepsilon_1,\varepsilon_2)) > \max\{r_j(t,\varepsilon_1), r_j(t,\varepsilon_2)\} \tag{A.4}$$

- $r_j(t,(\varepsilon_1,\varepsilon_2)) \geq \phi_j > \max\{r_j(t,\varepsilon_1), r_j(t,\varepsilon_2)\}$: by Equation 4.1, we have $u_j((\varepsilon_1,\varepsilon_2)) = c_j \cdot \alpha_j(t)$ and $u_j(\varepsilon_1) = u_j(\varepsilon_2) = 0$. By the relation of preferences and utility [Chambers and Echenique, 2016], $(\varepsilon_1,\varepsilon_2) \succ \varepsilon_1, (\varepsilon_1,\varepsilon_2) \succ \varepsilon_2$ and $\varepsilon_1 \sim \varepsilon_2$, where $\succ$ denotes strictly preferred to. If $\varepsilon_1 \sim \varepsilon_2$, then $(\varepsilon_1,\varepsilon_2) \sim \varepsilon_1 \sim \varepsilon_2$. This contradicts that $(\varepsilon_1,\varepsilon_2) \succ \varepsilon_1, (\varepsilon_1,\varepsilon_2) \succ \varepsilon_2$.

Proof of Theorem 2

If the information is vertical, we have $E(\alpha_i^{'} = 1) \geq E(\alpha_i^{''} = 1)$ by Lemma 4.1. If it is horizontal, $r_i(t, (\varepsilon_1, \varepsilon_2)) \leq \max\{r_i(t, (\varepsilon_1)), r_i(t, (\varepsilon_2))\}$ by Lemma 4.2. By Equation 4.1, there are three cases:

- $\phi_i \leq r_i(t, (\varepsilon_1, \varepsilon_2)) \leq \max\{r_i(t, \varepsilon_1), r_i(t, \varepsilon_2)\}$: in this case, backer i will contribute for both conditions, $E(\alpha_i^{'} = 1) = E(\alpha_i^{''} = 1) = 1$.
- $r_i(t, (\varepsilon_1, \varepsilon_2)) \leq \max\{r_i(t, \varepsilon_1), r_i(t, \varepsilon_2)\} < \phi_i$: in this case, backer i will not contribute for both conditions, $E(\alpha_i^{'} = 1) = E(\alpha_i^{''} = 1) = 0$.
- $r_i(t, (\varepsilon_1, \varepsilon_2)) < \phi_i \leq \max\{r_i(t, \varepsilon_1), r_i(t, \varepsilon_2)\}$: in this case, backer i will not contribute if given $(\varepsilon_1, \varepsilon_2)$, i.e., $E(\alpha_i^{''} = 1) = 0$. If given either ε_1 or ε_2, backer i will either contribute or not contribute, i.e., $E(\alpha_i^{'} = 1) \geq 0$.

Proof of Lemma 3

$\Longleftarrow$ If the right side holds for all $t' \leq t, i \in \mathcal{I}^{t'}$, then $\forall t' \leq t$: $|s(t')| - |s(t'-1)| \geq P/G = (t' - (t'-1)) \cdot P/G$. According to Proposition 4.2, $\forall i \in I(t) : (s(t'), t) \succsim (s(t'-1), t)$, where $t \geq t'$. That is, immediate disclosure is always preferred by all the backers in the campaign.

$\Longrightarrow$ We prove it by contradiction. Assume, to the contrary, that $|s(t')| - |s(t'-1)| < P/G$ such that $DP_{im}(t)$ is optimal for some i, t'. Without loss of generality, we let $|s(t')| = |s(t'-1)| + |\Delta s|$, where $|\Delta s| < P/G$. Since $|s(t')| = |s(t'-1)| + \sum_{j \in \mathcal{I}(t')} \alpha_j(t') \cdot P/G$, where $\alpha_i \in \{0, 1\}$, we have $|\Delta s| = \sum_{j \in \mathcal{I}(t')} \alpha_j(t') \cdot P/G < P/G$. Therefore, $|\Delta s| = 0$, which indicates that $|s(t')| = |s(t'-1)|$. According to Proposition 4.1, $\forall t \geq t' : (s(t'-1), t) \succsim (s(t'), t)$, which contradicts the supposition that DP_{im} is optimal.

Proof of Lemma 4

We prove it by contradiction. Assume, to the contrary, that: DP_{im} is not optimal for some

$d(i, t') = (s(t'), t')$. This indicates that at time $t' \geq t$, the entrepreneur could possibly profit by delaying information disclosure. Without loss of generality, we assume that $d(i, t')_{opt} = (s(k'), t')$ is the disclosure strategy used in the optimal disclosure policy for backer $i \in \mathcal{I}(t')$, where $k' < t'$. Depending on the relation between t and k', there are two cases (since $t' \geq t$) : (i) $t \leq k' < t'$; (ii) $k' < t \leq t'$.

- $t \leq k' < t'$: at time t', the project has already succeeded, which means there is no uncertainty for backer i since the campaign's PoS is 1. Therefore, delaying disclosure does not increase backer i's estimate on PoS, which contradicts the proposition that $d(i, t') = (s(t'), t')$ is not optimal.

- $k' < t \leq t'$: at time t', backer i's estimate $r_i(t', d(i, t')_{opt}) \leq 1$, while $r_i(t', d(i, t')) = 1$. That is, $d(i, t') \succsim d(i, t')_{opt}$ $(\forall i \in \mathcal{I}(t'))$, which contradicts the proposition that DP_{im} is not optimal.

A.2 Expert Algorithms

Random Selection

The random algorithm picks the project status $s(k_{sel})$ at random from $s(k) \in H_i(t)$ with equal probability. The expected increase of revenue $\Upsilon(s(k), i, t)$ is computed by averaging the revenue received when the entrepreneur disclosed the project status $s(k)$.

Greedy Selection

The greedy selection algorithm chooses the project status $s(k_{sel})$ based on the empirical responses that the entrepreneur has received from the backers, using a one-step-look-ahead approach.

At time t, given the disclosure strategy $d(i,t) = (s(k),t)$, the entrepreneur establishes a historical belief $\Upsilon_{old}(s(k),i,t)$ of the expected increase in the revenue, where $i \in \mathcal{I}(t)$, and $s(k) \in H_i(t)$: $\Upsilon_{old}(s(k),i,t) = \frac{1}{n_k(t)} \cdot \sum_{t'=1}^{t-1} \sum_{j \in \mathcal{I}(t')} \frac{\alpha_j(t')}{\eta(t')/\eta(t)}$, where $n_k(t)$ denotes the times that $s(k)$ has been revealed to backers up to time t. $\alpha_j(t')$ is backer j's action given disclosure strategy $(s(k),t')$. $\eta(t') = |s(t')|/t'$, and $\eta(t) = |s(t)|/t$ are the revenue growth rates up to time t' and t, respectively. Note that $\Upsilon_{old} = 0$ if $n_k(t) = 0$ or $t = 1$.

The historical belief represents the entrepreneur's estimate of the average revenue she receives by revealing project status $s(k)$ to the backers, with discounting of the revenue growth rates. It is a rough estimate of revenue increase that $d(i,j) = (s(k),t)$ brings.

At time t, the entrepreneur's temporal belief Υ_{tmp} of the expected revenue increase given the disclosure decision $d(i,t) = (s(k),t)$, is determined as follows:

$$\Upsilon_{tmp}(s(k),i,t) = \sum_{j \in \mathcal{I}(t-1)} \frac{\alpha_j(t-1)}{|\mathcal{I}(t-1)|} , \tag{A.5}$$

where $\mathcal{I}(t-1)$ denotes the set of backers in the campaign at time $t-1$ and $\alpha_j(t-1)$ is backer j's action at time $t-1$. If $|\mathcal{I}(t-1)| = 0$ or $t = 1$, then $\Upsilon_{tmp} = 0$. The temporal belief captures the latest decisions of the backers that will most probably stay in the campaign at time t.

The entrepreneur estimates the belief of the expected increase of revenue, given $d(i,t) = (s(k),t)$ for backer $i \in \mathcal{I}(t)$, with the following equation:

$$\Upsilon(s(k),i,t) = (1-\lambda)\Upsilon_{old}(s(k),i,t) + \lambda\Upsilon_{tmp}(s(k),i,t) , \tag{A.6}$$

where $\lambda \in [0,1]$ is the learning rate (we use $\lambda = 0.1$).

The greedy algorithm then selects the project status $s(k_{sel})$ by using the following equation:

$$s(k_{sel}) = \underset{s(k)\in H}{\operatorname{argmax}}\, \Upsilon(s(k), i, t)\ . \tag{A.7}$$

The probability for selects the project status $s(k_{sel})$ is 1 and 0 for others.

ϵ-Greedy Exploration

In ϵ-greedy exploration, with probability ϵ the algorithm selects a random choice $s(k)$. Otherwise, with probability $1-\epsilon$ it selects the greedy choice use the greedy selection. That is,

$$Pr(s(k)) = \begin{cases} 1-\epsilon+\frac{\epsilon}{n_k(t)}\ , & \text{greedy selection} \\ \frac{\epsilon}{n_k(t)}\ , & \text{random selection} \end{cases} \tag{A.8}$$

where $n_k(t)$ is the times that $s(k)$ has been reveled to backers up to time t, $\epsilon = c/n_k(t)$, and $c \in [0,1]$ is a constant. Note that $\epsilon = c$ if $n_k(t) = 0$.

Softmax Exploration

Softmax selects the choice using a Boltzmann distribution [Ross, 1996]. At time t, the algorithm selects choice $s(k)$ with the probability:

$$Prob(s(k)) = \frac{e^{\Upsilon(s(k),i,t)/\tau}}{\sum_{s(k)\in H_i(t)} e^{\Upsilon(s(k),i,t)/\tau}}\ , \tag{A.9}$$

where $\tau = \max\{\mu, C^t/\log n_k(t)\}$ is the temperature parameter. Here, $\mu = 0.0001$, $\tau = 1$ when $n_k(t) = 0$, and C^t is determined by:

$$C^t = \max_{s(k_1),s(k_2)\in H_i(t)} |\Upsilon(s(k_1),i,t) - \Upsilon(s(k_2),i,t)|\ . \tag{A.10}$$

The softmax exploration selects each choice with a probability that is proportional to the average Υ.

Appendix B

Derivations and Proofs for Chapter 6

B.1 Proofs

Proof of Theorem 1

Proof. By Definition 6.2, the task rewards for player v remain unchanged even if he creates false identities. To prove $\pi(v) \geq \sum_{r\in R}\pi(r)$, it suffices that if the v's diffusion rewards are greater than or equal to the sum of diffusion reward of her false identities. That is, $\pi_d(v) \geq \sum_{r\in R}\pi_d(r)$. To prove this condition holds, we first show that the virtual credits earned by v are strictly greater than the sum of the virtual credits earned by the replicas R. That is, $b_v > \sum_{r\in R} b_r$. There are three scenarios according the types of false-name attacks: type 1 attacks, type 2 attacks, and hybrid attacks. We prove each case in turn:

- Type 1 attacks: By Equation 6.3, the virtual credits of original node v: $b_v = \eta \cdot (t_v)^2 + t_v \cdot \sum_{u\in\kappa_v^+}\sum_{p\in P_{vu}} t_u \cdot \omega(p) \cdot \lambda^{|p|}$. For simplicity, we let $b(v) = \eta \cdot (t_v)^2 + t_v \cdot C(v)$.

Note that node v shares the same successors with replica r_m. Therefore, we have the virtual credits for replica r_m: $b_{r_m} = \eta \cdot (t_{r_m})^2 + t_{r_m} \cdot \sum_{u \in \kappa_r^+} \sum_{p \in P_{ru}} t_u \cdot \omega(p) \cdot \lambda^{|p|} = \eta \cdot (t_{r_m})^2 + t_{r_m} \cdot \sum_{u \in \kappa_v^+} \sum_{p \in P_{vu}} t_u \cdot \omega(p) \cdot \lambda^{|p|} = \eta \cdot (t_{r_m})^2 + t_{r_m} \cdot C(v)$.
Similarly, we have r_{m-1}: $b_{r_{m-1}} = \eta \cdot (t_{r_{m-1}})^2 + t_{r_{m-1}} \sum_{u \in \kappa_{r_{m-1}}^+} \sum_{p \in P_{r_{m-1}u}} t_u \cdot \omega(p) \cdot \lambda^{|p|}$ $= \eta \cdot (t_{r_{m-1}})^2 + t_{r_{m-1}} \cdot \sum_{u \in \kappa_{r_{m-1}}^+} \sum_{p \in P_{r_{m-1}u}} t_u \cdot \omega(p) \cdot \lambda^{|p|} = \eta \cdot (t_{r_{m-1}})^2 + \lambda t_{r_{m-1}} \cdot \sum_{u \in \kappa_{r_m}^+} \sum_{p \in P_{r_m u}} t_u \cdot \omega(p) \cdot \lambda^{|p|} + \lambda t_{r_{m-1}} \cdot t_{r_m} \eta \cdot (t_{r_{m-1}})^2 + \lambda t_{r_{m-1}} \cdot C(v) + \lambda \cdot t_{r_{m-1}} \cdot t_{r_m}$.

Likewise, we have: $b_{r_{m-2}} = \eta \cdot (t_{r_{m-2}})^2 + \lambda^2 t_{r_{m-2}} \cdot C(v) + \lambda^2 t_{r_{m-2}} \cdot t_{r_m} + \lambda t_{r_{m-1}} t_{r_{m-2}}$, ..., $b_{v_1'} = \eta \cdot (t_{r_1})^2 + \lambda^{m-1} t_{r_1} \cdot C(v) + \lambda t_{r_1} \cdot t_{r_2} + \lambda^2 \cdot t_{r_1} \cdot t_{r_3} + ... + \lambda^{m-1} t_{r_1} \cdot t_{r_m}$.

Now we have the sum of the virtual credits for all replicas:

$$
\begin{aligned}
\sum_{r \in R} b_r &= b_{r_1} + ... + b_{r_{m-1}} + b_{b_m} \\
&= \eta \cdot [(t_{r_1})^2 + ... + (t_{r_{m-1}})^2 + (t_{r_m})^2] \\
&[\lambda^{m-1} t_{r_1} + \lambda^{m-2} t_{r_2} + ... + \lambda t_{r_{m-1}} + t_{r_m}] + t_{r_m} \cdot \\
&C(v) + \lambda t_{r_1} \cdot t_{r_2} + \lambda^2 \cdot t_{r_1} \cdot t_{r_3} + ... + \lambda t_{r_{m-1}} \cdot t_{r_m} \\
&< \eta \cdot [(t_{r_1})^2 + ... + (t_{r_{m-1}})^2 + (t_{r_m})^2] + [t_{r_1} + \\
&t_{r_2} + ... + t_{r_m}] \cdot C(v) + t_{r_1} \cdot t_{r_2} + ... + t_{r_{m-1}} \cdot t_{r_m} \\
&= \eta [t_{r_1} + t_{r_2} + ... + t_{r_m}]^2 + t_v \cdot C(v) + \\
&(\lambda - 2\eta)[t_{r_1} \cdot t_{r_2} + ... + t_{r_{m-1}} \cdot t_{r_m}] \\
&\leq \eta (t_v)^2 + t_v \cdot C(v) \\
&= b(v) \ .
\end{aligned}
$$

The first inequality holds since $0 < \lambda < 1$. The second inequality holds because $\lambda \geq \eta/2$. Thus, $b_v > \sum_{r \in S} b_r$.

- Type 2 attacks: Note that replicas $r_1, ..., r_m$ share the same successors with node v:

$$\begin{aligned} b_{r_j} &= \eta \cdot (t_{r_j})^2 + t_{r_j} \cdot \sum_{u \in \kappa_{r_j}^+} \sum_{p \in P_{r_j u}} t_u \cdot \omega(p) \cdot \lambda^{|u|} \\ &= \eta \cdot (t_{r_j})^2 + t_{r_j} \cdot \sum_{u \in \kappa_v^+} \sum_{p \in P_{vu}} t_u \cdot \omega(p) \cdot \lambda^{|p|} \\ &= \eta \cdot (t_{r_j})^2 + t_{r_j} \cdot C(v) \ . \end{aligned}$$

Thus, we have:

$$\begin{aligned} \sum_{r \in R} b_r = \eta \cdot [(t_{r_1})^2 + ... + (t_{r_{m-1}})^2 + (t_{r_m})^2] + \\ [t_{r_1} + t_{r_2} + ... + t_{r_m}] \cdot C(v) \\ < \eta \cdot [t_{r_1} + t_{r_2} + ... + t_{r_m}]^2 + t_v \cdot C(v) \\ = b_v \ . \end{aligned}$$

The inequality is by Multinomial Theorem. Thus, $b_v > \sum_{r \in R} b_r$.

- Hybrid attacks: Since hybrid attacks are combinations of type 1 attacks, and type 2 attacks, it is trivial to know that under hybrid attacks $b_v > \sum_{r \in R} b_r$.

To prove $\pi_d(v) \geq \sum_{r \in R} \pi_d(r)$, it suffices if the following inequality holds:

$$\frac{(b_v)^\sigma}{\sum_{u \in V_{G_v}} (b_u)^\sigma} > \sum_{r \in R} \frac{(b_r)^\sigma}{\sum_{u \in V_{G_r}} (b_u)^\sigma} \ . \tag{B.1}$$

Note that $G_r = G_v \setminus \{v\} \cup R$. Let $x = (b_v)^\sigma$, $y = \sum_{u \in V_{G_v}} (b_u)^\sigma$, $z = \sum_{r \in R} (b_u)^\sigma$, and $q = \sum_{u \in V_{G_r}} (b_u)^\sigma$, we have $0 < \frac{x}{y} < 1$, $x > z > 0$, and $y > q > 0$. Equation B.1 is equivalent to: $\frac{x}{y} > \frac{z}{q}$. Since $0 < \frac{x}{y} < 1$, we have:

$$\frac{z}{q} / \frac{x}{y} = \frac{x-d}{y-d} / \frac{x}{y} = \frac{(x-d)y}{x(y-d)} = \frac{1-d/x}{1-d/y} < 1 \, . \tag{B.2}$$

Thus, $\frac{x}{y} > \frac{z}{q}$. Therefore, $\pi_d(v) \geq \sum_{r \in RS} \pi_d(r)$. Now it follows that the reward mechanism π is false-name proof. □

Proof of Theorem 2

Proof. By Equations 6.2 and 6.5 , we have player v's expected utility: $U(v) = \mu \cdot t_v + \pi_d(v) - \delta_v \cdot t_v = (\mu - \delta_v) \cdot t_v + \pi_d(v)$. Since the parameter μ is known to the player in advance, $\mu - \delta_v \geq 0$. Since $\pi_d(v) > 0$ if player has exerted diffusion contributions, we have $U(v) > 0$ for all players that have exerted both task efforts and diffusion contributions. □

Proof of Theorem 3

Proof. By Equation 6.5, $\sum_{v \in G} \pi(v) = \sum_{v \in G}(\mu \cdot t_v + \pi_d(v))$. By Equation 6.4, $\sum_{v \in G} \pi_d(v) \leq \phi \sum_{v \in G} t_v$. Thus, we have $\sum_{v \in G} \pi(v) \leq (\mu + \phi) \sum_{v \in G}$. Let $\vartheta = \mu + \phi$, we have $\sum_{v \in G} \pi(v) \leq \vartheta \sum_{v \in G}$. □

Proof of Theorem 4

Proof. Consider that the virtual credits of v_1 is $b_{v_1} = B$ when no successor is added. Now adding a direct successor v_i to v_2, v_1's virtual credits $b_{v_1'} = B + \sum_{p \in P_{v_1 vi}} \lambda^{|p|} \cdot t_{v_1} \cdot t_{v_i} \cdot \omega(p)$. Similarly, if adding a direct successor of a successor of v_2, we have v_1's virtual credits: $b_{v_1''} = B + \sum_{p \in P_{v_1 vj}} \lambda^{|p|} \cdot t_{v_1} \cdot t_{v_j} \cdot \omega(p)$. Note that $dist(v_1, v_i) < dist(v_1, v_j)$, and $t_{v_i} = t_{v_j}$, it suffices to show that $b'_{v_1} > b''_{v_1}$. Following the same method shown in Equation B.2, we have the diffusion rewards: $\pi_d(v_1^{'}) \geq \pi_d(v_1^{''})$. □

Proof of Theorem 5

Proof. According to Equation 6.4, player v virtual credits and consequentially the diffusion rewards are determined by a contest among players in the graph rooted at v, G_v. It follows that the MWC mechanism satisfies the subgraph constraint. $\square$