

© 2017 by Haiming Jin.

INCENTIVE MECHANISM DESIGN FOR MOBILE CROWD SENSING SYSTEMS

BY

HAIMING JIN

DISSERTATION

Submitted in partial fulfillment of the requirements  
for the degree of Doctor of Philosophy in Computer Science  
in the Graduate College of the  
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Doctoral Committee:

Professor Klara Nahrstedt, Chair & Director of Research  
Professor Rayadurgam Srikant  
Professor Carl A. Gunter  
Assistant Professor Ruta Mehta  
Professor Baochun Li, University of Toronto

# Abstract

The recent proliferation of increasingly capable and affordable mobile devices with a plethora of on-board and portable sensors that pervade every corner of the world has given rise to the fast development and wide deployment of mobile crowd sensing (MCS) systems. Nowadays, applications of MCS systems have covered almost every aspect of people’s everyday living and working, such as ambient environment monitoring, healthcare, floor plan reconstruction, smart transportation, indoor localization, and many others.

Despite their tremendous benefits, MCS systems pose great new research challenges, of which, this thesis targets one important facet, that is, to effectively incentivize (crowd) workers to achieve maximum participation in MCS systems. Participating in crowd sensing tasks is usually a costly procedure for individual workers. On one hand, it consumes workers’ resources, such as computing power, battery, and so forth. On the other hand, a considerable portion of sensing tasks require the submission of workers’ sensitive and private information, which causes privacy leakage for participants. Clearly, the power of crowd sensing could not be fully unleashed, unless workers are properly incentivized to participate via satisfactory rewards that effectively compensate their participation costs.

Targeting the above challenge, in this thesis, I present a series of novel incentive mechanisms, which can be utilized to effectively incentivize worker participation in MCS systems. The proposed mechanisms not only incorporate workers’ quality of information in order to selectively recruit relatively more reliable workers for sensing, but also preserve workers’ privacy so as to prevent workers from being disincentivized by excessive privacy leakage. I demonstrate through rigorous theoretical analyses and extensive simulations that the proposed incentive mechanisms bear many desirable properties theoretically, and have great potential to be practically applied.

*To my parents and my older brother.*

# Acknowledgments

First and foremost, I would like to express my deepest gratitude to my thesis advisor, Professor Klara Nahrstedt, for her invaluable guidance, continuous support, and persistent encouragement. I learned tremendously from her in every aspect. Her passion in research, teaching, and student supervision has motivated me to pursue a career in academia. She will be my lifetime role model, and I wish I could become a great researcher as well as advisor like her. I feel so fortunate to have her as my advisor.

Next, I would like to sincerely thank my other committee members, Professor R. Srikant, Professor Carl A. Gunter, Professor Ruta Mehta, and Professor Baochun Li for their invaluable inputs. Their constructive suggestions and comments helped me significantly improve my thesis. I am truly honored to have them in my doctoral committee.

During my Ph.D. study, I have worked as the teaching assistant of the course, *Discrete Structures*, under the supervision of Professor Margaret M. Fleck. She was always patient and helpful whenever I had problems. She showed me what it takes to be an excellent teacher and educator. I believe what I learned from her will constantly benefit me in my career.

I would like to extend my gratitude to the professors, researchers, and colleagues who helped me on this thesis and other research projects. In particular, I would like to thank Professor Lu Su, Professor Jinhui Xu, Professor Julia Chuzhoy, Professor Nikita Borisov, Professor Suleyman Uludag, Professor King-Shan Lui, Dr. Bolin Ding, Houping Xiao, Danyang Chen, Tuo Yu, Tingyu Wang, He Huang, Tianyuan Liu, Zhuotao Liu, and Hongpeng Guo for valuable discussions, suggestions, and collaborations.

I want to take this opportunity to thank all of my friends who supported me during these years. It is my good fortune to have so many friends, they make my Ph.D. journey a pleasant and exciting one. However, it is impossible to mention all the names. Here I may only be able to list some of my

fellow colleagues in UIUC System and Networking group: Bo Chen, Phuong V. Nguyen, Wenyu Ren, Tarek Elgamal, Hongyang Li, Shannon Chen, Raoul Rivas, Ahsan Arefin, Debish Fesehaye, Shen Li, Shiguang Wang, Shaohan Hu, Shuochao Yao, Yiran Zhao, Huajie Shao, Yunlong Gao, Soteris Demetriou, Avesta Hojjati, Yunhui Long, Guliz Seray Tuncay, Aston Zhang, Muhammad Naveed, Qingxi Li, Mo Dong, Tong Meng, Wenxuan Zhou, Sangeetha Abdu Jyothi, Chris Cai, Sayed Hadi Hashemi, Faraz Faghri, Mohammad Ahmad, and Rui Yang.

Also, I would like to thank various funding agencies, such as National Science Foundation, Department of Energy, Boeing Corporation, Ralph M. and Catherine V. Fisher Grant, for their financial support and assistance.

Lastly, and most importantly, I wish to thank my father Guangchun Jin, my mother Zhenzi Jin, and older brother Hailong Jin, for their continuous love and support. They are with me all these years, sharing my pain and happiness. My degree is at the expense of their sacrifices. My family is my most prized possession. To them I dedicate this thesis.

# Table of Contents

<b>List of Tables</b> . . . . .	<b>ix</b>
<b>List of Figures</b> . . . . .	<b>x</b>
<b>Chapter 1 Introduction</b> . . . . .	<b>1</b>
1.1 Motivation and Challenges . . . . .	1
1.2 Thesis Statement . . . . .	3
1.3 Thesis Overview . . . . .	3
1.3.1 QoI Aware Incentive Mechanisms for MCS Systems . . . . .	3
1.3.2 Incentivizing Multi-Requester Mobile Crowd Sensing . . . . .	4
1.3.3 Bid Privacy-Preserving Incentive Mechanism for MCS Systems . . . . .	4
1.3.4 Incentivizing Privacy-Preserving Data Aggregation in MCS Systems . . . . .	5
1.4 Thesis Organization . . . . .	6
<b>Chapter 2 QoI Aware Incentive Mechanisms for MCS Systems</b> . . . . .	<b>7</b>
2.1 Introduction . . . . .	7
2.2 Preliminaries . . . . .	10
2.2.1 System Overview . . . . .	10
2.2.2 Auction Model . . . . .	11
2.2.3 Design Objective . . . . .	13
2.3 SRC Auction . . . . .	14
2.3.1 Mathematical Formulation . . . . .	14
2.3.2 Mechanism Design . . . . .	17
2.3.3 Walk-through Example . . . . .	20
2.3.4 Analysis . . . . .	21
2.4 MRC Auction . . . . .	25
2.4.1 Mathematical Formulation . . . . .	25
2.4.2 Mechanism Design . . . . .	26
2.4.3 Analysis . . . . .	28
2.5 Performance Evaluation . . . . .	30
2.5.1 Baseline Method . . . . .	30
2.5.2 Simulation Settings . . . . .	31
2.5.3 Simulation Results . . . . .	32
2.6 Related Work . . . . .	33
2.7 Conclusion . . . . .	34

<b>Chapter 3</b>	<b>Incentivizing Multi-Requester Mobile Crowd Sensing</b>	<b>36</b>
3.1	Introduction	36
3.2	Preliminaries	38
3.2.1	System Overview	38
3.2.2	Reliability Level Model	39
3.2.3	Auction Model	40
3.2.4	Design Objectives	41
3.3	Design Details	42
3.3.1	Data Aggregation Mechanism	43
3.3.1.1	Proposed Mechanism	43
3.3.1.2	Analysis	44
3.3.2	Incentive Mechanism	47
3.3.2.1	Mathematical Formulation	47
3.3.2.2	Hardness Proof	48
3.3.2.3	Proposed Mechanism	52
3.3.2.4	Analysis of the Proposed Mechanism	55
3.4	Performance Evaluation	57
3.4.1	Baseline Methods	57
3.4.2	Simulation Settings	58
3.4.3	Simulation Results	58
3.5	Related Work	59
3.6	Conclusion	60
<b>Chapter 4</b>	<b>Bid Privacy-Preserving Incentive Mechanism for MCS Systems</b>	<b>61</b>
4.1	Introduction	61
4.2	Preliminaries	63
4.2.1	System Overview	63
4.2.2	Aggregation Method	64
4.2.3	Auction Model	65
4.2.4	Design Objective	66
4.3	Mathematical Formulation	68
4.4	Mechanism Design	70
4.5	Analysis	73
4.6	Performance Evaluation	78
4.6.1	Baseline Method	78
4.6.2	Simulation Settings	79
4.6.3	Simulation Results	79
4.7	Related Work	82
4.8	Conclusion	82
<b>Chapter 5</b>	<b>Incentivize Privacy-Preserving Data Aggregation in MCS Systems</b>	<b>84</b>
5.1	Introduction	84
5.2	Preliminaries	86
5.2.1	System Overview	86
5.2.2	Task Model	88
5.2.3	Reliability Level Model	88
5.2.4	Auction Model	89

5.2.5	Design Objective . . . . .	91
5.3	Design Details . . . . .	92
5.3.1	Data Aggregation Mechanism . . . . .	92
5.3.1.1	Proposed Mechanism . . . . .	92
5.3.1.2	Analysis . . . . .	94
5.3.2	Incentive Mechanism . . . . .	98
5.3.2.1	Mathematical Formulation . . . . .	99
5.3.2.2	Proposed Mechanism . . . . .	101
5.3.2.3	Analysis . . . . .	103
5.3.3	Data Perturbation Mechanism . . . . .	107
5.3.3.1	Proposed Mechanism . . . . .	107
5.3.3.2	Analysis . . . . .	109
5.3.4	Summary of Design Details . . . . .	112
5.4	Performance Evaluation . . . . .	112
5.4.1	Baseline Methods . . . . .	112
5.4.2	Simulation Settings . . . . .	113
5.4.3	Simulation Results . . . . .	115
5.5	Related Work . . . . .	117
5.6	Conclusion . . . . .	118
<b>Chapter 6 Conclusions and Future Work . . . . .</b>		<b>119</b>
6.1	Summary . . . . .	119
6.1.1	QoI Aware Incentive Mechanisms for MCS Systems . . . . .	119
6.1.2	Incentivizing Multi-Requester Mobile Crowd Sensing . . . . .	120
6.1.3	Bid Privacy-Preserving Incentive Mechanism for MCS Systems . . . . .	120
6.1.4	Incentivizing Privacy-Preserving Data Aggregation in MCS Systems . . . . .	120
6.2	Future Research Directions . . . . .	121
<b>References . . . . .</b>		<b>123</b>

# List of Tables

2.1	Summary of design objectives . . . . .	14
2.2	Parameter setting of the example . . . . .	20
2.3	An example of the QoI-SRC auction's winner determination procedure . . . . .	21
2.4	An example of QoI-SRC auction's pricing procedure . . . . .	21
2.5	Simulation setting 2.I and 2.II . . . . .	31
2.6	Simulation setting 2.III and 2.IV . . . . .	31
2.7	Execution time (seconds) for setting 2.I . . . . .	32
2.8	Execution time (seconds) for setting 2.II . . . . .	33
3.1	Simulation setting 3.I and 3.II . . . . .	58
4.1	Simulation setting 4.I-4.IV . . . . .	79
4.2	Execution time (seconds) for setting 4.I and 4.II . . . . .	80
5.1	Simulation setting 5.I-5.IV (continuous tasks only) . . . . .	114
5.2	Simulation setting 5.V-5.VIII (categorical tasks only) . . . . .	114
5.3	Execution time (seconds) for setting 5.I and 5.II . . . . .	116
5.4	Execution time (seconds) for setting 5.V and 5.VI . . . . .	117

# List of Figures

2.1	An example of the MedWatcher MCS system . . . . .	8
2.2	Bidding graph of the example . . . . .	20
2.3	Social welfare (setting 2.I) . . . . .	32
2.4	Social welfare (setting 2.II) . . . . .	32
2.5	Social welfare with varying $\epsilon$ (setting 2.III) . . . . .	32
2.6	Social welfare with varying $\epsilon$ (setting 2.IV) . . . . .	32
2.7	Social welfare with varying $\beta$ (setting 2.III) . . . . .	32
2.8	Social welfare with varying $\beta$ (setting 2.IV) . . . . .	32
3.1	Framework of CENTURION . . . . .	38
3.2	Social welfare (setting 3.I) . . . . .	59
3.3	Social welfare (setting 3.II) . . . . .	59
3.4	MAE (setting 3.I) . . . . .	59
3.5	MAE (setting 3.II) . . . . .	59
3.6	Error probability (setting 3.I) . . . . .	59
3.7	Error probability (setting 3.II) . . . . .	59
4.1	Platform’s total payment (setting 4.I) . . . . .	80
4.2	Platform’s total payment (setting 4.II) . . . . .	80
4.3	Platform’s total payment (setting 4.III) . . . . .	81
4.4	Platform’s total payment (setting 4.IV) . . . . .	81
4.5	Trade-off between the platform’s total payment and privacy leakage . . . . .	81
5.1	Framework of INCEPTION . . . . .	87
5.2	Platform’s total payment (setting 5.I) . . . . .	115
5.3	Platform’s total payment (setting 5.II) . . . . .	115
5.4	Platform’s total payment (setting 5.III) . . . . .	115
5.5	Platform’s total payment (setting 5.IV) . . . . .	115
5.6	MAE of data aggregation (continuous tasks) . . . . .	115
5.7	EP after perturbation (continuous tasks) . . . . .	115
5.8	Platform’s total payment (setting 5.V) . . . . .	116
5.9	Platform’s total payment (setting 5.VI) . . . . .	116
5.10	Platform’s total payment (setting 5.VII) . . . . .	116
5.11	Platform’s total payment (setting 5.VIII) . . . . .	116
5.12	MAE of data aggregation (categorical tasks) . . . . .	116
5.13	EP after perturbation (categorical tasks) . . . . .	116

# Chapter 1

## Introduction

The recent proliferation of increasingly capable and affordable mobile devices (e.g., smartphones, smartglasses, smartwatches) with a plethora of on-board and portable sensors (e.g., accelerometer, compass, gyroscope, GPS, camera) has given rise to the emergence of various people-centric mobile crowd sensing (MCS) systems [3–5, 7–10, 16, 17, 21, 30, 36, 38, 46, 80, 93, 99, 108]. As opposed to traditional sensing paradigms which usually leverage professionals and dedicatedly deployed sensors, MCS systems have enabled unprecedentedly fast and easy collection of large volume of sensory data from the public crowd.

In a typical MCS system, a central server, which is usually a cloud-based platform, aggregates and analyzes the sensory data submitted by participating users, namely (crowd) workers, whose mobile devices collect and may process in certain level the data before submitting them to the platform. Nowadays, applications of such MCS systems have already pervaded almost every corner of people’s everyday living and working. Examples include ambient environment (e.g., air quality [21], geomagnetic field [3], noise level [9]) monitoring, healthcare (e.g., disease trend prediction [5], drug side effect analysis [7], disease diagnosis [4]), floor plan reconstruction [16, 17, 38], smart transportation (e.g., pothole detection [30,80], traffic regulator classification [46], fuel-efficient map construction [36], traffic delay estimation [10,99]), indoor localization [93,108], and many others.

In a word, the explosion of applications of MCS systems, though having dramatically improved people’s living standard, poses new research challenges, which I discuss in great detail in this thesis.

### 1.1 Motivation and Challenges

Clearly, the power of crowd sensing could not be fully unleashed, unless MCS systems could attract a sufficient number of crowd workers to participate. However, participating in crowd sensing tasks

is usually costly for individual workers. On one hand, it consumes workers' resources, such as computing power, battery, and so forth. On the other hand, a considerable portion of sensing tasks require the submission of workers' sensitive and private information, which causes privacy leakage for participants. Without satisfactory rewards that properly compensate workers' participation costs, they will be rather reluctant to carry out the sensing tasks. Thus, in this thesis, I tackle the fundamental challenge of designing effective incentive mechanisms in order to achieve maximum worker participation in MCS systems.

An effective incentive mechanism should take into consideration workers' quality of information (QoI), as well as the preservation of workers' privacy.

- **Quality of Information:** In real practice, individual workers typically have diverse levels of reliability. The sensory data provided by less reliable workers may not accurately represent the real world. The possible reasons include poor sensor quality, lack of sensing effort, incomplete views of observations, environment and circuit board noise, lack of sensor calibration, and many others. Inaccurate information could mislead people's decisions, and eventually result in invaluable loss. Thus, it is highly necessary that an incentive mechanism incorporates workers' QoI, and selectively recruits workers who potentially could provide high quality sensory data to carry out the sensing tasks.
- **Privacy Preservation:** For an average crowd worker, participating in crowd sensing tasks may jeopardize her privacy in various aspects. On one hand, in auction-based incentive mechanisms, there possibly exist honest-but-curious workers who strictly follow the protocol of the system, but try to infer information about other workers' bids, which may oftentimes contain various types of workers' private information, including personal interest, knowledge base, location, and many others. On the other hand, the platform usually publishes the aggregated sensing results, which are beneficial to the community or society, but can cause privacy leakage to workers' data. Clearly, privacy leakage, if not tackled properly, could be a major factor that disincentivizes worker participation.

In a word, designing QoI aware and privacy-preserving incentive mechanisms that effectively incentivize worker participation in MCS systems is the challenge that I explore in this thesis.

## 1.2 Thesis Statement

Targeting the above challenge, I claim that the following thesis statement is true.

*By incorporating workers' quality of information and preserving their privacy, we can effectively incentivize worker participation, so as to improve the sensing coverage and sensing quality of mobile crowd sensing systems.*

## 1.3 Thesis Overview

In this thesis, towards the objective of effectively incentivizing worker participation, I present a series of incentive mechanisms for MCS systems. In this section, I provide an overview of each work and explain how they relate to each other.

### 1.3.1 QoI Aware Incentive Mechanisms for MCS Systems

Let us start with the discussion of QoI awareness, which is a crucial aspect that most of the existing incentive mechanisms ignore in their designs. Usually, the meaning of QoI varies in different applications. For example, in MCS systems that require workers to take and submit photos about a particular object or event [7, 8, 38], QoI refers to the quality (e.g., resolution, contrast, sharpness) of uploaded photos, as higher quality photos will help the platform better identify the object or event. As another example, in ambient environment monitoring systems [3, 9, 21], QoI means a worker's estimation accuracy of air quality, geomagnetic field, or noise level at a specific geographic location. As low quality sensory data could possibly lead to inaccurate aggregated sensing results or false decisions by the platform, which could eventually result in invaluable loss, QoI is clearly an important metric that should be considered in an incentive mechanism.

In this work [50], thus, I design QoI aware incentive mechanisms for MCS systems. I consider workers' strategic behavior, and design incentive mechanisms based on reverse auction, where the platform acts as the auctioneer that purchases sensory data from participating workers. Specifically, the proposed mechanisms yield close-to-optimal social welfare in a computationally efficient manner, which meanwhile satisfy other crucial desirable properties, namely truthfulness and individual rationality.

### 1.3.2 Incentivizing Multi-Requester Mobile Crowd Sensing

Currently, most of the existing incentive mechanisms, as well as the aforementioned QoI aware incentive mechanisms proposed in this thesis (in Section 1.3.1), assume that there is only one data requester who also serves as the platform in the MCS system. In practice, however, there are usually multiple data requesters competing for human resources, who usually outsource worker recruiting to third-party platforms (e.g., Amazon Mechanical Turk [1], Clickworker [2]) that have already gathered a large number of workers.

Therefore, in this work [54], I focus on MCS systems where three parties, including the data requesters, a platform, as well as a crowd of participating workers co-exist, and develop a novel incentive mechanism that can decide which worker serves which data requester at what price. Specifically, I propose a double auction-based incentive mechanism, which involves auctions among not only the workers, but also the data requesters, and is able to incentivize the participation of both data requesters and workers. I show through rigorous theoretical analyses that the proposed mechanism bears many desirable properties, including truthfulness, individual rationality, computational efficiency, as well as non-negative social welfare.

### 1.3.3 Bid Privacy-Preserving Incentive Mechanism for MCS Systems

In real practice, although the platform is usually considered to be trusted, there usually exist honest-but-curious workers who strictly follow the protocol of the system, but try to infer information about other workers' bids in auction-based incentive mechanisms, including our QoI aware (Section 1.3.1) and double auction-based (Section 1.3.2) incentive mechanisms, as well as many others.

Usually, the submitted bids contain various types of private and sensitive information about participating workers. For example, a worker's bidding task set could imply her personal interests, knowledge base, and so forth. In geotagging campaigns that provide accurate localization of physical objects (e.g., automated external defibrillator, pothole, litter), bidding task sets contain the places workers have visited or will visit, the disclosure of which breaches their location privacy. Similarly, bidding price could also be utilized to infer a worker's private information. For example, the types of mobile devices a worker uses for sensing tasks could possibly be implied from her bidding price, as usually workers tend to bid more if their mobile devices are more expensive. Thus,

it is of paramount importance to preserve workers’ bid privacy, so as to prevent them from being disincentivized by excessive privacy leakage.

To address this problem, in this work [51], I design an auction-based bid privacy-preserving incentive mechanism for MCS systems. I incorporate the notion of differential privacy [29, 77], and ensure that the change in any worker’s bid will not bring a significant change to our mechanism’s payments to participating workers. Such design philosophy reduces significantly the probability that a curious worker could successfully infer other workers’ bids from the different payments she receives in two rounds of the auction. Apart from privacy preservation, the proposed incentive mechanism also bears a suite of other desirable properties, including approximate truthfulness, individual rationality, computational efficiency, as well as yielding a guaranteed approximation ratio to the platform’s total payment.

### 1.3.4 Incentivizing Privacy-Preserving Data Aggregation in MCS Systems

Besides the bid privacy discussed in Section 1.3.3, participating workers in MCS systems usually face, as well, another type of equally possible and severe privacy breach, which is the leakage of their data privacy. In many MCS applications, the platform usually publishes the aggregated sensing results, which is oftentimes beneficial to the community or society, but jeopardizes workers’ privacy. Although the platform can be considered to be trusted, there exist adversaries highly motivated to infer workers’ data, which contain their sensitive and private information, from the published results. For example, publishing aggregated health data, such as treatment outcomes, improves people’s awareness about the effects of new drugs and medical devices, but poses threats to the privacy of participating patients. Therefore, it is entirely necessary for an MCS system to contain a data perturbation module that preserves workers’ data privacy by carefully perturbing the aggregated results before they are published.

In real practice, the various modules of an MCS system are far from isolated, but, in fact, interact with each other, and thus affect each other’s design. Thus, different from most of the past literature, in this work [53], I propose INCEPTION<sup>1</sup>, a novel MCS system framework with an integrated design of the incentive, data aggregation, and data perturbation mechanism. Specif-

---

<sup>1</sup>The name INCEPTION comes from INCEtive, Privacy, and data aggregaTION.

ically, INCEPTION has an auction-based incentive mechanism that selects reliable workers and compensates their costs for both sensing and privacy leakage, which meanwhile satisfies truthfulness and individual rationality, and minimizes the platforms total payment for worker recruiting with a guaranteed approximation ratio. The data aggregation mechanism of INCEPTION also incorporates workers' reliability and generates highly accurate aggregated results. Its data perturbation mechanism ensures satisfactory guarantee for the protection of workers privacy, as well as the accuracy of the final perturbed results.

## 1.4 Thesis Organization

In each of the next four chapters, I will elaborate on one of the four aforementioned incentive mechanisms, shedding light on its design philosophy, proving its desirable properties, and elaborating on the results of our extensive simulation. Specifically,

- In Chapter 2, I propose QoI aware incentive mechanisms for MCS systems, which selectively recruit workers who are more likely to provide high quality data in order to improve the quality of the final sensing results.
- In Chapter 3, to effectively incentivize participation in MCS systems where three parties, including the data requesters, a platform, as well as a crowd of participating workers co-exist, I develop a novel double auction-based incentive mechanism that can decide which worker serves which data requester at what price.
- In Chapter 4, to prevent workers from being disincentivized by excessive privacy leakage from their bids in auction-based incentive mechanisms, I propose a bid privacy-preserving incentive mechanism for MCS systems.
- In Chapter 5, I propose a joint framework with an integrated design of the incentive, data aggregation, and data perturbation mechanism, which captures the interactive effects among the different modules in MCS systems.

Finally, Chapter 6 concludes this thesis and provides a discussion of future research.

## Chapter 2

# QoI Aware Incentive Mechanisms for MCS Systems

### 2.1 Introduction

The ubiquity of human-carried mobile devices (e.g., smartphones, tablets, etc.) with a plethora of on-board and portable sensors (e.g., accelerometer, compass, camera, etc.) has given rise to the emergence of various people-centric mobile crowd sensing (MCS) systems [3–5, 7–10, 16, 17, 21, 30, 36, 38, 46, 80, 93, 99, 108]. In a typical MCS system, a cloud-based platform aggregates and analyzes the sensory data provided by the public crowd instead of professionals and dedicatedly deployed sensors. The mobile devices of participating users, namely (crowd) workers, collect and may process in certain level the data before submitting them to the platform.

Such MCS systems hold a wide spectrum of applications including healthcare, ambient environment monitoring, smart transportation, indoor localization, etc. For example, MedWatcher [7] is a US FDA advocated MCS system for post-market medical device surveillance. Participating workers upload photos of their medical devices to a cloud-based platform using the MedWatcher mobile application, which help identify visible problems with the devices. The platform aggregates and analyzes the worker-provided information, sends reports to the FDA and alerts users about medical device problems. Such a crowdsourcing paradigm enables easier detection of device safety issues and faster propagation of alerts to device users compared to traditional reporting methods such as mail or telephone. Moreover, air quality monitoring [21] is another area where MCS systems obtain their recent popularity. In such systems, crowdsourced air quality data are aggregated from a large number of people using air quality sensors ported to their smartphones, which help estimate the city or district level air quality.

Participating in such crowd sensing tasks is usually a costly procedure for individual workers. On one hand, it consumes workers' resources, such as computing power, battery and so forth.

On the other hand, a considerable portion of sensing tasks require the submission of some types of workers’ sensitive private information, which causes privacy leakage for participating workers. For example, by uploading the photos of their medical devices, workers reveal the types of their illnesses. By submitting air quality estimation samples, workers usually reveal information about their locations. Therefore, without satisfactory rewards that compensate participating costs, workers will be reluctant to carry out the sensing tasks. However, most of the existing MCS systems are based on voluntary worker participation or lack effective incentive mechanisms.

Aware of the paramount importance of stimulating worker participation, the research community has recently developed various game-theoretic incentive mechanisms for MCS systems [20,22,27,28,31–33,35,37,43,44,51,53,54,60,62,71–75,82,91,92,97,98,100,106,107,110–113,115–124]. However, most of the existing mechanisms fail to incorporate one important aspect, that is workers’ *quality of information (QoI)*, into their designs. The meaning of QoI varies in different applications. For example, in the aforementioned MedWatcher system [7] QoI refers to the quality (e.g., resolution, contrast, sharpness, etc.) of uploaded photos. Higher quality ones will help the platform better identify visible device problems. In air quality monitoring MCS systems [21], QoI means a worker’s estimation accuracy of air quality. The QoI of every worker could be affected by various factors, including poor sensor quality, noise, lack of sensor calibration and so forth.

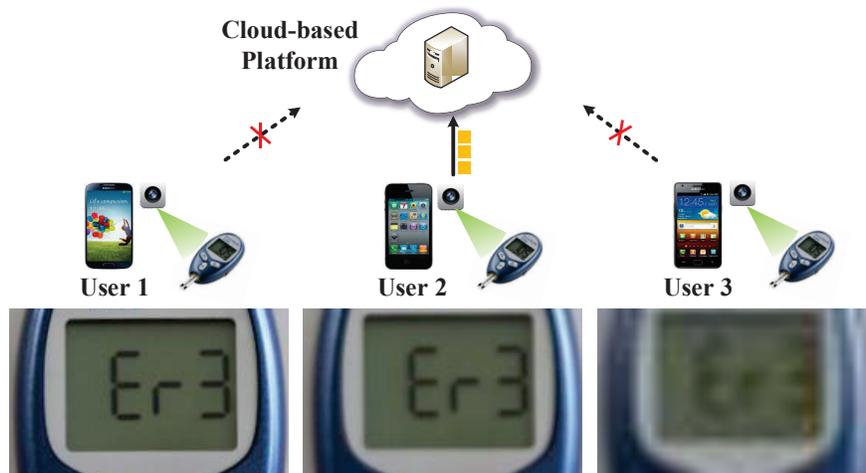


Figure 2.1: An example of the MedWatcher MCS system<sup>1</sup>

<sup>1</sup>In this example, 3 workers try to upload the photos of the error message "Er3" on the screens of their blood glucose meters to the MedWatcher platform. The prices that the 3 workers ask for cost compensation are 100\$, 10\$ and 1\$ respectively.

To compensate the cost of each worker’s participation, existing incentive mechanisms have used the worker’s bidding price as an important metric to allocate sensing tasks. However, as shown in the example in Figure 2.1, QoI is also a major factor that should be considered together with bidding price. Although worker 1 has the highest quality photo, her high price prohibits the platform from requesting her data. Furthermore, despite worker 3’s low price the platform will not be interested in her data either, because her low quality photo could hardly contribute to identifying the error message "Er3". By jointly considering price and QoI, the platform will select worker 2 with medium price and acceptable photo quality as the data provider. Therefore, our goal is to *design QoI aware incentive mechanisms for MCS systems*.

Considering workers’ *strategic* behaviors and the combinatorial nature of the tasks that every worker executes, we design incentive mechanisms based on *reverse combinatorial auctions*, where the platform acts as the auctioneer that purchases the data from participating workers. Not only do we study the *single-minded* scenario where every worker is willing to execute one subset of tasks, but also we investigate the *multi-minded* case in which any worker might be interested in executing multiple subsets of tasks. Similar to the traditional VCG mechanisms [23, 41], our mechanisms also aim to maximize the social welfare. Mechanism design for combinatorial auctions is typically challenging in that usually we aim to design a computationally efficient mechanism with close-to-optimal social welfare in the presence of an NP-hard winner determination problem, which meanwhile satisfies *truthfulness* and *individual rationality*. Addressing all these challenges, this chapter makes the following contributions.

- Different from most of the previous work, we design *QoI aware* incentive mechanisms for MCS systems.
- We use reverse combinatorial auction to design a truthful, individual rational and computationally efficient incentive mechanism that approximately maximizes the social welfare with a guaranteed approximation ratio for the single-minded case.
- For the multi-minded reverse combinatorial auction, we design an iterative descending mechanism that achieves close-to-optimal social welfare with a guaranteed approximation ratio while satisfying individual rationality and computational efficiency.

The rest of the chapter is organized as follows. We introduce the preliminaries in Section 2.2, and describe the design of our incentive mechanisms for the single-minded and multi-minded case in Section 2.3 and 2.4, respectively. Next, we present the results of our extensive simulation in Section 2.5, and summarize the related work in Section 2.6. Finally, we conclude this chapter in Section 2.7.

## 2.2 Preliminaries

In this section, we present an overview of MCS systems, our auction model and design objectives.

### 2.2.1 System Overview

The MCS system model considered in this chapter consists of a platform residing in the cloud and a set of  $N$  workers, denoted as  $\mathcal{N} = \{1, \dots, N\}$ . The workers execute a set of  $M$  sensing tasks, denoted as  $\mathcal{T} = \{\tau_1, \dots, \tau_M\}$  and send their sensory data to the platform. The workflow<sup>2</sup> of the system is described as follows.

1. Firstly, the platform announces the set of sensing tasks,  $\mathcal{T}$ , to workers.
2. Then, the platform and workers enter the auctioning stage in which the platform acts as the auctioneer that purchases the sensory data collected by individual workers. Every worker  $i \in \mathcal{N}$  submits her bid, which is a tuple  $(\Gamma_i, b_i)$  consisting of the set of tasks  $\Gamma_i \subseteq \mathcal{T}$  she wants to execute and her bidding price  $b_i$  for executing these tasks.
3. Based on workers' bids, the platform determines the set of winners, denoted as  $\mathcal{S} \subseteq \mathcal{N}$  and the payment to all workers, denoted as  $\vec{p} = \{p_1, \dots, p_N\}$ . Specifically, a loser does not execute any task and receives zero payment.
4. After the platform receives winners' sensory data, it gives the payment to the corresponding winners.

---

<sup>2</sup>Note that we are specifically interested in the scenario where all workers and tasks arrive at same time. We leave the investigation of the online scenario where workers and tasks arrive sequentially in an online manner in our future work.

One major difference between this chapter and most of the previous work is that we integrate workers' quality of information (QoI), denoted as  $\vec{q} = \{q_1, \dots, q_N\}$ , into the design of our incentive mechanisms. Generally speaking, QoI indicates the quality of workers' sensory data. The definition of QoI varies in different applications. For example, in the previously mentioned MedWatcher system [7], QoI refers to the quality (e.g., resolution, contrast, sharpness) of uploaded photos. Photos with higher quality will help the platform better identify visible problems with medical devices. In air quality monitoring systems [21], QoI refers to a worker's estimation accuracy of air quality. We assume that the platform maintains a historical record of workers' QoI profile  $\vec{q}$  used as inputs for winner and payment determination. There are many methods for the platform to compute workers' QoIs. In the cases where the platform has adequate ground truth data, QoIs can be obtained by directly computing the deviation of workers' data from the ground truths. However, even without ground truths, QoIs can still be effectively inferred from workers' data by algorithms such as those proposed in [68–70, 78, 96, 101]. Alternatively, QoIs can be inferred from other factors (e.g., the price of a worker's sensors, her experience and reputation for specific sensing tasks) using methods proposed in previous studies such as [63]. The problem of which method the platform adopts to compute workers' QoIs is application dependent and out of the scope of this chapter. Typically, workers may know some of the factors that affect their QoIs. However, they usually do not know exactly how QoIs are computed by the platform, and thus do not know the exact values of their QoIs.

### 2.2.2 Auction Model

In this chapter, we consider *strategic* and *selfish* workers that aim to maximize their own utilities. The fact that workers bid on subsets of tasks motivates us to use *reverse combinatorial auction* to model the problem. In the rest of the chapter, we use *bundle* to refer to any subset of tasks of  $\mathcal{T}$ . Different from traditional *forward combinatorial auction* [13, 15], in this paper, we formally define the concept of reverse combinatorial auction that is applied in our problem setting in the following Definition 1.

**Definition 1** (RC Auction). *In a reverse combinatorial auction (RC auction), each worker  $i \in \mathcal{N}$  is interested in a set of  $K_i \geq 1$  bundles, denoted as  $\mathcal{T}_i = \{\Gamma_i^1, \dots, \Gamma_i^{K_i}\}$ . For any bundle  $\Gamma \subseteq \mathcal{T}$ ,*

the worker has a cost function defined in Equation (2.1).

$$C_i(\Gamma) = \begin{cases} c_i, & \text{if } \exists \Gamma_i^j \in \mathcal{T}_i \text{ s.t. } \Gamma \subseteq \Gamma_i^j \\ +\infty, & \text{otherwise} \end{cases}. \quad (2.1)$$

Both  $\mathcal{T}_i$  and the cost function  $C_i(\cdot)$  are worker  $i$ 's private information. If  $K_i = 1$  for every worker, then the auction is defined as a single-minded reverse combinatorial auction (SRC auction). And it is defined as a multi-minded reverse combinatorial auction (MRC auction), if  $K_i > 1$  for at least one worker.

In an SRC auction,  $\mathcal{T}_i$  contains only worker  $i$ 's maximum executable task set  $\bar{\Gamma}_i$ . That is,  $\bar{\Gamma}_i$  consists of all the sensing tasks that worker  $i$  is able to execute. Since she is not capable to carry out tasks beyond  $\bar{\Gamma}_i$ , her cost for any bundle  $\Gamma \not\subseteq \bar{\Gamma}_i$  can be equivalently viewed as  $+\infty$ . Similarly in an MRC auction, the union of all the bundles in  $\mathcal{T}_i$  is  $\bar{\Gamma}_i$ . That is,  $\bigcup_{j=1}^{K_i} \Gamma_i^j = \bar{\Gamma}_i$ . If worker  $i$  is a winner of the RC auction, she will be paid  $p_i$  for executing the corresponding set of sensing tasks. In contrast, she will not be allocated any sensing task and will receive zero payment if she is a loser. We present the definitions of the utility of a worker and the profit of the platform formally in Definition 2 and 3.

**Definition 2** (A Worker's Utility). *The utility of any worker  $i \in \mathcal{N}$  is*

$$u_i = \begin{cases} p_i - c_i, & \text{if } i \in \mathcal{S} \\ 0, & \text{otherwise} \end{cases}. \quad (2.2)$$

**Definition 3** (Platform's Profit). *The profit of the platform given workers' QoI profile  $\vec{q}$  is*

$$u_0 = V_{\vec{q}}(\mathcal{S}) - \sum_{i \in \mathcal{S}} p_i, \quad (2.3)$$

where the value function  $V_{\vec{q}}(\cdot) : 2^{\mathcal{N}} \rightarrow \mathbb{R}^+$  maps the winner set  $\mathcal{S}$  to the value that the winners bring to the platform. Furthermore,  $V_{\vec{q}}(\cdot)$  is monotonic in  $\vec{q}$ . That is, for any  $\vec{q} = \{q_1, \dots, q_N\}$  and  $\vec{q}' = \{q'_1, \dots, q'_N\}$  such that  $q_i \geq q'_i$  holds  $\forall i \in \mathcal{N}$ , we have  $V_{\vec{q}}(\mathcal{S}) \geq V_{\vec{q}'}(\mathcal{S})$ .

Similar to the traditional VCG mechanism design [23, 41], we aim to design mechanisms that

maximize the social welfare, which is formally defined in Definition 4.

**Definition 4** (Social Welfare). *The social welfare of the whole MCS system is*

$$u_{\text{social}} = u_0 + \sum_{i \in \mathcal{N}} u_i = V_{\vec{q}}(\mathcal{S}) - \sum_{i \in \mathcal{S}} c_i. \quad (2.4)$$

### 2.2.3 Design Objective

In this chapter, we aim to design *dominant-strategy mechanisms* in which for every worker there exists a dominant strategy [83] defined in Definition 5.

**Definition 5** (Dominant Strategy). *A strategy  $st_i$  is the dominant strategy for worker  $i$  if and only if for any other strategy  $st'_i$  and any strategy profile of the other workers, denoted as  $st_{-i}$ , the property  $u_i(st_i, st_{-i}) \geq u_i(st'_i, st_{-i})$  holds.*

In our SRC auction, each worker submits to the platform a bid  $(\Gamma_i, b_i)$  consisting of her declared interested bundle  $\Gamma_i$  and the bidding price  $b_i$ . Since workers are strategic, it is possible that she declares a bid that deviates from the true value  $(\bar{\Gamma}_i, c_i)$ . However, one of our goals for the SRC auction is to design a *truthful* mechanism defined in Definition 6.

**Definition 6** (Truthfulness). *An SRC auction is truthful if and only if it is the dominant strategy for every worker  $i \in \mathcal{N}$  to bid her true value  $(\bar{\Gamma}_i, c_i)$ .*

Noticed from Definition 6 that we aim to ensure the truthfulness of both the cost  $c_i$  and bundle  $\bar{\Gamma}_i$ . Besides truthfulness, another design objective for the SRC auction is to ensure that every worker receives non-negative utility from participating. Such property is critical in incentive mechanisms because it ensures that workers will not be disincentivized to participate for receiving negative utilities. This property is defined as *individual rationality* in Definition 7.

**Definition 7** (Individual Rationality). *A mechanism is individual rational (IR) if and only if  $u_i \geq 0$  is satisfied for every worker  $i \in \mathcal{N}$ .*

As mentioned in Section 2.2.2, our mechanism aims to maximize the social welfare. However, as will be proved in Section 2.3, the problem of maximizing the social welfare in the SRC auction

is NP-hard. Hence, we aim to design a *polynomial-time* mechanism that gives us approximately optimal social welfare with a *guaranteed approximation ratio*.

In the domain of multi-minded combinatorial auction, requiring truthfulness limits the family of mechanisms that can be used, as pointed out in [12]. Hence, in our MRC auction, we aim to design a dominant-strategy mechanism that can still yield a *guaranteed approximation ratio* to the optimal social welfare without ensuring truthfulness. In fact, as mentioned in [13], the requirement of truthfulness is only to obtain close-to-optimal social welfare with strategic worker behaviors, but not the real essence. Therefore, as long as the approximation ratio is guaranteed when workers play their dominant strategies, it is justifiable for us to relax the truthfulness requirement. Additionally, we also require our mechanism to be *individual rational* and have a *polynomial* computational complexity.

Model	Dominant Strategy	Truthful	IR	Approx. Ratio	Complexity
SRC	✓	✓	✓	Guaranteed	Polynomial
MRC	✓	×	✓	Guaranteed	Polynomial

Table 2.1: Summary of design objectives

Authors in [13, 125] address the issue of mechanism design for multi-minded forward combinatorial auctions. Their mechanisms cannot ensure that workers have dominant strategies and cannot be applied to reverse combinatorial auctions. However, in contrast, we are able to design a dominant-strategy incentive mechanism for the MRC auction in this chapter. We summarize our design objectives for both the SRC and MRC auctions in Table 2.1.

## 2.3 SRC Auction

In this section, we introduce the mathematical formulation, mechanism design, an intuitive walk-through example and the corresponding analysis for the SRC auction.

### 2.3.1 Mathematical Formulation

In our SRC auction, each worker’s bid  $(\Gamma_i, b_i)$  consists of her declared interested bundle  $\Gamma_i$  and the bidding price  $b_i$ . Although our model is valid for any general value function  $V_{\vec{q}}(\cdot)$  that satisfies Definition 3, to simplify our analysis we assume that  $V_{\vec{q}}(\cdot)$  is the sum of the value,  $v_i$ , contributed

by every winner  $i \in \mathcal{S}$ . Furthermore, we assume that  $v_i$  is proportional to the total QoI provided by this worker. Given workers' bidding bundle profile  $\vec{\Gamma} = \{\Gamma_1, \dots, \Gamma_N\}$  and the winner set  $\mathcal{S}$ , the platform's value function  $V_{\vec{q}}(\cdot)$  can be represented by Equation (2.5).

$$V_{\vec{q}}(\mathcal{S}) = \sum_{i \in \mathcal{S}} v_i = \sum_{i \in \mathcal{S}} \alpha q_i |\Gamma_i|, \quad (2.5)$$

where  $\alpha$  is a coefficient that transforms QoI to monetary reward.

Another aspect that distinguishes this chapter from previous work is that we consider *QoI coverage* in the SRC auction. For the task that none of the workers capable to execute it has adequately high QoI, collective efforts of multiple workers are necessary to ensure sensing quality. We use  $Q_{\tau_j, \vec{q}}(\mathcal{S})$  to denote the total QoI that all winners have on task  $\tau_j \in \mathcal{T}$ . Furthermore, we approximate  $Q_{\tau_j, \vec{q}}(\mathcal{S})$  as the sum of the QoIs of the winners that execute this task. Therefore, QoI coverage is equivalent to guaranteeing that every task is executed by workers with sufficient amount of QoI in total. Based on this additive assumption of QoI,  $Q_{\tau_j, \vec{q}}(\mathcal{S})$  can be represented by Equation (2.6).

$$Q_{\tau_j, \vec{q}}(\mathcal{S}) = \sum_{i: \tau_j \in \Gamma_i, i \in \mathcal{S}} q_i. \quad (2.6)$$

Since we aim to maximize the social welfare given in Definition 4, the winner determination and pricing can be decoupled into two separate problems. We formulate the SRC auction winner determination (SRC-WD) problem as the following integer linear program.

**SRC-WD Problem:**

$$\max \sum_{i \in \mathcal{N}} (\alpha q_i |\Gamma_i| - b_i) x_i \quad (2.7)$$

$$\text{s.t.} \quad \sum_{i: \tau_j \in \Gamma_i, i \in \mathcal{N}} q_i x_i \geq Q_j, \quad \forall \tau_j \in \mathcal{T} \quad (2.8)$$

$$x_i \in \{0, 1\}, \quad \forall i \in \mathcal{N} \quad (2.9)$$

**Constants.** The SRC-WD problem takes as input constants  $\alpha$ , workers' bid profile  $\{(\Gamma_1, b_1), \dots, (\Gamma_N, b_N)\}$ , workers' QoI profile  $\vec{q}$  and tasks' QoI requirement profile  $\vec{Q} = \{Q_1, \dots, Q_M\}$ .

**Variables.** In the SRC-WD problem, we have a set of binary variables  $\{x_1, \dots, x_N\}$  for every worker  $i \in \mathcal{N}$ . If worker  $i$  is in the winner set  $\mathcal{S}$ , then  $x_i = 1$ . Otherwise,  $x_i = 0$ .

**Objective function.** Since the platform does not know the true values of workers' interested bundles and the corresponding costs,  $\{(\bar{\Gamma}_1, c_1), \dots, (\bar{\Gamma}_N, c_N)\}$ , the objective function that it directly tries to maximize is the social welfare based on workers' bid profile  $\{(\Gamma_1, b_1), \dots, (\Gamma_N, b_N)\}$ . We use  $\vec{w} = \{w_1, \dots, w_N\}$ , in which  $w_i = \alpha q_i |\Gamma_i| - b_i$ , to denote the marginal social welfare profile of all workers based on workers' bids. Then, we have the objective function  $\sum_{i \in \mathcal{S}} w_i = \sum_{i \in \mathcal{S}} (\alpha q_i |\Gamma_i| - b_i) = \sum_{i \in \mathcal{N}} (\alpha q_i |\Gamma_i| - b_i) x_i$ . Later in Section 2.3.4, we will show that in our mechanism every worker in fact bids truthfully. Hence, the objective function is equivalent to the actual social welfare.

**Constraints.** Constraint (2.8) represents the QoI coverage for every task  $\tau_j \in \mathcal{T}$ , which ensures that the total QoI of all the winners for this task, calculated as  $Q_{\tau_j, \vec{q}}(\mathcal{S}) = \sum_{i: \tau_j \in \Gamma_i, i \in \mathcal{S}} q_i = \sum_{i: \tau_j \in \Gamma_i, i \in \mathcal{N}} q_i x_i$ , is no less than the QoI requirement  $Q_j$ .

Next, we prove the NP-hardness of the SRC-WD problem.

**Theorem 1.** *The SRC-WD problem is NP-hard.*

*Proof.* In this proof, we demonstrate that the NP-complete minimum weight set cover (MWSC) problem is polynomial-time reducible to the SRC-WD problem. The reduction starts with an instance of the MWSC problem consisting of a universe of elements  $\mathcal{U} = \{\tau_1, \dots, \tau_M\}$  and a set of  $N$  sets  $\mathcal{O} = \{\Gamma_1, \dots, \Gamma_N\}$  whose union equals  $\mathcal{U}$ . Every set  $\Gamma_i \in \mathcal{O}$  is associated with a non-negative weight  $w_i$ . The MWSC problem is to find the subset of  $\mathcal{O}$  with the minimum total weight whose union contains all the elements in  $\mathcal{U}$ .

Based on the instance of the MWSC problem, we construct an instance of the SRC-WD problem. Firstly, we transform  $\Gamma_i$  into  $\Gamma'_i$  such that for every element in  $\Gamma_i$  there exist  $l_i \in \mathbb{Z}^+$  copies of the same element in  $\Gamma'_i$ . We require that every element  $\tau_j \in \mathcal{U}$  is covered for at least  $L_j \in \mathbb{Z}^+$  times. After the reduction, we obtain an instance of the SRC-WD problem in which workers' QoI profile is  $\vec{q} = \{q_1, \dots, q_N\}$ , workers' bidding bundle profile is  $\vec{\Gamma} = \{\Gamma_1, \dots, \Gamma_N\}$ , workers' marginal social welfare profile is  $\vec{w} = \{-w_1, \dots, -w_N\}$  and tasks' QoI requirement profile is  $\vec{Q} = \{L_1, \dots, L_M\}$ . Noticed that the SRC-WD problem represents a richer family of problems in which any worker  $i$ 's QoI,  $q_i$ , and any task  $j$ 's QoI requirement,  $Q_j$ , could take any value in  $\mathbb{R}^+$ . Furthermore, the marginal social welfare can take any value in  $\mathbb{R}$ . Hence, every instance of the MWSC problem

is polynomial-time reducible to an instance of the SRC-WD problem. The SRC-WD problem is NP-hard.  $\square$

### 2.3.2 Mechanism Design

Because of the NP-hardness of the SRC-WD problem, it is impossible to compute the set of winners that maximize the social welfare in polynomial time unless  $P = NP$ . As a result, we cannot use the off-the-shelf VCG mechanism [23, 41] since the truthfulness of VCG mechanism requires that the social welfare is exactly maximized. Therefore, as mentioned in Section 2.2.3, we aim to design a mechanism that approximately maximizes the social welfare while guaranteeing truthfulness.

Myerson's characterizations of truthfulness for single-parameter auctions [81] are not directly applicable in our scenario, because our SRC auction is a double-parameter auction that considers both bundle and cost truthfulness. Moreover, different from the characterizations of truthfulness for single-minded forward combinatorial auctions proposed in [15], we describe and prove the necessary and sufficient conditions for a truthful SRC auction in Lemma 1.

**Lemma 1.** *An SRC auction is truthful if and only if the following two properties hold:*

- **Monotonicity.** *Any worker  $i$  who wins by bidding  $(\Gamma_i, b_i)$  still wins by bidding any  $b'_i < b_i$  and any  $\Gamma'_i \supset \Gamma_i$  given that other workers' bids are fixed.*
- **Critical payment.** *Any winner  $i$  with bid  $(\Gamma_i, b_i)$  is paid the supremum of all bidding prices  $b'_i$  such that bidding  $(\Gamma_i, b'_i)$  still wins, which is defined as worker  $i$ 's critical payment.*

*Proof.* It is easily verifiable that a truthful bidder will never receive negative utility. If worker  $i$ 's any untruthful bid  $(\Gamma_i, b_i)$  is losing or  $\Gamma_i \not\subseteq \bar{\Gamma}_i$ , her utility from bidding  $(\Gamma_i, b_i)$  will be non-positive. Therefore, we only need to consider the case in which  $(\Gamma_i, b_i)$  is winning and  $\Gamma_i \subseteq \bar{\Gamma}_i$ .

- Because of the property of monotonicity,  $(\bar{\Gamma}_i, b_i)$  is also a winning bid. Suppose the payment for bid  $(\Gamma_i, b_i)$  is  $p$  and that for bid  $(\bar{\Gamma}_i, b_i)$  is  $\bar{p}$ . Every bid  $(\bar{\Gamma}_i, b'_i)$  with  $b'_i > \bar{p}$  is losing because  $\bar{p}$  is the worker  $i$ 's critical payment given bundle  $\bar{\Gamma}_i$ . From monotonicity, bidding  $(\Gamma_i, b'_i)$  is also losing. Therefore, the critical payment for  $(\Gamma_i, b_i)$  is at most that for  $(\bar{\Gamma}_i, b_i)$ , which means  $p \leq \bar{p}$ . Hence, the worker will not increase her utility by bidding  $(\Gamma_i, b_i)$  instead of  $(\bar{\Gamma}_i, b_i)$ .

- Then, we consider the case in which bidding truthfully  $(\bar{\Gamma}_i, c_i)$  wins. This bid earns the same payment  $\bar{p}$  as  $(\bar{\Gamma}_i, b_i)$ . Then her utilities from these two bids will be the same. If bidding  $(\bar{\Gamma}_i, c_i)$  loses, then we have  $c_i > \bar{p} \geq b_i$ . Hence, bidding  $(\bar{\Gamma}_i, b_i)$  will receive negative utility. Therefore,  $(\bar{\Gamma}_i, b_i)$  will also not increase her utility compared to  $(\bar{\Gamma}_i, c_i)$ .

Thus, we conclude that an SRC auction is truthful if and only if the monotonicity and critical payment properties hold.  $\square$

We utilize the rationale provided in Lemma 1 to design a *quality of information aware SRC (QoI-SRC) auction*. Specifically, we present the winner determination and pricing mechanisms of the QoI-SRC auction respectively in Algorithm 1 and 2.

---

**Algorithm 1:** QoI-SRC Auction Winner Determination

---

**Input:**  $\mathcal{T}, \mathcal{N}, \vec{w}, \vec{q}, \vec{Q}, \vec{\Gamma}$ ;  
**Output:**  $\mathcal{S}$ ;  
// Initialization  
1  $\mathcal{N}^- \leftarrow \emptyset, \mathcal{S} \leftarrow \emptyset$ ;  
// Select non-negative marginal social welfare workers  
2 **foreach**  $i$  s.t.  $w_i \geq 0$  **do**  
3      $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$ ;  
4  $\mathcal{N}^- \leftarrow \mathcal{N} \setminus \mathcal{S}$ ;  
// Calculate residual QoI requirement  
5 **foreach**  $j$  s.t.  $\tau_j \in \mathcal{T}$  **do**  
6      $Q'_j \leftarrow Q_j - \min\{Q_j, \sum_{i:\tau_j \in \Gamma_i, i \in \mathcal{S}} q_i\}$ ;  
// Main loop  
7 **while**  $\sum_{j:\tau_j \in \mathcal{T}} Q'_j \neq 0$  **do**  
8     // Find the worker with the minimum marginal social welfare effectiveness  
9      $l = \arg \min_{i \in \mathcal{N}^-} \frac{|w_i|}{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_i\}}$ ;  
10      $\mathcal{S} \leftarrow \mathcal{S} \cup \{l\}$ ;  
11      $\mathcal{N}^- \leftarrow \mathcal{N}^- \setminus \{l\}$ ;  
12     // Update residual requirement  
13     **foreach**  $j$  s.t.  $\tau_j \in \mathcal{T}$  **do**  
14          $Q'_j \leftarrow Q'_j - \min\{Q'_j, q_l\}$ ;  
15 **return**  $\mathcal{S}$ ;

---

The platform calculates workers' marginal social welfare profile  $\vec{w}$  using workers' bids  $\{(\Gamma_1, b_1), \dots, (\Gamma_N, b_N)\}$  and utilizes  $\vec{w}$  as input to the winner determination algorithm shown in Algorithm 1. Firstly, the platform includes all workers with non-negative marginal social welfare

into the winner set  $\mathcal{S}$  (line 2-3). By removing the current winners from  $\mathcal{N}$ , the platform gets the set of workers  $\mathcal{N}^-$  with negative marginal social welfare (line 4). Then, the platform calculates tasks' residual QoI requirement profile  $\vec{Q}'$  by subtracting from  $\vec{Q}$  the QoI provided by the currently selected winners (line 5-6). The main loop (line 7-12) is executed until every task's QoI requirement is satisfied. In the main loop, winner selection is based on *marginal social welfare effectiveness* (*MSWE*), defined as the ratio between the absolute value of worker  $i$ 's marginal social welfare  $|w_i|$  and her effective QoI contribution  $\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_i\}$ . In every iteration, the worker with the minimum MSWE among the remaining workers in  $\mathcal{N}^-$  is included into  $\mathcal{S}$  (line 8-9). After that, the platform updates  $\mathcal{N}^-$  and tasks' residual QoI requirement profile  $\vec{Q}'$  (line 10-12).

---

**Algorithm 2:** QoI-SRC Auction Pricing

---

**Input:**  $\mathcal{S}, \alpha, \vec{q}, \vec{w}, \vec{\Gamma}$ ;  
**Output:**  $\vec{p}$ ;  
// Initialization  
1  $\mathcal{N}^+ \leftarrow \emptyset, \vec{p} \leftarrow \{0, \dots, 0\}$ ;  
// Find non-negative marginal welfare workers  
2 **foreach**  $i$  *s.t.*  $w_i \geq 0$  **do**  
3    $\mathcal{N}^+ \leftarrow \mathcal{N}^+ \cup \{i\}$ ;  
// Main loop  
4 **foreach**  $i \in \mathcal{S}$  **do**  
5   run Algorithm 1 on  $\mathcal{N} \setminus \{i\}$  until  $\sum_{j:\tau_j \in \Gamma_i} Q'_j = 0$ ;  
6    $\mathcal{S}' \leftarrow$  the winner set when step 5 stops;  
// Calculate payment  
7   **if**  $|\mathcal{S}'| < |\mathcal{N}^+|$  **then**  
8      $p_i \leftarrow \alpha q_i |\Gamma_i|$ ;  
9   **else**  
10    **foreach**  $k \in \mathcal{S}' \setminus \mathcal{N}^+$  **do**  
11      $\vec{Q}' \leftarrow$  tasks' residual QoI requirement profile when winner  $k$  is selected;  
12      $p_i \leftarrow \max \left\{ p_i, \alpha q_i |\Gamma_i| - w_k \frac{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_i\}}{\sum_{j:\tau_j \in \Gamma_k} \min\{Q'_j, q_k\}} \right\}$ ;  
13 **return**  $\vec{p}$ ;

---

Algorithm 2 describes the pricing mechanism. It takes the winner set  $\mathcal{S}$  as input and outputs the payment profile  $\vec{p}$ . Firstly,  $\vec{p}$  is initialized as a zero vector (line 1). Then, the platform includes all workers with non-negative marginal social welfare into  $\mathcal{N}^+$  (line 2-3). The main loop (line 4-12) computes the platform's payment to each winner. For each winner  $i \in \mathcal{S}$ , the winner determination mechanism in Algorithm 1 is executed with all workers except worker  $i$  until the

QoI requirement of every task in  $\Gamma_i$  has been satisfied (line 5). We reach the point such that it is impossible for worker  $i$  to be selected as a winner in future iterations of Algorithm 1. Then, the platform gets the current winner set  $\mathcal{S}'$  (line 6) and computes  $p_i$  in the following two cases.

- **Case 1** (line 7-8). Any winner  $i$  belonging to case 1 has  $w_i \geq 0$ . Hence, this worker's critical payment is the bidding price  $b'_i$  that satisfies  $w'_i = \alpha q_i |\Gamma_i| - b'_i = 0$ . That is,  $p_i = \alpha q_i |\Gamma_i|$ .
- **Case 2** (line 10-12). For any winner  $i$  belonging to case 2, we go through every worker  $k \in \mathcal{S}' \setminus \mathcal{N}^+$ . We calculate worker  $i$ 's maximum bidding price  $b'_i$  to be able to substitute worker  $k$  as the winner. That is,  $b'_i$  satisfies Equation (2.10).

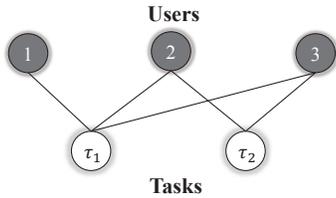
$$\frac{b'_i - \alpha q_i |\Gamma_i|}{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_i\}} = \frac{|w_k|}{\sum_{j:\tau_j \in \Gamma_k} \min\{Q'_j, q_k\}}. \quad (2.10)$$

This means

$$b'_i = \alpha q_i |\Gamma_i| - w_k \frac{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_i\}}{\sum_{j:\tau_j \in \Gamma_k} \min\{Q'_j, q_k\}}. \quad (2.11)$$

Finally, the maximum value among all  $b'_i$ 's is used as the payment to worker  $i$ .

### 2.3.3 Walk-through Example



Parameters	Value
$\alpha$	0.1
$\vec{q}$	{0.8, 1.2, 1.2}
$\vec{b}$	{0.2, 2.6, 2.7}
$\vec{Q}$	{1.1, 0.8}
$\vec{w}$	{0.4, -0.2, -0.3}

Figure 2.2: Bidding graph of the example

Table 2.2: Parameter setting of the example

In this section, we use a simple toy example to illustrate how the QoI-SRC auction works. In this example, there are 3 workers  $\mathcal{N} = \{1, 2, 3\}$  and 2 tasks  $\mathcal{T} = \{\tau_1, \tau_2\}$ . In Figure 2.2, an edge between a worker  $i$  and a task  $\tau_j$  indicates that  $\tau_j \in \Gamma_i$ . That is, workers' bidding bundles are  $\Gamma_1 = \{\tau_1\}$ ,  $\Gamma_2 = \{\tau_1, \tau_2\}$  and  $\Gamma_3 = \{\tau_1, \tau_2\}$ . Other parameters are shown in Table 2.2. We then demonstrate the process of the QoI-SRC winner determination and pricing in the following Table 2.3 and 2.4, respectively.

$\mathcal{S}$	$\vec{Q}'$	New Winner	Reason
$\emptyset$	$\{1.1, 0.8\}$	$\{1\}$	$w_1 > 0$
$\{1\}$	$\{0.3, 0.8\}$	$\{2\}$	$\text{MSWE}_{\text{worker } 2} = \frac{0.2}{1.1} < \text{MSWE}_{\text{worker } 3} = \frac{0.3}{1.1}$
$\{1, 2\}$	$\{0, 0\}$	$\emptyset$	$Q'_1 = Q'_2 = 0$ and $w_3 < 0$

Table 2.3: An example of the QoI-SRC auction's winner determination procedure

Winner	$\mathcal{S}'$	Payment
1	$\{2\}$	$p_1 = 0.1 \times 0.8 \times 1 + 0.2 \times \frac{0.8}{1.9} \approx 0.164$
2	$\{1, 3\}$	$p_2 = 0.1 \times 1.2 \times 2 + 0.3 \times \frac{1.1}{1.1} \approx 0.540$

Table 2.4: An example of QoI-SRC auction's pricing procedure

### 2.3.4 Analysis

Firstly, we prove that the QoI-SRC auction is truthful and individual rational in Theorem 2 and 3.

**Theorem 2.** *The QoI-SRC auction is truthful.*

*Proof.* Suppose worker  $i$  wins by bidding  $(\Gamma_i, b_i)$ . We consider worker  $i$ 's any other bid  $(\Gamma'_i, b'_i)$  such that  $b'_i < b_i$  or  $\Gamma'_i \supset \Gamma_i$ .

- **Case 1** ( $w_i \geq 0$ ). The marginal social welfare for bidding  $(\Gamma'_i, b'_i)$  is  $w'_i = \alpha q_i |\Gamma'_i| - b'_i > \alpha q_i |\Gamma_i| - b_i \geq 0$ .
- **Case 2** ( $w_i < 0$ ). Bidding  $(\Gamma'_i, b'_i)$  will make  $w'_i \geq 0$  or decrease the value of worker  $i$ 's MSWE.

Hence, worker  $i$  is still a winner by bidding  $(\Gamma'_i, b'_i)$  and the QoI-SRC auction winner determination algorithm satisfies both bidding bundle and price monotonicity. Furthermore, it is easily verifiable that the pricing mechanism in Algorithm 2 uses the supremum of bidding prices  $b'_i$  such that bidding  $(\Gamma_i, b'_i)$  still wins. Hence, from Lemma 1 we conclude that the QoI-SRC auction is truthful.  $\square$

**Theorem 3.** *The QoI-SRC auction is individual rational.*

*Proof.* From Theorem 2, we have proved that workers bid truthfully in our QoI-SRC auction. Hence, any worker  $i$  bids its true cost  $c_i$ . Since every winner  $i$  is paid the supremum of bidding prices given the bundle  $\Gamma_i$ , we have  $p_i \geq c_i$  for every winner. Apparently, losers have zero utilities in our QoI-SRC auction. Therefore, the utility for every worker  $i$  satisfies  $u_i \geq 0$  and the QoI-SRC auction is individual rational.  $\square$

Then, we analyze the algorithmic properties of the QoI-SRC auction including its computational complexity and approximation ratio to the optimal social welfare in Theorem 4 and 5.

**Theorem 4.** *The computational complexity of the QoI-SRC auction is  $O(N^2M)$ .*

*Proof.* The computational complexity of Algorithm 1 is dominated by the main loop, which terminates after  $N$  iterations in the worst case. In every iteration, the algorithm goes through every task  $\tau_j \in \mathcal{T}$ . Hence, the computational complexity of Algorithm 1 is  $O(NM)$ . Similarly, we have that the computational complexity of Algorithm 2 is  $O(N^2M)$ . Therefore, we conclude that computational complexity of the QoI-SRC auction is  $O(N^2M)$ .  $\square$

Then, we provide our analysis about the approximation ratio of the QoI-SRC auction using the method similar to the one proposed in [94]. In our following analysis, we use  $\mathcal{N}^-$  to denote all workers  $i \in \mathcal{N}$  with negative  $w_i$  and  $\vec{Q}^- = \{Q_1^-, \dots, Q_M^-\}$  to denote tasks' residual QoI requirement profile after Algorithm 1 includes all workers with  $w_i \geq 0$  into the winner set. Then, we normalize the  $w_i$  for every worker  $i \in \mathcal{N}^-$ , such that the normalized marginal social welfare  $w'_i = \frac{w_i}{\max_{n \in \mathcal{N}^-} w_n} > 0$ . Thus, with only a multiplicative factor change to the objective function, we formulate the linear program relaxation of the residual SRC-WD problem defined on worker set  $\mathcal{N}^-$  as the normalized primal linear program **P**. The dual program is formulated in program **D**.

$$\mathbf{P} : \min \sum_{i \in \mathcal{N}^-} w'_i x_i \quad (2.12)$$

$$\text{s.t.} \quad \sum_{i: \tau_j \in \Gamma_i, i \in \mathcal{N}^-} q_i x_i \geq Q_j^-, \quad \forall \tau_j \in \mathcal{T} \quad (2.13)$$

$$0 \leq x_i \leq 1, \quad \forall i \in \mathcal{N}^- \quad (2.14)$$

$$\mathbf{D} : \max \sum_{j: \tau_j \in \mathcal{T}} Q_j^- y_j - \sum_{i \in \mathcal{N}^-} z_i \quad (2.15)$$

$$\text{s.t.} \quad \sum_{j: \tau_j \in \Gamma_i} q_i y_j - z_i \leq w'_i, \quad \forall i \in \mathcal{N}^- \quad (2.16)$$

$$y_j \geq 0, \quad \forall \tau_j \in \mathcal{T} \quad (2.17)$$

$$z_i \geq 0, \quad \forall i \in \mathcal{N}^- \quad (2.18)$$

It is easily verifiable that the  $|\max_{i \in \mathcal{N}^-} w_i|$  multiplicative factor difference between the objective functions of  $\mathbf{P}$  and the SRC-WD problem does not affect the approximation ratio of Algorithm 1. Next, we introduce several notations and concepts utilized in our following analysis.

We define any task  $\tau_j \in \mathcal{T}$  as *alive* at any particular iteration of the main loop in Algorithm 1 if its QoI requirement is not fully satisfied. Furthermore, we define that task  $\tau_j$  is *covered* by  $\Gamma_i$  if  $\tau_j \in \Gamma_i$  and  $\tau_j$  is alive when worker  $i$  is selected. The coverage relationship is represented as  $\tau_j \preceq \Gamma_i$ . Then, we define the minimum measure of QoI as  $\Delta q$ , the *unit QoI*. Suppose when worker  $i$  is about to be selected, the residual QoI requirement profile is  $\vec{Q}' = \{Q'_1, \dots, Q'_M\}$  and  $\Gamma_i$  is the  $i_j$ th set that covers  $\tau_j$ , the corresponding normalized MSWE in terms of unit QoI can be represented in Equation (2.19).

$$W(\tau_j, i_j) = \frac{w'_i \Delta q}{\sum_{j: \tau_j \in \Gamma_i} \min\{Q'_j, q_i\}}. \quad (2.19)$$

We assume that  $\tau_j$  is covered by  $k_j$  sets and we have  $W(\tau_j, 1) \leq \dots \leq W(\tau_j, k_j)$  from Equation (2.19). Then, we define constants  $\theta = \max_{i,j} q_i |\Gamma_i| w'_j$  and  $m = \frac{1}{\Delta q} \sum_{j: \tau_j \in \mathcal{T}} Q_j^-$  that are used in the presentation of the following Lemma 2.

**Lemma 2.** *The following assignments of the variables  $y_j$  and  $z_i$  for  $\forall \tau_j \in \mathcal{T}$  and  $\forall i \in \mathcal{N}^-$  are feasible to  $\mathbf{D}$ .*

$$\begin{cases} y_j = \frac{W(\tau_j, k_j)}{2\theta H_m \Delta q}, \quad \forall \tau_j \in \mathcal{T} \\ z_i = \begin{cases} \frac{\sum_{j: \tau_j \preceq \Gamma_i} \left( \min\{Q'_j, q_i\} (W(\tau_j, k_j) - W(\tau_j, i_j)) \right)}{2\theta H_m \Delta q}, & \forall i \in \mathcal{S} \\ 0, & \forall i \notin \mathcal{S} \end{cases} \end{cases}.$$

*Proof.* Suppose for any worker  $i \in \mathcal{N}^-$ , there are  $t_i$  tasks in bundle  $\Gamma_i$ . We reorder these tasks in the order in which they are fully covered. If worker  $i$  is not a winner, we have  $z_i = 0$ . Suppose when the last unit QoI of  $\tau_j$  is about to be covered, the residual QoI requirement profile is  $\vec{Q}'' = \{Q''_1, \dots, Q''_M\}$ , then the total residual QoI of all the alive tasks contained by  $\Gamma_i$  is  $\sum_{h=j}^{t_i} \min\{Q''_h, q_i\}$ .

We have that

$$W(\tau_j, k_j) \leq \frac{w'_i \Delta q}{\sum_{h=j}^{t_i} \min\{Q''_h, q_i\}}.$$

Therefore, we have

$$\sum_{j=1}^{t_i} q_i y_j - z_i \leq \sum_{j=1}^{t_i} \frac{w'_i q_i}{2\theta H_m \sum_{h=j}^{t_i} \min\{Q'_h, q_i\}} - 0 \leq \frac{w'_i}{H_m} \left(1 + \frac{1}{2} + \dots + \frac{1}{m}\right) \leq w'_i$$

If worker  $i \in \mathcal{S}$ , then we assume that when worker  $i$  is selected as a winner,  $t'_i$  tasks in  $\Gamma_i$  have already been fully covered. We have

$$\begin{aligned} \sum_{j=1}^{t_i} q_i y_j - z_i &= \frac{\sum_{j=1}^{t_i} q_i W(\tau_j, k_j)}{2\theta H_m \Delta q} - \frac{\sum_{j=t'_i+1}^{t_i} \min\{Q'_j, q_i\} (W(\tau_j, k_j) - W(\tau_j, i_j))}{2\theta H_m \Delta q} \\ &= \frac{\sum_{j=1}^{t'_i} q_i W(\tau_j, k_j)}{2\theta H_m \Delta q} + \frac{\sum_{j=t'_i+1}^{t_i} \min\{Q'_j, q_i\} W(\tau_j, i_j)}{2\theta H_m \Delta q} + \frac{\sum_{j=t'_i+1}^{t_i} (q_i - \min\{Q'_j, q_i\}) W(\tau_j, k_j)}{2\theta H_m \Delta q} \\ &\leq \frac{\sum_{j=1}^{t'_i} \frac{q_i w'_i}{\sum_{h=j}^{t_i} \min\{Q'_h, q_i\}}}{2\theta H_m} + \frac{w'_i}{2\theta H_m} + \frac{\theta}{2\theta H_m} \leq w'_i \end{aligned}$$

Therefore, we arrive at the conclusion that the assignments of  $y_j$  and  $z_i$  in Lemma 2 are feasible to **D**.  $\square$

Then in Theorem 5, we present our result regarding the approximation ratio of Algorithm 1.

**Theorem 5.** *Algorithm 1 is a  $2\theta H_m$ -approximation algorithm for the residual SRC-WD problem defined on worker set  $\mathcal{N}^-$ .*

*Proof.* By substituting the dual assignments given in Lemma 2 into the objective function (2.15), we have

$$\begin{aligned} \sum_{j:\tau_j \in \mathcal{T}} Q_j^- y_j - \sum_{i \in \mathcal{N}^-} z_i &= \frac{\sum_{i \in \mathcal{N}^- \cap \mathcal{S}} \sum_{j:\tau_j \leq \Gamma_i} \left( \min\{Q'_j, q_i\} (W(\tau_j, i_j) - W(\tau_j, k_j)) \right)}{2\theta H_m \Delta q} + \frac{\sum_{j:\tau_j \in \mathcal{T}} Q_j^- W(\tau_j, k_j)}{2\theta H_m \Delta q} \\ &= \frac{\sum_{i \in \mathcal{N}^- \cap \mathcal{S}} \sum_{j:\tau_j \leq \Gamma_i} \min\{Q'_j, q_i\} \frac{w'_i \Delta q}{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_i\}}}{2\theta H_m \Delta q} = \frac{\sum_{i \in \mathcal{N}^- \cap \mathcal{S}} w'_i}{2\theta H_m} \end{aligned}$$

Because **D** is the dual program of **P**, we have

$$\frac{\sum_{i \in \mathcal{N}^- \cap \mathcal{S}} w'_i}{2\theta H_m} \leq \text{OPT}_{\mathbf{D}} \leq \text{OPT}_{\mathbf{P}} \leq \text{OPT}_{\text{SRC-WD}}.$$

Therefore, Algorithm 1 is a  $2\theta H_m$ -approximation algorithm for the residual SRC-WD problem

defined on worker set  $\mathcal{N}^-$ . □

Note that there exists a  $\max_{i \in \mathcal{N}} |\Gamma_i|$  factor in  $\theta$ , which could be large theoretically, and in worst case  $\max_{i \in \mathcal{N}} |\Gamma_i| = M$ . However, in practice, a worker typically has a limited capability and interest in terms of the number of tasks she can and wants to execute. Thus, practically, we have that  $\max_{i \in \mathcal{N}} |\Gamma_i| \ll M$ , which prevents the  $2\theta H_m$  approximation ratio derived in Theorem 5 from growing excessively large as  $M$  increases. Thus far, this is the best approximation ratio we have found, and we leave the proof of its tightness or the derivation of a better one, as well as the calculation of a lower bound for the ratio, in our future work.

## 2.4 MRC Auction

In this section, we present the mathematical formulation, mechanism design and the corresponding analysis for the MRC auction.

### 2.4.1 Mathematical Formulation

In the MRC auction, we also use the form of the platform's value function  $V_{\vec{q}}(\cdot)$  given in Equation (2.5). If the platform is given workers' cost function profile, denote as  $\vec{C} = \{C_1(\cdot), \dots, C_N(\cdot)\}$ , the MRC auction winner determination (MRC-WD) problem can be formulated as follows.

**MRC-WD Problem:**

$$\max \sum_{i \in \mathcal{N}} (\alpha q_i |\Gamma_i| - C_i(\Gamma_i)) x_i \tag{2.20}$$

$$\text{s.t. } \Gamma_i \subseteq \Gamma_i^j, \exists \Gamma_i^j \in \mathcal{T}_i, \quad \forall i \in \mathcal{N} \tag{2.21}$$

$$x_i \in \{0, 1\}, \quad \forall i \in \mathcal{N} \tag{2.22}$$

The MRC-WD problem takes the parameter  $\alpha$ , workers' QoI profile  $\vec{q}$  and workers' cost function profile  $\vec{C}$  as input. It has a set of binary variables  $\{x_1, \dots, x_n\}$  indicating whether worker  $i$  is selected in the winner set  $\mathcal{S}$ . That is, if  $i \in \mathcal{S}$ , then  $x_i = 1$ . Otherwise,  $x_i = 0$ .

Furthermore, for every worker  $i$ , we have a variable  $\Gamma_i$  indicating the set of sensing tasks that the platform allocates to this worker. Constraint (2.21) ensures that  $\Gamma_i$  is the subset of at least

one bundle  $\Gamma_i^j \in \mathcal{T}_i$ . Therefore, the MRC-WD problem aims to find the set of winners  $\mathcal{S}$  and the corresponding task allocation profile denoted as  $\vec{\Gamma} = \{\Gamma_1, \dots, \Gamma_N\}$  that maximize the social welfare represented by the objective function. We use  $\Gamma_{\max}^i$  to denote the bundle with the maximum cardinality in  $\mathcal{T}_i$  and  $w_{\max}^i = \alpha q_i |\Gamma_{\max}^i| - c_i$  to denote worker  $i$ 's marginal social welfare for  $\Gamma_{\max}^i$ . The maximum social welfare is achieved by selecting all workers with positive  $w_{\max}^i$  as winners and allocating to every winner  $i$  the set of tasks  $\Gamma_{\max}^i$ .

However, the challenge is that cost function profile  $\vec{C}$  is not known by the platform and we still aim to design a mechanism that approximately maximizes the social welfare with a guaranteed approximation ratio. Then, we present the design of our mechanism in Section 2.4.2 that achieves this objective while ensuring individual rationality and polynomial computational complexity.

## 2.4.2 Mechanism Design

Requiring truthfulness in multi-minded combinatorial auctions limits the family of mechanisms that can be used, as mentioned in [12]. As long as the mechanism can achieve close-to-optimal social welfare with a guaranteed approximation ratio, it is justifiable for us to relax the truthfulness requirement, as pointed out in [13]. In Algorithm 3 we describe our design of the iterative descending dominant-strategy *quality of information aware MRC (QoI-MRC)* auction which is different from the mechanisms designed for multi-minded forward combinatorial auctions proposed in [13, 125].

The QoI-MRC auction described in Algorithm 3 consists of a winner determination phase (line 1-18) and a pricing phase (line 19). Every winner  $i \in \mathcal{S}$  will be allocated her bidding bundle  $\Gamma_i$  and be paid her bidding price  $b_i$  of the final iteration of the winner determination phase. We assume that the platform has the information about the upper bound and lower bound of workers' costs denoted as  $c_{\max}$  and  $c_{\min}$  respectively. The platform initializes every worker  $i$ 's bidding bundle and bidding price as  $\Gamma_i = \emptyset$  and  $b_{\max} \geq c_{\max}$  (line 2). Moreover, the input parameters  $\beta > 1$  and  $\epsilon \in (0, c_{\min}]$ .

The main loop (line 3-17) is executed until every worker is either a winner or a loser. In every iteration of the main loop, every worker  $i$  such that  $\alpha q_i |\Gamma_i| - b_i \geq \epsilon$  is included in the winner set  $\mathcal{S}$  (line 5-6). For any worker  $i$  that is neither a winner nor a loser in the current iteration, the Algorithm gives her an option to choose whether she will enlarge her current bidding bundle  $\Gamma_i$  to

any bundle  $\Gamma'_i$  that contains  $\Gamma_i$  (line 8). If after the bundle enlarging  $\alpha q_i |\Gamma'_i| - b_i \geq \epsilon$  holds, this worker is included in the winner set (line 11-12). Otherwise, she is given the following two options to choose from.

---

**Algorithm 3:** QoI-MRC Auction

---

**Input:**  $\mathcal{N}, b_{\max}, \epsilon, \alpha, \beta, \vec{q}$ ;  
**Output:**  $\mathcal{S}, \vec{p}, \vec{\Gamma}$ ;  
// Winner determination  
// Initialize winner and loser sets  
1  $\mathcal{S} \leftarrow \emptyset, \mathcal{L} \leftarrow \emptyset$ ;  
// Initialize bidding bundles and prices  
2  $\vec{\Gamma} \leftarrow \{\emptyset, \dots, \emptyset\}, \vec{\Gamma}' \leftarrow \vec{\Gamma}, \vec{b} \leftarrow \{b_{\max}, \dots, b_{\max}\}$ ;  
// Main loop  
3 **while**  $\mathcal{S} \cup \mathcal{L} \neq \mathcal{N}$  **do**  
4     **foreach**  $i \in \mathcal{N} \setminus (\mathcal{S} \cup \mathcal{L})$  **do**  
5         **if**  $\alpha q_i |\Gamma_i| - b_i \geq \epsilon$  **then**  
6              $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$ ;  
7         **else**  
8             // Give worker  $i$  the option to enlarge her bidding bundle  
9             allow worker  $i$  to enlarge  $\Gamma_i$  to any  $\Gamma'_i$  s.t.  $\Gamma'_i \supseteq \Gamma_i$ ;  
10             // Update bidding bundle  
11             **if**  $\Gamma_i \neq \Gamma'_i$  **then**  
12                  $\Gamma_i \leftarrow \Gamma'_i$ ;  
13             **if**  $\alpha q_i |\Gamma_i| - b_i \geq \epsilon$  **then**  
14                  $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$ ;  
15         **foreach**  $i \in \mathcal{N} \setminus (\mathcal{S} \cup \mathcal{L})$  **do**  
16             // Give worker  $i$  two options  
17             option 1:  $b_i \leftarrow \frac{b_i}{\beta}$ ;  
18             option 2:  $b_i \leftarrow 0$ ;  
19             **if**  $b_i = 0$  **then**  
20                  $\mathcal{L} \leftarrow \mathcal{L} \cup \{i\}$ ;  
21  $\vec{\Gamma} \leftarrow \{\Gamma_i \in \vec{\Gamma}' | i \in \mathcal{S}\}$ ;  
// Pricing  
22  $\vec{p} \leftarrow \vec{b}$ ;  
23 **return**  $\mathcal{S}, \vec{p}, \vec{\Gamma}$ ;

---

- **Option 1** (line 14). By choosing option 1, worker  $i$  divides her bidding price  $b_i$  by  $\beta$ . As long as she is fully rational, she will choose option 1 rather than option 2 to drop out of the auction, if  $\frac{b_i}{\beta} > c_i$  hold. By doing so, she keeps herself in the auction and makes it still

possible for her to win in one of the future iterations to receive positive utility.

- **Option 2** (line 15). By choosing option 2, the worker  $i$  drops out of the auction. If  $\frac{b_i}{\beta} \leq c_i$ , any rational worker  $i$  will choose option 2 because it is impossible for her to obtain positive utility even though she remains in the auction in this case.

Finally, every winner  $i$  is allocated her bidding bundle  $\Gamma_i$  (line 18) and be paid her bidding price  $b_i$  (line 19) of the final iteration of the winner determination phase.

### 2.4.3 Analysis

Although the QoI-MRC auction cannot guarantee truthfulness because workers' bidding prices when Algorithm 3 terminates will possibly not be equal to workers' true costs, we show in the following Theorem 6 that every worker still has a dominant strategy.

**Theorem 6.** *Every worker  $i \in \mathcal{N}$  has the following dominant strategy in the QoI-MRC auction.*

- *Worker  $i$  enlarges bundle  $\Gamma_i$  to  $\Gamma_{\max}^i$  in the first iteration.*
- *When worker  $i$  is given the options to divide her bidding price  $b_i$  by  $\beta$  or drop out of the auction, she will always choose the former as long as  $\frac{b_i}{\beta} > c_i$  and the latter if  $\frac{b_i}{\beta} \leq c_i$ .*

*Proof.* Obviously, any rational worker  $i$  will choose to divide her current bidding price  $b_i$  by  $\beta$  as long as  $\frac{b_i}{\beta} > c_i$  when she is given the two options. By doing so, it is still possible for her to win the auction and be paid  $p_i > c_i$ . If  $\frac{b_i}{\beta} \leq c_i$ , then even if she wins the auction the payment  $p_i$  will not be larger than  $c_i$ . Hence, she will drop out in this case.

Then, we study whether any worker  $i$  will enlarge her bundle to some  $\Gamma'_i \neq \Gamma_{\max}^i$  in the first iteration.

- **Case 1** ( $\alpha q_i |\Gamma_{\max}^i| - b_{\max} > \alpha q_i |\Gamma'_i| - b_{\max} \geq \epsilon$ ). Both  $\Gamma_{\max}^i$  and  $\Gamma'_i$  will make the worker win the auction in the first iteration and be paid  $b_{\max}$ . We have  $u(\Gamma_{\max}^i) = u(\Gamma'_i)$ .
- **Case 2** ( $\alpha q_i |\Gamma_{\max}^i| - b_{\max} \geq \epsilon > \alpha q_i |\Gamma'_i| - b_{\max}$ ). The worker will win and be paid  $b_{\max}$  by enlarging to  $\Gamma_{\max}^i$  in the first iteration and we have  $u(\Gamma_{\max}^i) = b_{\max} - c_i$ . If she proposes  $\Gamma'_i$  instead of  $\Gamma_{\max}^i$ , she will be asked to decrease her bid or drop out in the first iteration.

Eventually, she could lose or win with being paid  $b'_i < b_{\max}$ . Her utility could either be  $u(\Gamma'_i) = 0$  or  $u(\Gamma'_i) = b'_i - c_i$ . We have  $u(\Gamma_{\max}^i) > u(\Gamma'_i)$ .

- **Case 3** ( $\epsilon > \alpha q_i |\Gamma_{\max}^i| - b_{\max} > \alpha q_i |\Gamma'_i| - b_{\max}$ ). Both  $\Gamma_{\max}^i$  and  $\Gamma'_i$  will make the worker face the choices of decreasing her bid or dropping out in the first iteration. If eventually she wins in both cases, then the number of iterations before she wins if she proposes  $\Gamma_{\max}^i$  will be smaller than or equal to that of  $\Gamma'_i$ . The payments  $p_i$  and  $p'_i$  for the two cases satisfy  $p_i \geq p'_i$  and we have  $u(\Gamma_{\max}^i) \geq u(\Gamma'_i)$ . If she loses in both cases, then  $u(\Gamma_{\max}^i) = u(\Gamma'_i) = 0$ . The last scenario is that she wins by proposing  $\Gamma_{\max}^i$  and loses by proposing  $\Gamma'_i$  in the first iteration. Then, we have  $u(\Gamma_{\max}^i) > 0 = u(\Gamma'_i)$ .

We have  $u(\Gamma_{\max}^i) \geq u(\Gamma'_i)$  with at least one scenario with strict inequality. Hence, worker  $i$  enlarges bundle  $\Gamma_i$  to  $\Gamma_{\max}^i$  in the first iteration. We arrive at the conclusion about any worker's dominant strategy stated in Theorem 6.  $\square$

**Theorem 7.** *The QoI-MRC auction is individual rational.*

*Proof.* When a worker is given the choices to decrease her bid or drops out of the auction, any worker  $i$  will drop out if  $\frac{b_i}{\beta} \leq c_i$ . She becomes a loser and obtains  $u_i = 0$ . The worker only chooses to divide  $b_i$  by  $\beta$  if  $\frac{b_i}{\beta} > c_i$ , which ensures that her payment  $p_i > c_i$  if she wins. In this case, we have  $u_i > 0$ . Therefore,  $u_i \geq 0$  and the QoI-MRC auction is individual rational.  $\square$

Then, we analyze the algorithmic properties of the QoI-MRC auction including its computational complexity and approximation ratio in Theorem 8 and 9.

**Theorem 8.** *The computational complexity of the QoI-MRC auction is  $O(N)$ .*

*Proof.* It is easily verifiable that the main loop of Algorithm 3 terminates after  $O(\log_{\beta} \frac{b_{\max}}{c_{\min}})$  number of iterations. The computational complexity inside the main loop is  $O(N)$ . Therefore, the computational complexity of the QoI-MRC auction is  $O(N)$ .  $\square$

In Theorem 9, we present our results about the approximation ratio of the QoI-MRC auction to the optimal social welfare. We use  $\mathcal{S}_{\text{OPT}}$  to denote the winner set of the optimal solution of the MRC-WD problem,  $q_{\max}$  to denote the maximum QoI in  $\vec{q}$  and  $\Gamma_{\max}$  to denote the maximum-cardinality bundle in  $\{\Gamma_{\max}^1, \dots, \Gamma_{\max}^N\}$ .

**Theorem 9.** *The approximation ratio of the QoI-MRC auction to the optimal social welfare is*

$$\frac{|\mathcal{S}|\epsilon}{|\mathcal{S}_{OPT}|(\alpha q_{\max}|\Gamma_{\max}| - c_{\min})}.$$

*Proof.* We use APP to denote the social welfare resulted by the QoI-MRC auction. From Theorem 6, every worker  $i \in \mathcal{N}$  enlarges her bundle to  $\Gamma_{\max}^i$  in the first iteration. The winner set  $\mathcal{S}$  output by Algorithm 3 consists of winners  $\mathcal{S}_1$  that win in the first iteration and  $\mathcal{S}_2$  that win in iteration  $r_i > 1$  with bidding price  $b_i^{r_i}$ . We have

$$\begin{aligned} \text{APP} &= \sum_{i \in \mathcal{S}} (\alpha q_i |\Gamma_{\max}^i| - c_i) \geq \sum_{i \in \mathcal{S}_1} (\alpha q_i |\Gamma_{\max}^i| - b_{\max}) + \sum_{i \in \mathcal{S}_2} (\alpha q_i |\Gamma_{\max}^i| - b_i^{r_i}) \\ &\geq |\mathcal{S}_1|\epsilon + |\mathcal{S}_2|\epsilon = |\mathcal{S}|\epsilon. \end{aligned}$$

Similarly, the optimal solution OPT is

$$\begin{aligned} \text{OPT} &= \sum_{i \in \mathcal{S}_{OPT}} (\alpha q_i |\Gamma_{\max}^i| - c_i) \leq |\mathcal{S}_{OPT}|(\alpha q_{\max}|\Gamma_{\max}| - c_{\min}) \\ &= \frac{|\mathcal{S}_{OPT}|(\alpha q_{\max}|\Gamma_{\max}| - c_{\min})}{|\mathcal{S}|\epsilon} \cdot |\mathcal{S}|\epsilon \leq \frac{|\mathcal{S}_{OPT}|(\alpha q_{\max}|\Gamma_{\max}| - c_{\min})}{|\mathcal{S}|\epsilon} \cdot \text{APP}. \end{aligned}$$

Therefore, the approximation ratio of the QoI-MRC auction to the optimal social welfare is

$$\frac{|\mathcal{S}|\epsilon}{|\mathcal{S}_{OPT}|(\alpha q_{\max}|\Gamma_{\max}| - c_{\min})}. \quad \square$$

## 2.5 Performance Evaluation

In this section, we introduce the baseline methods, as well as simulation settings and results.

### 2.5.1 Baseline Method

The first baseline approach is a modified version of the traditional VCG auction [23, 41]. We integrate the concept of QoI and the QoI coverage constraint defined in Section 2.3 into the VCG winner determination (VCG-WD) problem. We call the modified VCG auction *quality of information aware VCG (QoI-VCG)* auction, in which the VCG-WD problem is solved optimally and the VCG pricing mechanism [23, 41] is utilized to derive winners' payments.

Another baseline method is the marginal social welfare greedy (MSW-Greedy) auction. Its winner determination algorithm firstly includes every worker  $i$  with  $w_i \geq 0$  into the winner set.

Then, it selects the worker with the largest marginal social welfare among the remaining workers in every iteration until tasks' QoI requirements are fully satisfied. The pricing mechanism is similar to Algorithm 2 which essentially pays every winner her supremum bidding price to win given her current bidding bundle. It is easily verifiable that the MSW-Greedy auction is truthful and individual rational.

### 2.5.2 Simulation Settings

Setting	$\alpha$	$c_i$	$q_i$	$Q_j$	$ \bar{\Gamma}_i $	$N$	$M$
2.I	0.1	[2, 4]	[1, 2]	[10, 13]	[20, 30]	[200, 500]	100
2.II	0.1	[4, 8]	[2, 4]	[10, 13]	[20, 30]	300	[300, 600]

Table 2.5: Simulation setting 2.I and 2.II

For our simulation of the SRC auction, we consider the two settings described in Table 2.5. In setting 2.I, we fix the number of tasks as  $M = 100$  and vary the number of workers from 200 to 500. In setting 2.II, we fix the number of workers as  $N = 300$  and vary the number of tasks from 300 to 600. The parameter  $\alpha = 0.1$  in both settings and the values of  $c_i$ ,  $q_i$ ,  $|\bar{\Gamma}_i|$  for any worker  $i \in \mathcal{N}$  and  $Q_j$  for any task  $\tau_j \in \mathcal{T}$  are generated uniformly at random from the ranges given in Table 2.5. Worker  $i$ 's maximum executable task set  $\bar{\Gamma}_i$  consists of  $|\bar{\Gamma}_i|$  tasks selected uniformly at random from  $\mathcal{T}$ . Furthermore, the optimal solution to the VCG-WD problem of the QoI-VCG mechanism is calculated using the GUROBI optimization solver [6].

Setting	$\alpha$	$b_{\max}$	$c_i$	$q_i$	$ \bar{\Gamma}_i $	$N$	$M$
2.III	0.2	100	[4, 6]	[1, 2]	[20, 30]	[200, 500]	100
2.IV	0.2	100	[6, 10]	[2, 4]	[20, 30]	300	[200, 400]

Table 2.6: Simulation setting 2.III and 2.IV

For our simulation of the MRC auction, we consider the two settings described in Table 2.6. In setting 2.III, we fix the number of tasks as  $M = 100$  and vary the number of workers from 200 to 500. In setting 2.II, we fix the number of workers as  $N = 300$  and vary the number of tasks from 200 to 400. The parameters  $\alpha = 0.2$  and  $b_{\max} = 0.2$  in both settings and the values of  $c_i$ ,  $q_i$ ,  $|\bar{\Gamma}_i|$  for any worker  $i \in \mathcal{N}$  are generated uniformly at random from the ranges given in Table 2.6. Worker  $i$ 's maximum executable task set  $\bar{\Gamma}_i$  consists of  $|\bar{\Gamma}_i|$  tasks selected uniformly at random from  $\mathcal{T}$ . Worker  $i$ 's interested bundle set consists of randomly selected subsets of  $\bar{\Gamma}_i$  whose union

is  $\bar{\Gamma}_i$ . Note that we leave the study of the values of these parameters in real-world applications in our future work.

### 2.5.3 Simulation Results

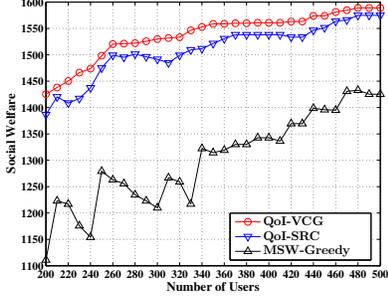


Figure 2.3: Social welfare (setting 2.I)

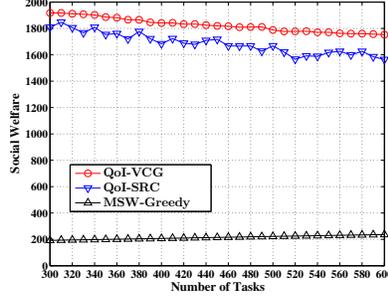


Figure 2.4: Social welfare (setting 2.II)

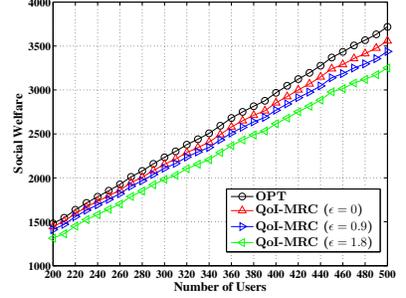


Figure 2.5: Social welfare with varying  $\epsilon$  (setting 2.III)

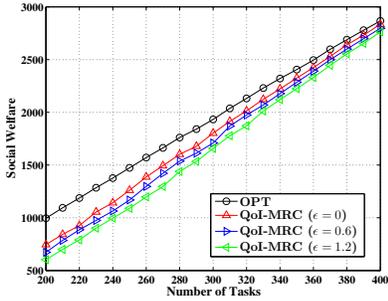


Figure 2.6: Social welfare with varying  $\epsilon$  (setting 2.IV)

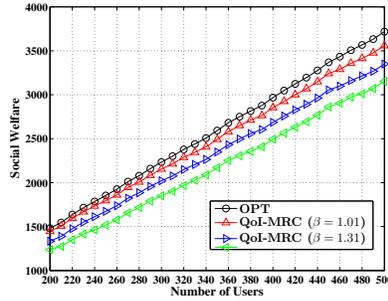


Figure 2.7: Social welfare with varying  $\beta$  (setting 2.III)

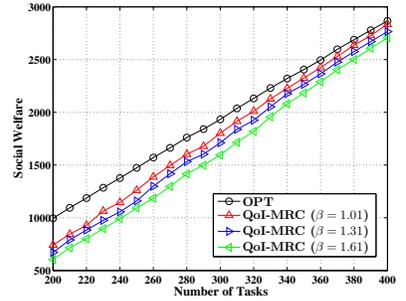


Figure 2.8: Social welfare with varying  $\beta$  (setting 2.IV)

In Figure 2.3 and 2.4, we compare the social welfare generated by the QoI-VCG auction, the QoI-SRC auction and the MSW-Greedy auction. The social welfare of the QoI-VCG auction equals to the optimal solution of the SRC-WD problem. From Figure 2.3 and 2.4, we arrive at the conclusion that the social welfare of the QoI-SRC auction is close to optimal and far better than that of the baseline MSW-Greedy auction.

$N$	200	220	240	260	280	300	320	340
QoI-VCG	10.19	16.06	11.22	11.71	58.64	63.14	79.37	10.51
QoI-SRC	0.019	0.014	0.015	0.015	0.020	0.022	0.018	0.019
$N$	360	380	400	420	440	460	480	500
QoI-VCG	43.52	93.44	94.25	273.6	52.54	72.26	860.9	2043
QoI-SRC	0.019	0.021	0.021	0.019	0.023	0.021	0.021	0.024

Table 2.7: Execution time (seconds) for setting 2.I

In Table 2.7 and 2.8, we show the comparison of the execution time of the QoI-VCG and QoI-

SRC auctions. It is obvious from these two tables that the QoI-SRC auction executes in significantly less time than the QoI-VCG auction. With the increasing of the number of workers and tasks, the execution time of the QoI-VCG auction gradually becomes so long that makes it infeasible to be utilized in practice. In contrast, the QoI-SRC auction keeps low execution time regardless of the growth of the worker and task numbers. The QoI-SRC auction is much more computationally efficient than the QoI-VCG auction.

$M$	300	320	340	360	380	400	420	440
QoI-VCG	18.70	1.337	2.715	15.47	21.42	43.38	88.57	224.3
QoI-SRC	0.066	0.076	0.075	0.076	0.073	0.090	0.075	0.077
$M$	460	480	500	520	540	560	580	600
QoI-VCG	67.85	50.68	183.5	229.3	474.8	751.1	1206	1269
QoI-SRC	0.079	0.117	0.099	0.130	0.111	0.122	0.123	0.147

Table 2.8: Execution time (seconds) for setting 2.II

In Figure 2.5 and 2.6, we compare the social welfare generated by the QoI-MRC auction with the optimal social welfare in both setting 2.III and 2.IV. We fix the parameter  $\beta = 1.01$  and vary the choices of  $\epsilon$ . From the two figures, we observe that the QoI-MRC auction obtains close-to-optimal social welfare and it becomes closer to the optimal social welfare when  $\epsilon$  approaches 0.

In Figure 2.7 and 2.8, we fix the parameter  $\epsilon = 0.01$  and vary the choices of  $\beta$ . From these two figures, we also observe that the QoI-MRC auction obtains close-to-optimal social welfare and as  $\beta$  approaches 1, it becomes closer to the optimal social welfare.

## 2.6 Related Work

Game theory has been widely utilized to tackle networking problems such as spectrum sharing [18, 26, 47, 49, 55, 125], cooperative communication [19], channel and bandwidth allocation [25, 114], and so forth. Similar to many other problems, when it comes to incentive mechanism design in MCS systems, game-theoretic models are also frequently adopted by researchers due to their ability to capture and tackle workers' strategic behaviors.

Thus far, a series of game-theoretic incentive mechanisms [20, 22, 27, 28, 31–33, 35, 37, 43, 44, 51, 53, 54, 60, 62, 71–75, 82, 91, 92, 97, 98, 100, 106, 107, 110–113, 115–124] have been proposed by the research community. Among them, one major category [32, 33, 35, 37, 51, 53, 54, 60, 62, 71, 72, 75, 97, 100, 106, 107, 112, 113, 115–118, 120–122, 124] are based on reverse auctions, whereas others

adopt other game-theoretic models, including Stackelberg game [20, 27, 28, 112, 113], reputation mechanism [82, 110, 111, 119], peer truth serum [31, 91, 92], task selection game [22], contest model [73, 74], posted price mechanism [43, 44, 123], as well as market model [98]. Apart from the shared goal of incentivizing worker participation, these mechanisms optimize, as well, various system-wide objectives, including maximizing the social welfare [22, 33, 37, 54, 107, 119–121, 124] or the platform’s profit [20, 27, 28, 35, 62, 72–75, 82, 106, 112, 113, 116–118, 123], and minimizing the social cost [32, 71] or the platform’s total payment [43, 44, 51, 53, 60, 100, 110, 111, 115, 122, 123].

However, a common feature of most of the existing incentive mechanisms is that workers’ QoI is not incorporated into the designs. In contrast, we consider workers’ QoI, and treat it as a crucial parameter in our mechanisms, which distinguishes our work with most of the existing ones.

Although some existing incentive mechanisms also consider workers’ QoI, our mechanisms are different with them in various aspects. One line of prior work [20, 60, 72–75] explore the relationship between workers’ QoI and the levels of their sensing effort. These work invariably assume the existence of *a priori* known distributions of workers’ sensing costs, whereas we do not make such assumption. The QoI aware dynamic participant selection protocols proposed in [45, 95] do not utilize game theoretic frameworks, and thus cannot handle workers’ strategic behavior. Note that QoI awareness is much more commonly adopted in recent incentive mechanisms [20, 35, 51, 53, 54, 82, 88, 98, 100, 107, 115, 116, 123] developed after our work [50], which is among the first ones that consider this issue. Technically, we adopt auction, a branch of game theory that has been extensively studied by not only economists, but also computer scientists, as the fundamental framework of our incentive mechanisms. We argue that the various auction models [12, 13, 15, 23, 40, 41, 76, 81] developed over the past several decades usually cannot meet the specific needs of effectively incentivizing worker participation in MCS systems, and thus cannot be readily applied in our scenario. On one hand, none of the existing auctions consider workers’ QoI. On the other hand, in terms of mathematical formulation, few of them work in problem settings with coverage (like) constraints.

## 2.7 Conclusion

In this chapter, we design QoI aware incentive mechanisms for MCS systems based on RC auctions. For the SRC auction, we design a truthful, individual rational and computationally efficient mech-

anism that approximately maximizes the social welfare with a guaranteed approximation ratio. For the MRC auction, we design an iterative descending mechanism that achieves close-to-optimal social welfare with a guaranteed approximation ratio while satisfying individual rationality and computational efficiency. Also, we validate our theoretical analysis through extensive simulations.

## Chapter 3

# Incentivizing Multi-Requester Mobile Crowd Sensing

### 3.1 Introduction

As previously mentioned, aware of the paramount importance of stimulating user participation in MCS systems, a series of incentive mechanisms [20, 22, 27, 28, 31–33, 35, 37, 43–45, 50, 51, 53, 57, 58, 60, 62, 66, 67, 71–75, 82, 86, 88, 89, 91, 92, 95, 97, 98, 100, 106, 107, 110–113, 115–124] have been recently developed by the research community. However, most of the existing incentive mechanisms assume that there is only one data requester who also serves as the platform in the MCS system. In practice, however, there are usually *multiple data requesters* competing for human resources, who usually outsource worker recruiting to third-party platforms (e.g., Amazon Mechanical Turk [1], Clickworker [2]) that have already gathered a large number of workers. Therefore, in this chapter, we focus on such MCS systems where three parties, including the data requesters, a platform (i.e., a cloud-based central server), as well as a crowd of participating workers co-exist, and aim to develop *a new incentive mechanism that can decide which worker serves which data requester at what price.*

In real practice, the sensory data provided by individual workers are usually quite unreliable due to various factors (e.g., poor sensor quality, lack of sensor calibration, environment noise). Hence, in order to cancel out the possible errors from individual workers, it is highly necessary that the platform utilizes a *data aggregation mechanism* to properly aggregate their noisy and even conflicting data. In an MCS system, the incentive and the data aggregation mechanism are usually not isolated from each other. In fact, the data aggregation mechanism typically interacts with the incentive mechanism, and thus, affects its design and performance. Intuitively, if the platform aggregates workers' data in naive ways (e.g., voting and average) that treat all workers' data equally, the incentive mechanism does not need to distinguish them with respect to their reliability. However, a weighted aggregation method that puts higher weights on more reliable

workers is much more desirable, because it shifts the aggregated results towards the data provided by the workers with higher reliability. Accordingly, the incentive mechanism should also incorporate workers' reliability, and selects workers that are more likely to provide reliable data.

Therefore, different from most of the aforementioned existing work [20,22,27,28,31–33,35,37,43–45,50,51,53,54,57,58,60,62,66,67,71–75,82,86,88,89,91,92,95,97,98,100,106,107,110–113,115–124], we propose CENTURION<sup>1</sup>, a novel *integrated framework* for *multi-requester* MCS systems, which consists of a *weighted data aggregation mechanism* that considers workers' diverse reliability in the calculation of the aggregated results, together with an incentive mechanism that selects workers who potentially will provide more reliable data. Specifically, CENTURION's incentive mechanism is based on *double auction* [76], which involves auctions among not only the workers, but also the data requesters, and is able to incentivize the participation of both data requesters and workers. This chapter makes the following contributions.

- Different from existing work, we propose a novel *integrated framework* for *multi-requester* MCS systems, called CENTURION, consisting of a data aggregation and an incentive mechanism. Such an integrated design, which captures the interactive effects between the two mechanisms, is much more complicated and challenging than designing them separately.
- CENTURION's double auction-based incentive mechanism is able to incentivize the participation of both data requesters and workers, and bears many desirable properties, including *truthfulness*, *individual rationality*, *computational efficiency*, as well as *non-negative social welfare*.
- The data aggregation mechanism of CENTURION takes into consideration workers' reliability, and calculates *highly accurate* aggregated results.

In the rest of this chapter, we first introduce the preliminaries in Section 3.2. Then, the design details of CENTURION's data aggregation and incentive mechanism are described in Section 3.3. In Section 3.4, we conduct extensive simulations to validate the desirable properties of CENTURION. Next, we discuss the past literature that are related to this work in Section 3.5. Finally, in Section 3.6, we conclude this chapter.

---

<sup>1</sup>The name CENTURION comes from inCENTivizing mUlti-Requester mobile crOwd seNsing.

## 3.2 Preliminaries

In this section, we introduce the system overview, reliability level model, auction model, as well as the design objectives.

### 3.2.1 System Overview

CENTURION is an MCS system framework consisting of a cloud-based platform, a set of participating workers, denoted as  $\mathcal{W} = \{w_1, \dots, w_N\}$ , and a set of requesters, denoted as  $\mathcal{R} = \{r_1, \dots, r_M\}$ . Each requester  $r_j \in \mathcal{R}$  has a sensing task  $\tau_j$  to be executed by the workers. The set of all requesters' tasks is denoted as  $\mathcal{T} = \{\tau_1, \dots, \tau_M\}$ . We are specifically interested in the scenario where  $\mathcal{T}$  is a set of  $M$  different *binary classification tasks* that require workers to locally decide the classes of the events or objects, and report to the platform their local decisions (i.e., the labels of the observed events or objects). Such MCS systems, collecting binary labels from the crowd, constitute a large portion of the currently deployed MCS systems (e.g., congestion detection systems that decide whether or not particular road segments are congested [99], geotagging campaigns that tag whether bumps or potholes exist on specific segments of road surface [30, 80]).

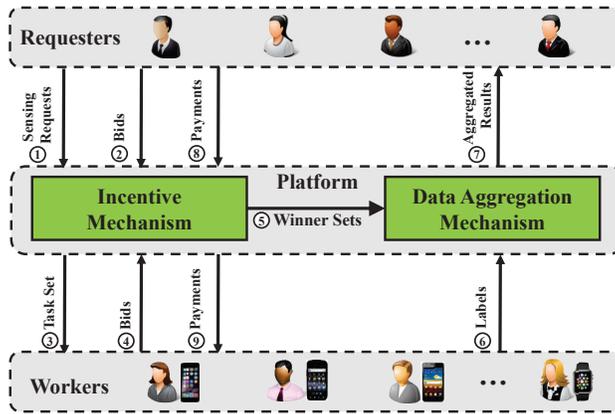


Figure 3.1: Framework of CENTURION<sup>2</sup>

Each task  $\tau_j$  has a true label  $l_j \in \{-1, +1\}$ , unknown to the requesters, the platform, and the workers. If a worker  $w_i$  is chosen to execute task  $\tau_j$ , she will provide to the platform a label  $l_{i,j}$ . We define  $\mathbf{l} = [l_{i,j}] \in \{-1, +1, \perp\}^{N \times M}$  as the matrix containing all workers' labels, where  $l_{i,j} = \perp$  means that task  $\tau_j$  is not executed by worker  $w_i$ . For every task  $\tau_j$ , the platform aggregates workers'

<sup>2</sup>In this figure, circled numbers represent the order of the events.

labels into an aggregated result, denoted as  $\widehat{l}_j$ , so as to cancel out the errors from individual workers. The framework of CENTURION is given in Figure 3.1, and we describe its workflow<sup>3</sup> as follows.

- **Incentive Mechanism.** Firstly, in the double auction-based incentive mechanism, each requester  $r_j$  submits to the platform a sensing request containing the sensing task  $\tau_j$  to be executed (step ①), and a bid  $a_j$ , the amount she is willing to pay if the task is executed (step ②). Then, the platform announces the set of sensing tasks  $\mathcal{T}$  to the workers (step ③). After receiving the task set, every worker  $w_i$  sends to the platform the set of tasks she wants to execute, denoted as  $\Gamma_i \subseteq \mathcal{T}$ , as well as a bid  $b_i$ , which is her bidding price for executing them (step ④). Based on received bids, the platform determines the set of winning requesters  $\mathcal{S}_{\mathcal{R}}$ , the set of winning workers  $\mathcal{S}_{\mathcal{W}}$ , as well as the payment  $p_j^r$  charged from every winning requester  $r_j$  and the payment  $p_i^w$  paid to every winning worker  $w_i$  (step ⑤). Note that losing requesters' tasks are not executed, and thus, they do not submit any payment. Similarly, losing workers do not receive any payment, as they do not execute any task.
- **Data Aggregation Mechanism.** Next, the platform collects the labels submitted by the winning workers (step ⑥), calculates the aggregated results, and sends them to the winning requesters (step ⑦).
- Finally, the platform charges  $p_j^r$  from winning requester  $r_j$  (step ⑧), and pays  $p_i^w$  to winning worker  $w_i$  (step ⑨).

We denote the requesters' and workers' bid profile as  $\mathbf{a} = (a_1, \dots, a_M)$  and  $\mathbf{b} = (b_1, \dots, b_N)$ , respectively. Moreover, the requesters' and workers' payment profile is denoted as  $\mathbf{p}^r = (p_1^r, \dots, p_M^r)$  and  $\mathbf{p}^w = (p_1^w, \dots, p_N^w)$ , respectively.

### 3.2.2 Reliability Level Model

Before worker  $w_i$  executes task  $\tau_j$ , her label about this task can be regarded as a random variable  $L_{i,j}$ . Then, we define the *reliability level* of a worker in Definition 8.

---

<sup>3</sup>Note that we are specifically interested in the scenario where all workers and requesters arrive at same time. We leave the investigation of the online scenario where workers and requesters arrive sequentially in an online manner in our future work.

**Definition 8** (Reliability Level). *A worker  $w_i$ 's reliability level  $\theta_{i,j}$  about task  $\tau_j$  is defined as the probability that she provides a correct label about this task, i.e.,*

$$\theta_{i,j} = \Pr[L_{i,j} = l_j] \in [0, 1]. \quad (3.1)$$

Moreover, we denote the workers' reliability level matrix as  $\boldsymbol{\theta} = [\theta_{i,j}] \in [0, 1]^{N \times M}$ .

We assume that the platform knows the reliability level matrix  $\boldsymbol{\theta}$  *a priori*, and maintains a historical record of it. In practice, the platform could obtain  $\boldsymbol{\theta}$  through various approaches. For example, as, in many scenarios, workers tend to have similar reliability levels for similar tasks, the platform could assign to workers some tasks with known labels, and use workers' labels about these tasks to estimate their reliability levels for similar tasks as in [87]. In cases where ground truth labels are not available,  $\boldsymbol{\theta}$  can still be effectively inferred from workers' characteristics (e.g., the prices of a worker's sensors, a worker's experience and reputation for similar tasks) using the algorithms proposed in [63], or estimated using the labels previously submitted by workers about similar tasks by the methods in [68–70, 78, 96, 101].

### 3.2.3 Auction Model

In this paper, we consider the scenario where both requesters and workers are *strategic* and *selfish* that aim to maximize their own utilities. Since CENTURION involves auctions among not only the workers, but also the requesters, we utilize the following *double auction for Multi-rEquester mobiLe crowd seNsing (MELON double auction)*, formally defined in Definition 9, as the incentive mechanism.

**Definition 9** (MELON Double Auction). *In a double auction for multi-requester mobile crowd sensing (MELON double auction), each requester  $r_j$  obtains a value  $v_j$ , if her task  $\tau_j$  is executed, and bids to the platform  $a_j$ , the amount she is willing to pay for the execution of her task. Each worker  $w_i$  is interested in executing one subset of the tasks, denoted as  $\Gamma_i \subseteq \mathcal{T}$ , and bids to the platform  $b_i$ , her bidding price for executing these tasks. Her actual sensing cost for executing all tasks in  $\Gamma_i$  is denoted as  $c_i$ . Both the requesters' values and workers' costs are unknown to the platform.*

Then, we define a requester's and worker's utility, as well as the platform's profit in Definition 10, 11, and 12.

**Definition 10** (Requester's Utility). *A requester  $r_j$ 's utility is defined as*

$$u_j^r = \begin{cases} v_j - p_j^r, & \text{if } r_j \in \mathcal{S}_{\mathcal{R}} \\ 0, & \text{otherwise} \end{cases}. \quad (3.2)$$

**Definition 11** (Worker's Utility). *A worker  $w_i$ 's utility is defined as*

$$u_i^w = \begin{cases} p_i^w - c_i, & \text{if } w_i \in \mathcal{S}_{\mathcal{W}} \\ 0, & \text{otherwise} \end{cases}. \quad (3.3)$$

**Definition 12** (Platform's Profit). *The profit of the platform is defined as*

$$u_0 = \sum_{j:r_j \in \mathcal{S}_{\mathcal{R}}} p_j^r - \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}} p_i^w. \quad (3.4)$$

Based on Definition 10, 11, and 12, we define the social welfare of the MCS system in Definition 13.

**Definition 13** (Social Welfare). *The social welfare of the MCS system is defined as*

$$u_{\text{social}} = u_0 + \sum_{i:w_i \in \mathcal{W}} u_i^w + \sum_{j:r_j \in \mathcal{R}} u_j^r = \sum_{j:r_j \in \mathcal{S}_{\mathcal{R}}} v_j - \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}} c_i. \quad (3.5)$$

Clearly, the social welfare is the sum of the platform's profit and all requesters' and workers' utilities.

### 3.2.4 Design Objectives

In this chapter, we aim to ensure that CENTURION bears the following advantageous properties.

Since the requesters are strategic and selfish in our model, it is possible that any requester  $r_j$  submits a bid  $a_j$  that deviates from  $v_j$  (i.e., her value for task  $\tau_j$ ). Similarly, any worker  $w_i$  might also submit a bid  $b_i$  that differs from  $c_i$  (i.e., her cost for executing all tasks in  $\Gamma_i$ ). Thus, one of our objectives is to design a *truthful* incentive mechanism defined in Definition 14.

**Definition 14** (Truthfulness). *A MELON double auction is truthful if and only if bidding  $v_j$  and  $c_i$  is the dominant strategy for each requester  $r_j$  and worker  $w_i$ , i.e., bidding  $v_j$  and  $c_i$  maximizes, respectively, the utility of each requester  $r_j$  and worker  $w_i$ , regardless of other requesters' and workers' bids.*

By Definition 14, we aim to ensure that both requesters and workers bid truthfully to the platform. Apart from truthfulness, another desirable property that we aim to achieve is *individual rationality* defined in Definition 15.

**Definition 15** (Individual Rationality). *A MELON double auction is individual rational if and only if no requesters or workers receive negative utilities, i.e., we have  $u_j^r \geq 0$ , and  $u_i^w \geq 0$ , for every requester  $r_j$  and worker  $w_i$ , respectively.*

Individual rationality is a crucial property to stimulate the participation of both requesters and workers, because it ensures that the charge to a requester is no larger than her value, and a worker's sensing cost is also totally compensated. As mentioned in Section 3.2.1, CENTURION aggregates workers' labels to ensure that the aggregated results have satisfactory accuracy, which is mathematically defined in Definition 16.

**Definition 16** ( $\beta_j$ -Accuracy). *A task  $\tau_j$  is executed with  $\beta_j$ -accuracy if and only if  $\Pr[\widehat{L}_j \neq l_j] \leq \beta_j$ , where  $\beta_j \in (0, 1)$ , and  $\widehat{L}_j$  denotes the random variable representing the aggregated result for task  $\tau_j$ .*

By Definition 16,  $\beta_j$ -accuracy ensures that the aggregated result equals to the true label with high probability. Note that, for every task  $\tau_j$ ,  $\beta_j$  is a parameter chosen by the platform, and a smaller  $\beta_j$  implies a stronger requirement for the accuracy.

In short, our objectives are to ensure that the proposed CENTURION framework provides satisfactory *accuracy guarantee* for the aggregated results of all executed tasks, and incentivizes the participation of both requesters and workers in a *truthful* and *individual rational* manner.

### 3.3 Design Details

In this section, we present the design details of the incentive and data aggregation mechanism of CENTURION.

### 3.3.1 Data Aggregation Mechanism

In this section, we introduce the design details of CENTURION’s data aggregation mechanism, as well as the corresponding analyses.

#### 3.3.1.1 Proposed Mechanism

Although the data aggregation mechanism follows the incentive mechanism in CENTURION’s workflow, we introduce it first, as it affects the design of the incentive mechanism.

In order to capture the effect of workers’ diverse reliability on the calculation of the aggregated results, CENTURION adopts the following *weighted aggregation method*. That is, the aggregated result  $\hat{l}_j$  for every executed task  $\tau_j$  is calculated as

$$\hat{l}_j = \text{sign} \left( \sum_{i:w_i \in \mathcal{S}_W, \tau_j \in \Gamma_i} \lambda_{i,j} l_{i,j} \right), \quad (3.6)$$

where  $\lambda_{i,j} > 0$  is worker  $w_i$ ’s weight on task  $\tau_j$ . Furthermore, the function  $\text{sign}(x)$  equals to  $+1$ , if  $x \geq 0$ , and  $-1$  otherwise.

Intuitively, higher weights should be assigned to workers who are more likely to submit correct labels, which makes the aggregated results closer to the labels provided by more reliable workers. In fact, many state-of-the-art literature [68–70, 78, 96, 101] utilize such weighted aggregation method to aggregate workers’ data. As the weight  $\lambda_{i,j}$ ’s highly affect the accuracy of the aggregated results, we propose, in the following Algorithm 4, the data aggregation mechanism of CENTURION.

---

**Algorithm 4:** Data Aggregation Mechanism

---

**Input:**  $\theta, \mathbf{l}, \Gamma, \mathcal{S}_R, \mathcal{S}_W$ ;  
**Output:**  $\{\hat{l}_j | r_j \in \mathcal{S}_R\}$ ;  
**1** **foreach**  $j$  *s.t.*  $r_j \in \mathcal{S}_R$  **do**  
**2**    $\hat{l}_j \leftarrow \text{sign} \left( \sum_{i:w_i \in \mathcal{S}_W, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1) l_{i,j} \right)$ ;  
**3** **return**  $\{\hat{l}_j | r_j \in \mathcal{S}_R\}$ ;

---

Algorithm 4 takes as inputs the reliability level matrix  $\theta$ , the workers’ label matrix  $\mathbf{l}$ , the profile of workers’ interested task sets, denoted as  $\Gamma = (\Gamma_1, \dots, \Gamma_N)$ , the winning requester set  $\mathcal{S}_R$ , and the winning worker set  $\mathcal{S}_W$ . Note that a large  $\theta_{i,j}$  indicates that a worker  $w_i$  has a high reliability

level for task  $\tau_j$ , and any worker  $w_i$  with  $\theta_{i,j} \leq 0.5$  will not be selected as a winner by the incentive mechanism. The aggregated result  $\widehat{l}_j$  for each winning requester  $r_j$ 's task  $\tau_j$  is calculated (line 1-2) using Equation (3.6) with the weight

$$\lambda_{i,j} = 2\theta_{i,j} - 1, \quad \forall r_j \in \mathcal{S}_{\mathcal{R}}, \quad w_i \in \mathcal{S}_{\mathcal{W}}, \quad \tau_j \in \Gamma_i. \quad (3.7)$$

By Equation (3.7), we have that  $\lambda_{i,j}$ , i.e., worker  $w_i$ 's weight for task  $\tau_j$ , increases with  $\theta_{i,j}$ , which conforms to our intuition that the higher the probability that worker  $w_i$  provides a correct label about task  $\tau_j$ , the more her label  $l_{i,j}$  should be counted in the calculation of the aggregated result about this task. We provide the formal analysis about the data aggregation mechanism in Section 3.3.1.2.

### 3.3.1.2 Analysis

In this section, we firstly analyze Algorithm 4's guarantee of aggregation accuracy for each executed task. In the following Lemma 3, we establish an upper bound for the accuracy of the aggregated result  $\widehat{l}_j$  of each executed task  $\tau_j$  compared to its truth label  $l_j$ .

**Lemma 3.** *For each executed task  $\tau_j$ , given the winning worker set  $\mathcal{S}_{\mathcal{W}}$ , the reliability level matrix  $\theta$ , as well as workers' weights  $\lambda_{i,j}$ 's on this task, we have that*

$$\Pr[\widehat{L}_j \neq l_j] \leq \exp\left(-\frac{(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j} (2\theta_{i,j} - 1))^2}{2 \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right) \quad (3.8)$$

by aggregating workers' data according to Equation (3.6).

*Proof.* We denote  $X_{i,j}$  as the random variable for worker  $w_i$ 's weighted label about task  $\tau_j$ , i.e.,  $X_{i,j} = \lambda_{i,j}l_j$  with probability  $\theta_{i,j}$ , and  $X_{i,j} = -\lambda_{i,j}l_j$  with probability  $1 - \theta_{i,j}$ . Then, for each task  $\tau_j$ , we define

$$X_j = \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} X_{i,j},$$

and thus, we have that

$$\mathbb{E}[X_j] = \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \mathbb{E}[X_{i,j}] = \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} l_j \lambda_{i,j} (2\theta_{i,j} - 1).$$

Based on the Hoeffding bound, we have that

$$\begin{aligned}\Pr[X_j < 0 | l_j = +1] &= \Pr[\mathbb{E}[X_j] - X_j > \mathbb{E}[X_j] | l_j = +1] \leq \exp\left(-\frac{2(\mathbb{E}[X_j | l_j = +1])^2}{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\lambda_{i,j})^2}\right) \\ &= \exp\left(-\frac{(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j} (2\theta_{i,j} - 1))^2}{2 \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right).\end{aligned}$$

Similarly, we have that

$$\Pr[X_j \geq 0 | l_j = -1] \leq \exp\left(-\frac{(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j} (2\theta_{i,j} - 1))^2}{2 \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right).$$

As the error probability of the aggregated result can be calculated as  $\Pr[\widehat{L}_j \neq l_j] = \Pr[X_j < 0 | l_j = +1] \Pr[l_j = +1] + \Pr[X_j \geq 0 | l_j = -1] \Pr[l_j = -1]$ , we have that

$$\Pr[\widehat{L}_j \neq l_j] \leq \exp\left(-\frac{(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j} (2\theta_{i,j} - 1))^2}{2 \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right),$$

which exactly proves Lemma 3.  $\square$

Clearly, Lemma 3 gives us an upper bound for the probability  $\Pr[\widehat{L}_j \neq l_j]$  for each executed task  $\tau_j$ . Then, in the following Theorem 10, we will prove that this upper bound is minimized by our data aggregation mechanism proposed Algorithm 4.

**Theorem 10.** *For each executed task  $\tau_j$ , the data aggregation mechanism proposed in Algorithm 4 minimizes the upper bound of the probability  $\Pr[\widehat{L}_j \neq l_j]$  established in Lemma 3, and ensures that*

$$\Pr[\widehat{L}_j \neq l_j] \leq \exp\left(-\frac{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2}{2}\right). \quad (3.9)$$

*Proof.* We define the vector  $\boldsymbol{\lambda}_j = [\lambda_{i,j}]$  for every executed task  $\tau_j$ , which contains every  $\lambda_{i,j}$  such that  $w_i \in \mathcal{S}_{\mathcal{W}}$ , and  $\tau_j \in \Gamma_i$ . Therefore, minimizing the upper bound of  $\Pr[\widehat{L}_j \neq l_j]$  is equivalent to finding the vector  $\boldsymbol{\lambda}_j$  that maximizes the function  $f(\boldsymbol{\lambda}_j)$  defined as

$$f(\boldsymbol{\lambda}_j) = \frac{(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j} (2\theta_{i,j} - 1))^2}{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}.$$

Based on the Cauchy-Schwarz inequality, we have

$$f(\boldsymbol{\lambda}_j) \leq \frac{(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2)(\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2)}{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} \lambda_{i,j}^2} = \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2,$$

and equality is achieved if and only if  $\lambda_{i,j} \propto 2\theta_{i,j} - 1$ . Thus,

$$\Pr[\widehat{L}_j \neq l_j] \leq \exp\left(-\frac{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2}{2}\right). \quad (3.10)$$

From Inequality (3.10), in order to minimize the upper bound of  $\Pr[\widehat{L}_j \neq l_j]$ , we can let  $\lambda_{i,j} = 2\theta_{i,j} - 1$ , and thus, we have that

$$\Pr[\widehat{L}_j \neq l_j] \leq \exp\left(-\frac{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2}{2}\right),$$

which exactly proves Theorem 10. □

By Theorem 10, we have that the data aggregation mechanism proposed in Algorithm 4 upper bounds the error probability  $\Pr[\widehat{L}_j \neq l_j]$  by  $\exp\left(-\frac{1}{2} \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2\right)$ , which in fact is the minimum value of the upper bound of this probability established in Lemma 3. Next, we derive Corollary 1, which is directly utilized in our design of the incentive mechanism in Section 3.3.2.

**Corollary 1.** *For every executed task  $\tau_j$ , the data aggregation mechanism proposed in Algorithm 4 satisfies that if*

$$\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 \geq 2 \ln\left(\frac{1}{\beta_j}\right), \quad (3.11)$$

then  $\Pr[\widehat{L}_j \neq l_j] \leq \beta_j$ , i.e.,  $\beta_j$ -accuracy is satisfied for this task  $\tau_j$ , where  $\beta_j \in (0, 1)$  is a platform chosen parameter. Moreover, we define  $\boldsymbol{\beta}$  as the vector  $(\beta_1, \dots, \beta_M)$ .

*Proof.* By setting the upper bound of  $\Pr[\widehat{L}_j \neq l_j]$  given in Theorem 10 to be no greater than  $\beta_j \in (0, 1)$ , we have

$$\exp\left(-\frac{\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2}{2}\right) \leq \beta_j,$$

which is equivalent to

$$\sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 \geq 2 \ln\left(\frac{1}{\beta_j}\right). \quad (3.12)$$

Hence, together with Theorem 10, we have that Inequality (3.12) indicates that  $\Pr[\widehat{L}_j \neq l_j] \leq \beta_j$ .  $\square$

Corollary 1 gives us a sufficient condition, represented by Inequality (3.11), that the set of winning workers  $\mathcal{S}_{\mathcal{W}}$  selected by the incentive mechanism (proposed in Section 3.3.2) should satisfy so as to achieve  $\beta_j$ -accuracy for each executed task  $\tau_j$ .

### 3.3.2 Incentive Mechanism

Now, we introduce the design details of CENTURION’s incentive mechanism, including its mathematical formulation, the hardness proof of the formulated integer program, the proposed mechanism, as well as the corresponding analysis.

#### 3.3.2.1 Mathematical Formulation

As mentioned in Section 3.2.3, CENTURION’s incentive mechanism is based on the MELON double auction defined in Definition 9. In this chapter, we aim to design a MELON double auction that *maximizes the social welfare*, while guaranteeing *satisfactory data aggregation accuracy*. The formal mathematical formulation of its winner selection problem is provided in the following *MELON double auction social welfare maximization (MELON-SWM) problem*.

##### MELON-SWM Problem:

$$\max \sum_{j:\tau_j \in \mathcal{T}} a_j y_j - \sum_{i:w_i \in \mathcal{W}} b_i x_i \quad (3.13)$$

$$\text{s.t.} \quad \sum_{i:w_i \in \mathcal{W}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 x_i \geq 2 \ln \left( \frac{1}{\beta_j} \right) y_j, \quad \forall \tau_j \in \mathcal{T} \quad (3.14)$$

$$x_i, y_j \in \{0, 1\}, \quad \forall w_i \in \mathcal{W}, \tau_j \in \mathcal{T} \quad (3.15)$$

**Constants.** The MELON-SWM problem takes as inputs the task set  $\mathcal{T}$ , the worker set  $\mathcal{W}$ , the requesters’ and workers’ bid profile  $\mathbf{a}$  and  $\mathbf{b}$ , the profile of workers’ interested task sets  $\mathbf{\Gamma}$ , the workers’ reliability level matrix  $\mathbf{\theta}$ , and the  $\mathbf{\beta}$  vector.

**Variables.** On one hand, the MELON-SWM problem has a vector of  $M$  binary variables, denoted as  $\mathbf{y} = (y_1, \dots, y_M)$ . Any  $y_j = 1$  indicates that task  $\tau_j$  will be executed, and thus,

requester  $r_j$  is a winning requester (i.e.,  $r_j \in \mathcal{S}_{\mathcal{R}}$ ), whereas  $y_j = 0$  means  $r_j \notin \mathcal{S}_{\mathcal{R}}$ . On the other hand, the problem has another vector of  $N$  binary variables, denoted as  $\mathbf{x} = (x_1, \dots, x_N)$ , where  $x_i = 1$  indicates that worker  $w_i$  is a winning worker (i.e.,  $w_i \in \mathcal{S}_{\mathcal{W}}$ ), and  $x_i = 0$  means  $w_i \notin \mathcal{S}_{\mathcal{W}}$ .

**Objective function.** The objective function satisfies that  $\sum_{j:\tau_j \in \mathcal{T}} a_j y_j - \sum_{i:w_i \in \mathcal{W}} b_i x_i = \sum_{j:r_j \in \mathcal{S}_{\mathcal{R}}} a_j - \sum_{i:w_i \in \mathcal{S}_{\mathcal{W}}} b_i$ , which is exactly the social welfare defined in Definition 13 based on the requesters' and workers' bids.

**Constraints.** For each task  $\tau_j$ , Constraint (3.14) naturally holds, if  $y_j = 0$ . When  $y_j = 1$ , it is equivalent to Inequality (3.11) given in Corollary 1, which specifies the condition that the set of selected winning workers  $\mathcal{S}_{\mathcal{W}}$  should satisfy in order to guarantee  $\beta_j$ -accuracy for task  $\tau_j$ . To simplify the presentation, we introduce the following notations, namely  $q_{i,j} = (2\theta_{i,j} - 1)^2$ ,  $\mathbf{q} = [q_{i,j}] \in [0, 1]^{N \times M}$ ,  $Q_j = 2 \ln(\frac{1}{\beta_j})$ , and  $\mathbf{Q} = [Q_j] \in [0, +\infty)^{M \times 1}$ . Thus, Constraint (3.14) can be simplified as

$$\sum_{i:w_i \in \mathcal{W}, \tau_j \in \Gamma_i} q_{i,j} x_i \geq Q_j y_j, \quad \forall \tau_j \in \mathcal{T}. \quad (3.16)$$

Besides, we say a task  $\tau_j$  is *covered* by a solution, if  $y_j = 1$ .

### 3.3.2.2 Hardness Proof

We prove the NP-hardness of the MELON-SWM problem by performing a polynomial-time reduction from the 3SAT(5) problem which is formally defined in Definition 17.

**Definition 17** (3SAT(5) Problem). *In a 3SAT(5) problem, we are given a set  $\mathcal{O} = \{z_1, \dots, z_n\}$  of  $n$  Boolean variables, and a collection  $C_1, \dots, C_m$  of  $m$  clauses. Each clause is an OR of exactly three literals, and every literal is either a variable of  $\mathcal{O}$  or its negation. Moreover, every variable participates in exactly 5 clauses. Therefore,  $m = \frac{5n}{3}$ . Given some constant  $0 < \epsilon < 1$ , a 3SAT(5) instance  $\varphi$  is a Yes-Instance if there is an assignment to the variables of  $\mathcal{O}$  satisfying all clauses, whereas it is a No-Instance (with respect to  $\epsilon$ ), if every assignment to the variables satisfies at most  $(1 - \epsilon)m$  clauses. An algorithm  $\mathcal{A}$  distinguishes between the Yes- and No-instances of the problem, if, given a Yes-Instance, it returns a “YES” answer, and given a No-Instance it returns a “NO” answer.*

Regarding the hardness of the 3SAT(5) problem, we introduce without proof the following

well-known Lemma 4, which is a consequence of the PCP theorem [11].

**Lemma 4.** *There is some constant  $0 < \epsilon < 1$ , such that distinguishing between the Yes- and No-instances of the 3SAT(5) problem, defined with respect to  $\epsilon$ , is NP-complete.*

Next, we introduce Theorem 11 and 12 that will be utilized to prove the NP-hardness of the MELON-SWM problem.

**Theorem 11.** *Any 3SAT(5) instance is polynomial-time reducible to an instance of the MELON-SWM problem.*

*Proof.* The reduction goes as follows. Assume there is a 3SAT(5) instance  $\varphi$  on  $n$  variables and  $m$  clauses. We define 3 parameters:  $X = \frac{\epsilon m}{100}$  ( $0 < \epsilon < 1$ ),  $Y = mnX$ , and  $Z = mnY$ . The exact values of  $Y$  and  $Z$  are not important. We just need to ensure  $Z \gg Y \gg X$ . We construct an instance of the MELON-SWM problem corresponding to  $\varphi$ , by defining the task set  $\mathcal{T}$ , and the profile of workers' interested task sets  $\mathbf{\Gamma}$ .

Out of the 8 possible assignments to the variables of some clause  $C_k \in \varphi$ , exactly one does not satisfy  $C_k$ . Let  $A_k$  be the set of the remaining 7 assignments. We define a set of tasks  $\Gamma(C_k, \alpha)$  for each clause  $C_k$  and assignment  $\alpha \in A_k$ , let  $\mathbf{\Gamma} = [\Gamma(C_k, \alpha)]$  for each clause  $C_k \in \varphi$  and assignment  $\alpha \in A_k$ , set the  $q_{i,j}$  value of each worker  $w_i$  and task  $\tau_j \in \Gamma_i$  as  $q_{i,j} = 1$ , and set her bid as  $b_i = 3 + Y + Z$ . We also create a dummy worker  $w_0$ , with  $q_0 = 1$ ,  $b_0 = 0$ , and  $\Gamma_0$  being her interested task set. We start with all set  $\Gamma(C_k, \alpha)$ 's being empty, gradually define the tasks, and specify which sets they belong to. The task set  $\mathcal{T}$  consists of 4 subsets.

- The 1st subset  $E_1$  contains a task  $\tau(z_l, \gamma)$  for each variable  $z_l \in \mathcal{O}$  and assignment  $\gamma \in \{T, F\}$  to this variable.  $\tau(z_l, \gamma)$  belongs to each set  $\Gamma(C_k, \alpha)$ , such that  $z_l$  participates in  $C_k$ , and the assignment  $\alpha$  to the variables of  $C_k$  gives assignment  $\gamma$  to  $z_l$ . The  $Q_j$  value of the task  $\tau_j$  corresponding to  $\tau(z_l, \gamma)$  is set as 5 – the number of the clauses containing  $z_l$ , and the value  $v_j$  of this task is set as 5.
- The 2nd subset  $E_2$  contains  $m$  tasks  $\tau_1, \dots, \tau_m$ . Each  $\tau_k \in E_2$  belongs to all sets corresponding to  $C_k$  and  $C_{k+1}$ , i.e.,  $\tau_k$  belongs to all sets  $\{\Gamma(C_k, \alpha) | \alpha \in A_k\} \cup \{\Gamma(C_{k+1}, \alpha') | \alpha' \in A_{k+1}\}$  with the subscripts being modulo  $m$ . The  $Q_k$  value of each such  $\tau_k$  is set as 2, and its value  $v_k$  is set as  $Y$ .

- The 3rd subset  $E_3$  contains a task  $\tau(C_k)$  for each clause  $C_k$ , and  $\tau(C_k)$  belongs to set  $\Gamma(C_k, \alpha)$  for each  $\alpha \in A_k$ . The  $Q_j$  value of the task  $\tau_j$  corresponding to  $\tau(C_k)$  is set as 1, and its value  $v_j$  is set as  $Z$ .
- The 4th subset  $E_4$  contains a single task  $\tau^*$ , whose  $Q_j$  value is set as 1 and value  $v_j$  is set as  $X$ . The task  $\tau^*$  only belongs to set  $\Gamma_0$ .

This finishes the description of the reduction. Clearly, given a 3SAT(5) instance  $\varphi$ , we can construct an instance of the MELON-SWM problem in time polynomial in  $n$ .  $\square$

We now analyze the optimal social welfare for an instance of the MELON-SWM problem that corresponds to a 3SAT(5) instance  $\varphi$ , when  $\varphi$  is a Yes- or No-Instance. Note that the following analysis uses the same reduction as in Theorem 11.

**Theorem 12.** *If the 3SAT(5) instance  $\varphi$  is a Yes-Instance, then there is a solution to the resulting instance of the MELON-SWM problem whose social welfare is  $X$ . If  $\varphi$  is a No-Instance, then any solution has social welfare at most 0.*

*Proof.* Let  $\varphi$  be a Yes-Instance, and  $A$  be an assignment to the variables satisfying all clauses. We construct a solution  $\mathcal{S}'$  to the MELON-SWM problem. Firstly, we add  $\Gamma_0$  to  $\mathcal{S}'$ . Next, for each clause  $C_k$ , we add to  $\mathcal{S}'$  the unique set  $\Gamma(C_k, \alpha)$ , where  $\alpha$  is the assignment consistent with  $A$ . Then  $|\mathcal{S}'| = m$ , and the total cost of all sets is  $(Y + Z + 3)m$ . We now analyze the number of tasks covered by  $\mathcal{S}'$ , and their values. Clearly,  $\tau^*$  is covered by  $\mathcal{S}'$ , and it contributes  $X$  to the solution value.

- For each clause  $C_k \in \varphi$ , the unique task  $\tau(C_k) \in E_3$  is covered. Thus, all tasks in  $E_3$  are covered, and overall they contribute value  $mZ$  to the solution.
- Consider some  $\tau_k \in E_2$ .  $\mathcal{S}'$  contains one set corresponding to  $C_k$  and  $C_{k+1}$ , respectively. Since  $\tau_k$  belongs to both these sets, and its  $Q_k$  is 2, it is covered. Thus, all tasks in  $E_2$  are covered, and they contribute value  $mY$  to the solution.
- Consider some variable  $z_k \in \mathcal{O}$ , and let  $\gamma_k \in \{T, F\}$  be the assignment to  $z_k$  under  $A$ . If  $C_k$  is any clause containing  $z_k$ , and  $\Gamma(C_k, \alpha)$  is the set that belongs to  $\mathcal{S}'$ , then  $\alpha$  gives the

assignment  $\gamma_k$  to  $z_k$ . Thus, for all five clauses containing  $z_k$ , the corresponding sets chosen to  $\mathcal{S}'$  contain  $\tau(z_k, \gamma_k)$ , and this task is covered. So the total number of tasks of  $E_1$  covered by  $\mathcal{S}'$  is  $n$ . Each such task contributes value 5, and the total value contributed by the tasks in  $E_1$  is  $5n = 3m$ .

Therefore, the overall social welfare of this solution is  $X + mZ + mY + 3m - (Z + Y + 3)m = X$ .

Assume now that  $\varphi$  is a No-Instance, and let  $\mathcal{S}'$  be any solution with positive social welfare. We can assume that  $\Gamma_0 \in \mathcal{S}'$ , and task  $\tau^*$  is covered by  $\mathcal{S}'$ . We then introduce the following observations. Because of space limit, we place the proofs of these observations in the technical report [52].

**Observation 1.** *For every clause  $C_k$  of  $\varphi$ , at most one of the sets  $\{\Gamma(C_k, \alpha) \mid \alpha \in A_k\}$  belongs to  $\mathcal{S}'$ , and  $|\mathcal{S}'| = m$ .*

**Observation 2.** *For every variable  $z_k \in \mathcal{O}$ , at most one of the two tasks  $\tau(z_k, T)$  and  $\tau(z_k, F)$  is covered by  $\mathcal{S}'$ .*

We say that a variable  $z_k \in \mathcal{O}$  is bad if neither  $\tau(z_k, T)$  nor  $\tau(z_k, F)$  is covered by  $\mathcal{S}'$ ; otherwise it is good. We next show that only a small number of the variables are bad.

**Observation 3.** *There are at most  $\frac{\epsilon n}{100}$  bad variables.*

In the next step, we construct the following assignment to the variables of  $\mathcal{O}$ . If variable  $z_k \in \mathcal{O}$  is good, then there is a unique value  $\gamma_k \in \{T, F\}$ , such that task  $\tau(z_k, \gamma_k)$  is covered by  $\mathcal{S}'$ . We then assign  $z_k$  the value  $\gamma_k$ . If  $z_k$  is bad, we assign it any value arbitrarily. We now make the claim that the above assignment satisfies more than  $(1 - \epsilon)m$  clauses. We say that a clause is bad if it contains a bad variable, and it is good otherwise. Since there are at most  $\frac{\epsilon n}{100}$  bad variables, and each variable participates in 5 clauses, the number of bad clauses is at most  $\frac{\epsilon n}{20} \leq \frac{3\epsilon m}{100}$ . So there are more than  $(1 - \epsilon)m$  good clauses. Let  $C_l$  be a good clause, and  $\Gamma(C_l, \alpha)$  be the set corresponding to  $C_l$  that belongs to  $\mathcal{S}'$ . Then  $\alpha$  is an assignment to the variables of  $C_l$  that satisfies  $C_l$ , and each variable participating in  $C_l$  was assigned a value that is consistent with  $\alpha$ . As a result, clause  $C_l$  is satisfied.

To conclude, we have assumed that  $\varphi$  is a No-Instance, and showed that, if the MELON-SWM problem has a solution with non-negative social welfare, there is an assignment to the variables of

$\varphi$  satisfying more than  $(1 - \epsilon)m$  of its clauses, which is impossible for a No-Instance. Therefore, if  $\varphi$  is a No-Instance, every solution has social welfare at most 0.  $\square$

Next, we describe Theorem 13 that states the *NP-hardness* and *inapproximability* of the MELON-SWM problem.

**Theorem 13.** *The MELON-SWM problem is NP-hard, and for any factor  $\phi$ , there is no efficient  $\phi$ -approximation algorithm to the MELON-SWM problem.*

*Proof.* Based on Theorem 11, there exists a reduction from any 3SAT(5) problem instance  $\varphi$  to an instance  $\mathcal{I}(\varphi)$  of the MELON-SWM problem. From Theorem 12, we have that the optimal solution to  $\mathcal{I}(\varphi)$  also gives a solution to  $\varphi$ . That is, if the optimal social welfare of  $\mathcal{I}(\varphi)$  is positive, then  $\varphi$  is a Yes-Instance; otherwise,  $\varphi$  is a No-Instance. Together with Lemma 4 stating the NP-completeness of the 3SAT(5) problem, we conclude that the MELON-SWM problem is NP-hard.

In fact, Theorem 11 and 12 give an inapproximability result about the MELON-SWM, as well. Suppose there is an efficient factor- $\phi$  approximation algorithm  $\mathcal{A}$  for the MELON-SWM problem. We can use it to distinguish Yes- and No-instances of the 3SAT(5) problem on  $n \gg \phi$  variables. If  $\varphi$  is a Yes-Instance, then the algorithm has to return a solution with positive social welfare for  $\mathcal{I}(\varphi)$ , and if  $\varphi$  is a No-Instance, then any solution has social welfare at most 0. So algorithm  $\mathcal{A}$  distinguishes the Yes- and the No-instances of 3SAT(5), contradicting Lemma 4.  $\square$

### 3.3.2.3 Proposed Mechanism

Theorem 13 not only shows the NP-hardness of the MELON-SWM problem, but also indicates that there is no efficient algorithm with a guaranteed approximation ratio for it. Therefore, we relax the requirement of provable approximation ratio, and propose the following MELON double auction that aims to ensure *non-negative social welfare*, instead. Its winner selection algorithm is given in the following Algorithm 5.

Algorithm 5 takes as inputs the task set  $\mathcal{T}$ , the requester set  $\mathcal{R}$ , the worker set  $\mathcal{W}$ , the profile of workers' interested task sets  $\mathbf{\Gamma}$ , the requesters' and workers' bid profile  $\mathbf{a}$  and  $\mathbf{b}$ , the  $\mathbf{q}$  matrix, as well as the  $\mathbf{Q}$  vector. Firstly, it initializes the winning requester and worker set as  $\emptyset$  (line 1). Then, it calculates a *feasible cover*, denoted by  $\mathcal{C}$ , containing the set of workers that make Constraint

---

**Algorithm 5:** MELON Double Auction Winner Selection
 

---

**Input:**  $\mathcal{T}, \mathcal{R}, \mathcal{W}, \mathbf{\Gamma}, \mathbf{a}, \mathbf{b}, \mathbf{q}, \mathbf{Q}$ ;  
**Output:**  $\mathcal{S}_{\mathcal{R}}, \mathcal{S}_{\mathcal{W}}, \mathcal{C}$ ;  
 // Initialization  
 1  $\mathcal{S}_{\mathcal{R}} \leftarrow \emptyset, \mathcal{S}_{\mathcal{W}} \leftarrow \emptyset$ ;  
 // Find a feasible cover  
 2  $\mathcal{C} \leftarrow \text{FC}(\mathcal{T}, \mathbf{\Gamma}, \mathbf{q}, \mathbf{Q})$ ;  
 3 **foreach**  $j$  s.t.  $\tau_j \in \mathcal{T}$  **do**  
 4    $\mathcal{C}_j \leftarrow \{w_i | w_i \in \mathcal{C}, \tau_j \in \Gamma_i\}$ ;  
 // Main loop  
 5 **while**  $\max_{j:r_j \in \mathcal{R}} (a_j - \sum_{i:w_i \in \mathcal{C}_j} b_i) \geq 0$  **do**  
 6    $j^* \leftarrow \arg \max_{j:r_j \in \mathcal{R}} (a_j - \sum_{i:w_i \in \mathcal{C}_j} b_i)$ ;  
 7    $\mathcal{S}_{\mathcal{R}} \leftarrow \mathcal{S}_{\mathcal{R}} \cup \{r_{j^*}\}$ ;  
 8    $\mathcal{R} \leftarrow \mathcal{R} \setminus \{r_{j^*}\}$ ;  
 9    $\mathcal{S}_{\mathcal{W}} \leftarrow \mathcal{S}_{\mathcal{W}} \cup \mathcal{C}_{j^*}$ ;  
 10   **foreach**  $j$  s.t.  $r_i \in \mathcal{R}$  **do**  
 11      $\mathcal{C}_j \leftarrow \mathcal{C}_j \setminus \mathcal{C}_{j^*}$ ;  
 12 **return**  $\mathcal{S}_{\mathcal{R}}, \mathcal{S}_{\mathcal{W}}$ ;  


---

(3.16) feasible for each task  $\tau_j$  given that each  $y_j = 1$ , by calling another algorithm FC which takes the task set  $\mathcal{T}$ , the profile of workers' interested task sets  $\mathbf{\Gamma}$ , the  $\mathbf{q}$  matrix, and the  $\mathbf{Q}$  vector as inputs (line 2). Algorithm FC can be easily implemented in time polynomial in  $M$  and  $N$ . For example, FC could greedily select each worker  $w_i$  into the feasible cover in a decreasing order of the value  $\sum_{j:\tau_j \in \Gamma_i} q_{i,j}$  until all constraints are satisfied. The computational complexity of such FC is  $O(N)$ . We assume that FC adopts such a greedy approach in the rest of this chapter. Note that the specific choice of FC is not important, as long as it returns a feasible cover in polynomial time. Next, for each task  $\tau_j$ , Algorithm 5 chooses from the feasible cover the set of workers  $\mathcal{C}_j$  whose interested task sets contain this task (line 3-4).

Based on  $\mathcal{C}$ , the main loop (line 5-11) of the algorithm selects the set of winning requesters and workers that give non-negative social welfare. It executes until  $\max_{j:r_j \in \mathcal{R}} (a_j - \sum_{i:w_i \in \mathcal{C}_j} b_i)$ , the *maximum marginal social welfare* of including a new requester  $r_j$  and the set of workers  $\mathcal{C}_j$  into, respectively, the winning requester and worker set, becomes negative (line 5). In each iteration of the main loop, the Algorithm finds first the index  $j^*$  of the requester  $r_{j^*}$  that provides the *maximum marginal social welfare* (line 6). Next, it includes  $r_{j^*}$  into the winning requester set  $\mathcal{S}_{\mathcal{R}}$  (line 7), removes  $r_{j^*}$  from the requester set  $\mathcal{R}$  (line 8), and includes all workers in  $\mathcal{C}_{j^*}$  into the winning

worker set  $\mathcal{S}_{\mathcal{W}}$  (line 9). The last step of the main loop is to remove all workers in  $\mathcal{C}_{j^*}$  from  $\mathcal{C}_j$  for each task  $\tau_j$  (line 10). Finally, Algorithm 5 returns the winning requester and worker set  $\mathcal{S}_{\mathcal{R}}$  and  $\mathcal{S}_{\mathcal{W}}$  (line 12).

Next, we present the pricing algorithm of the MELON double auction in Algorithm 6.

---

**Algorithm 6:** MELON Double Auction Pricing

---

**Input:**  $\mathcal{T}, \mathcal{R}, \mathcal{W}, \Gamma, \mathbf{a}, \mathbf{b}, \mathbf{q}, \mathbf{Q}, \mathcal{S}_{\mathcal{R}}, \mathcal{S}_{\mathcal{W}}$ ;  
**Output:**  $\mathbf{p}^r, \mathbf{p}^w$ ;  
// Initialization  
**1**  $\mathbf{p}^r \leftarrow \mathbf{0}, \mathbf{p}^w \leftarrow \mathbf{0}$ ;  
// Pricing for winning requesters  
**2** **foreach**  $j$  *s.t.*  $r_j \in \mathcal{S}_{\mathcal{R}}$  **do**  
**3**     run Algorithm 5 on  $\mathcal{R} \setminus \{r_j\}$  and  $\mathcal{W}$ ;  
**4**      $\mathcal{S}'_{\mathcal{R}} \leftarrow$  winning requester set when line 3 stops;  
**5**     **foreach**  $k$  *s.t.*  $r_k \in \mathcal{S}'_{\mathcal{R}}$  **do**  
**6**          $\lfloor p_j^r \leftarrow \min \{p_j^r, \sum_{w_i \in \mathcal{C}'_j} b_i + a_k - \sum_{w_i \in \mathcal{C}'_k} b_i\}$ ;  
**7**     **if**  $\mathcal{C}'_j = \emptyset$  **then**  
**8**          $\lfloor p_j^r \leftarrow \min\{p_j^r, 0\}$ ;  
// Pricing for winning workers  
**9** **foreach**  $i$  *s.t.*  $w_i \in \mathcal{S}_{\mathcal{W}}$  **do**  
**10**     run Algorithm 5 on  $\mathcal{R}$  and  $\mathcal{W} \setminus \{w_i\}$ ;  
**11**      $\mathcal{S}'_{\mathcal{R}} \leftarrow$  winning requester set when line 10 stops;  
**12**     **foreach**  $k$  *s.t.*  $w_i \in \mathcal{C}'_k$  and  $r_k \in \mathcal{S}'_{\mathcal{R}}$  **do**  
**13**         sort requesters according to the decreasing order of  $a_j - \sum_{i:w_i \in \mathcal{C}'_j} b_i$ ;  
**14**          $f \leftarrow$  index of the first requester with  $w_i \notin \mathcal{C}'_f$ ;  
**15**         **if**  $r_f \in \mathcal{S}'_{\mathcal{R}}$  **then**  
**16**              $\lfloor p_i^w \leftarrow \max \{p_i^w, a_k - \sum_{w_h \in \mathcal{C}'_k} b_h - (a_f - \sum_{w_h \in \mathcal{C}'_f} b_h)\}$ ;  
**17**         **else**  
**18**              $\lfloor p_i^w \leftarrow \max \{p_i^w, a_k - \sum_{w_h \in \mathcal{C}'_k} b_h\}$ ;  
**19** **return**  $\mathbf{p}^r, \mathbf{p}^w$ ;

---

Apart from the same inputs to Algorithm 5, Algorithm 6 also takes as inputs the winning requester and worker set  $\mathcal{S}_{\mathcal{R}}$  and  $\mathcal{S}_{\mathcal{W}}$ , outputted by Algorithm 5. Firstly, Algorithm 6 initializes the requesters' and workers' payment profile as zero vectors (line 1). Then, it calculates the payment  $p_j^r$  charged from each winning requester (line 2-8). For each  $r_j \in \mathcal{S}_{\mathcal{R}}$ , Algorithm 5 is executed on the worker set  $\mathcal{W}$  and requester set  $\mathcal{R}$  except requester  $r_j$  (line 3). Next, it sets  $\mathcal{S}'_{\mathcal{R}}$  as the winning requester set when line 3 stops (line 4). For each  $r_k \in \mathcal{S}'_{\mathcal{R}}$ , Algorithm 6 finds the

minimum bid  $a_{j,k}$  for requester  $r_j$  to replace  $r_k$  as the winner. To achieve this,  $a_{j,k}$  should satisfy  $a_{j,k} - \sum_{w_i \in \mathcal{C}'_j} b_i = a_k - \sum_{w_i \in \mathcal{C}'_k} b_i$ , which is equivalent to  $a_{j,k} = \sum_{w_i \in \mathcal{C}'_j} b_i + a_k - \sum_{w_i \in \mathcal{C}'_k} b_i$ . Note that  $\mathcal{C}'_1, \dots, \mathcal{C}'_M$  denote the sets  $\mathcal{C}_1, \dots, \mathcal{C}_M$  when the specific requester  $r_k$  is selected into  $\mathcal{S}'_{\mathcal{R}}$ . If  $\mathcal{C}'_j$  is not empty, the minimum value among these  $a_{j,k}$ 's is chosen as the payment  $p_j^r$  (line 5-6); otherwise, it is further compared with 0 (line 7-8), since requester  $r_j$  could win, in this case, as long as her bid is non-negative.

Next, Algorithm 6 derives the payment  $p_i^w$  to each winning worker  $w_i$  (line 9-18). Similar to line 3, Algorithm 5 is executed on the requester set  $\mathcal{R}$  and worker set  $\mathcal{W}$  except worker  $w_i$  (line 10), and  $\mathcal{S}'_{\mathcal{R}}$  is set as the winning requester set when line 10 stops (line 11). In the rest of the algorithm, we also use  $\mathcal{C}'_1, \dots, \mathcal{C}'_M$  to denote the sets  $\mathcal{C}_1, \dots, \mathcal{C}_M$  when the specific requester  $r_k$  is selected into  $\mathcal{S}'_{\mathcal{R}}$ . For each set  $\mathcal{C}'_k$  such that  $w_i$  belongs to  $\mathcal{C}'_k$  and  $r_k$  belongs to  $\mathcal{S}'_{\mathcal{R}}$ , the algorithm calculates the maximum bid  $b_{i,k}$  for worker  $w_i$  to be selected as a winner at this point (line 12-18). The calculation firstly sorts requesters in the decreasing order of their marginal social welfare, i.e.,  $a_j - \sum_{i:w_i \in \mathcal{C}'_j} b_i$  (line 13), and finds the index  $f$  of the first the requester in this order such that  $w_i$  does not belong to  $\mathcal{C}'_f$  (line 14). If  $r_f$  is a winning requester in  $\mathcal{S}'_{\mathcal{R}}$ , then  $b_{i,k}$  should satisfy  $a_k - (\sum_{w_h \in \mathcal{C}'_k} b_h + b_{i,k}) = a_f - \sum_{w_h \in \mathcal{C}'_f} b_h$ , which is equivalent to  $b_{i,k} = a_k - \sum_{w_h \in \mathcal{C}'_k} b_h - (a_f - \sum_{w_h \in \mathcal{C}'_f} b_h)$ ; otherwise,  $b_{i,k}$  should satisfy  $a_k - (\sum_{w_h \in \mathcal{C}'_k} b_h + b_{i,k}) = 0$ , which is equivalent to  $b_{i,k} = a_k - \sum_{w_h \in \mathcal{C}'_k} b_h$ . Then, the maximum value among these  $b_{i,k}$ 's are chosen as the payment  $p_i^w$  (line 15-18). Finally, Algorithm 6 returns the requesters' and workers' payment profile  $\mathbf{p}^r$  and  $\mathbf{p}^w$  (line 19).

### 3.3.2.4 Analysis of the Proposed Mechanism

In this section, we prove several desirable properties of our MELON double auction, described in Algorithm 5 and 6. Firstly, we show its truthfulness in Theorem 14.

**Theorem 14.** *The proposed MELON double auction is truthful.*

*Proof.* We prove the truthfulness of the MELON double auction by showing that it satisfies the properties of *monotonicity* and *critical payment*.

- **Monotonicity.** The algorithm FC called by Algorithm 5 is independent of the requesters' and workers' bids, and winners are selected based on a decreasing order of the value  $a_j - \sum_{i:w_i \in \mathcal{C}_j} b_i$ .

Thus, if a requester  $r_j$  wins by bidding  $a_j$ , she will also win the auction by bidding any  $a'_j > a_j$ . Similarly, if a worker  $w_i$  wins by bidding  $b_i$ , she will win the auction, as well, if her bid takes any value  $b'_i < b_i$ .

- **Critical payment.** Algorithm 6 in fact pays every winning requester and worker the infimum and supremum of her bid, respectively, that can make her a winner.

As proved in [15], these two properties make an auction truthful, i.e., each requester  $r_j$  maximizes her utility by bidding  $v_j$ , and each worker  $w_i$  maximizes her utility by bidding  $c_i$ . Therefore, the MELON double auction is truthful.  $\square$

Next, we show that the proposed MELON double auction satisfies individual rationality in Theorem 15.

**Theorem 15.** *The proposed MELON double auction is individual rational.*

*Proof.* By Definition 10 and 11, losers of the MELON double auction receive zero utilities. From Theorem 14, every winning requester  $r_j$  bids  $v_j$ , and every winning worker  $w_i$  bids  $c_i$  to the platform. Moreover, they are paid, respectively, the infimum and supremum of the bid for them to win the auction. Therefore, it is guaranteed that all requesters and workers receive non-negative utilities, and thus the proposed MELON double auction is individual rational.  $\square$

In Theorem 16, we prove that the proposed MELON double auction has a polynomial-time computational complexity.

**Theorem 16.** *The computational complexity of the proposed MELON double auction is  $O(M^3N + M^2N^2)$ .*

*Proof.* As mentioned in Section 3.3.1.1, the algorithm FC (line 2) in Algorithm 5 takes a greedy approach, and has a computational complexity of  $O(N)$ . Line 3-4 of Algorithm 5 that find the sets  $\mathcal{C}_1, \dots, \mathcal{C}_M$  terminate at most after  $MN$  steps. Next, the main loop (line 5-11) terminates after  $M$  iterations in worst case. Within each iteration, finding the index of the requester that provides the maximum marginal social welfare (line 6) takes  $O(M)$  time, and updating the sets  $\mathcal{C}_1, \dots, \mathcal{C}_M$  takes  $O(MN)$  time. Therefore, the computational complexity of the main loop is  $O(MN)$ , and

thus, that of Algorithm 5 is  $O(M^2N)$  overall. After Algorithm 5, our MELON double auction executes its pricing algorithm described by Algorithm 6, where the loop for requester pricing (line 1-8) terminates in worst case after  $M$  iterations. Clearly, the computational complexity of each iteration of the loop is dominated by the execution of Algorithm 5 in line 3. Therefore, the requester pricing (line 1-8) in Algorithm 6 takes  $O(M^3N)$  time. Following a similar method of analysis, we can conclude that the worker pricing in Algorithm 6 takes  $O(M^2N^2)$  time. Hence, the computation complexity of Algorithm 6, as well as that of the overall MELON double auction is  $O(M^3N + M^2N^2)$ .  $\square$

Finally, we show in Theorem 17 that our MELON double auction guarantees non-negative social welfare, as required.

**Theorem 17.** *The MELON double auction guarantees non-negative social welfare.*

*Proof.* Clearly, in the winner selection algorithm described by Algorithm 5, a requester  $r_j$  and the workers in  $\mathcal{C}_j$  could be selected as winners, only if the corresponding marginal social welfare  $a_j - \sum_{i:w_i \in \mathcal{C}_j} b_i$  is non-negative (line 5). Thus, as the overall social welfare given by Algorithm 5 is the sum of the aforementioned marginal social welfare of every iteration where new winners are selected, the MELON double auction guarantees non-negative social welfare.  $\square$

## 3.4 Performance Evaluation

In this section, we introduce the baseline methods, simulations settings, as well as simulation results of the performance evaluation about our proposed CENTURION framework.

### 3.4.1 Baseline Methods

In our evaluation of the incentive mechanism, the first baseline auction is the Marginal Social Welfare greedy (MSW-Greedy) double auction. As in Algorithm 5, it also initializes the winner sets as  $\emptyset$ , executes the algorithm FC to obtain a feasible cover  $\mathcal{C}$ , and chooses from  $\mathcal{C}$  the set  $\mathcal{C}_j$  containing each worker  $w_i$  such that  $\tau_j \in \Gamma_i$  for each task  $\tau_j$ . Different from the MELON double auction, it sorts requesters in a decreasing order of their marginal social welfare, i.e., the value  $a_j - \sum_{i:w_i \in \mathcal{C}_j} b_i$  for each requester  $r_j$ . Then, it selects the requester  $r_j$  and the set of workers in

$\mathcal{C}_j$  as winners until the marginal social welfare becomes negative. Its pricing algorithm is the same as that of the MELON double auction. Clearly, the MSW-Greedy double auction is truthful and individual rational. Another baseline auction is the one that initializes the feasible cover  $\mathcal{C}$  as the entire worker set  $\mathcal{W}$ , which we call AIR double auction. The rest of its winner selection, as well as the entire pricing algorithm is the same as those of our MELON double auction. It is easily provable that the AIR double auction is also truthful and individual rational.

Furthermore, we compare our weighted data aggregation mechanism with a mean aggregation mechanism, which outputs +1 as the aggregated result for a task if the mean of workers’ labels about this task is non-negative, and outputs  $-1$ , otherwise. Another baseline aggregation mechanism that we consider is the median aggregation that takes the median of workers’ labels about a task as its aggregated result.

### 3.4.2 Simulation Settings

Setting	$v_j$	$c_i$	$\theta_{i,j}$	$\beta_j$	$ \Gamma_i^* $	$N$	$M$
3.I	[10, 20]	[5, 15]	[0, 1]	[0.05, 0.1]	[15, 20]	[90, 150]	60
3.II	[10, 20]	[5, 15]	[0, 1]	[0.05, 0.1]	[15, 20]	60	[20, 80]

Table 3.1: Simulation setting 3.I and 3.II

The parameter settings in our simulation are given in Table 3.1. Parameters  $v_j$ ,  $c_i$ ,  $\theta_{i,j}$ ,  $\beta_j$ , and  $|\Gamma_i^*|$  are sampled uniformly at random from the intervals given in Table 3.1. The worker  $w_i$ ’s true interested task set  $\Gamma_i^*$  contains  $|\Gamma_i^*|$  tasks randomly selected from the task set  $\mathcal{T}$ . In setting 3.I, we fix the number of requesters as 60 and vary the number of workers from 90 to 150, whereas we fix the number of workers as 60 and vary the number of requesters from 20 to 80 in setting 3.II. Note that we leave the study of the values of these parameters in real-world scenarios in our future work.

### 3.4.3 Simulation Results

In Figure 3.2 and 3.3, we compare the social welfare of our MELON double auction with those of the two baseline auctions. These two figures show that our MELON double auction generates social welfare far more than the MSW-Greedy and AIR double auction in both setting 3.I and 3.II.

We evaluate CENTURION’s accuracy guarantee in setting 3.I and 3.II with a minor change of the parameter  $\beta_j$ , i.e.,  $\beta_j$  for each task  $\tau_j$  is fixed as 0.05 to simplify presentation. We compare

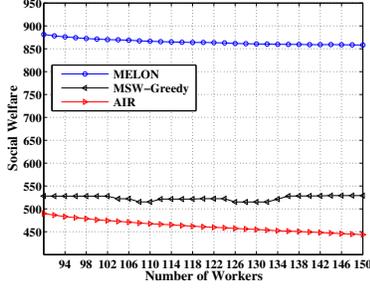


Figure 3.2: Social welfare (setting 3.I)

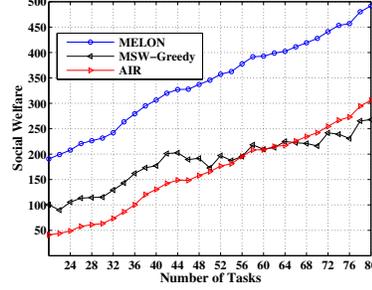


Figure 3.3: Social welfare (setting 3.II)

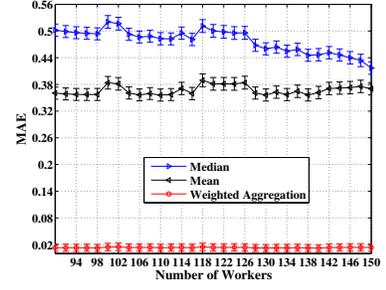


Figure 3.4: MAE (setting 3.I)

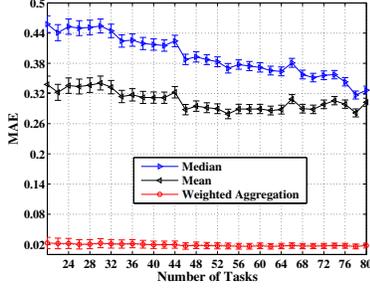


Figure 3.5: MAE (setting 3.II)

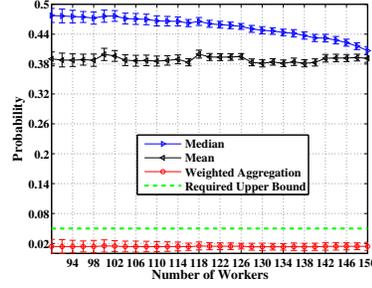


Figure 3.6: Error probability (setting 3.I)

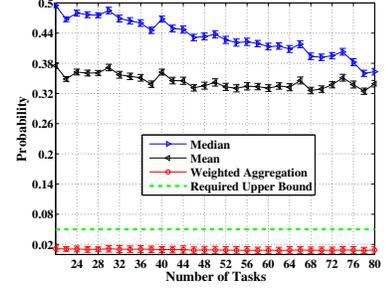


Figure 3.7: Error probability (setting 3.II)

the mean absolute error (MAE) for all tasks, which is defined as  $MAE = \frac{1}{M} \sum_{j:\tau_j \in \mathcal{T}} |\hat{l}_j - l_j|$ , of our weighted aggregation mechanism proposed in Algorithm 4 with those of the mean and median aggregation. The simulation for each combination of worker and requester number is repeated for 50000 times, and we plot the means and standard deviations of the MAEs in Figure 3.4 and 3.5. From these two figures, we observe that the MAE of our weighted aggregation mechanism is far less than those of the mean and median aggregation. Then, we show our simulation results about  $\Pr[|\hat{l}_j - l_j|]$ , referred to as task  $\tau_j$ 's error probability (EP). After 50000 repetitions of the simulation for any given combination of worker and requester number, empirical values of the EPs are calculated, and the means and standard deviations of the empirical EPs are plotted in Figure 3.6 and 3.7. These two figures show that the empirical EPs are less than the required upper bound  $\beta_j$  and far less than those of the mean and median aggregation.

### 3.5 Related Work

As previously mentioned, a series of incentive mechanisms [20, 22, 27, 28, 31–33, 35, 37, 43–45, 50, 51, 53, 57, 58, 60, 62, 66, 67, 71–75, 82, 86, 88, 89, 91, 92, 95, 97, 98, 100, 106, 107, 110–113, 115–124] have

recently been developed to serve the objective of incentivizing worker participation in MCS systems. Different from most of the these prior work which assume that there exists only one data requester, CENTURION’s incentive mechanism works in MCS systems with *multiple data requesters* that compete for human resources. In fact, there do exist several past literature work [32, 89, 98, 106, 107, 110, 111, 115, 117, 119, 123] aiming at designing incentive mechanisms in similar multi-requester scenarios. However, they do not provide any *joint design* of the data aggregation and the incentive mechanism as in this chapter, which is much more challenging than designing the two mechanisms as isolated modules. Technically, CENTURION’s incentive mechanism is based on the framework of double auction [14, 24, 76, 102, 126]. These existing double auction models typically aim to solve matching problems, and thus cannot be readily applied in our problem setting with coverage (like) constraints.

### 3.6 Conclusion

In this chapter, we propose CENTURION, a novel integrated framework for multi-requester MCS systems, consisting of a double auction-based incentive mechanism that stimulates the participation of both requesters and workers, and a data aggregation mechanism that aggregates workers data. Its incentive mechanism bears many desirable properties including truthfulness, individual rationality, computational efficiency, as well as non-negative social welfare, and its data aggregation mechanism generates highly accurate aggregated results.

## Chapter 4

# Bid Privacy-Preserving Incentive Mechanism for MCS Systems

### 4.1 Introduction

Among the various existing incentive mechanisms developed by the research community to stimulate worker participation, one important category adopts the framework of reverse auction [32, 33, 35, 37, 50, 53, 54, 60, 62, 71, 72, 75, 97, 100, 106, 107, 112, 113, 115–118, 120–122, 124]. In these auction-based mechanisms, a worker submits a bid to the platform containing one or multiple tasks she is interested in and her bidding price for executing these tasks. Based on workers' bids, the platform acting as the auctioneer determines the winners who are assigned to execute the tasks they bid and the payments paid to the selected winners. Furthermore, designing a truthful auction where every worker bids to the platform her true interested tasks and the corresponding true task execution cost is a common objective.

However, all the aforementioned incentive mechanisms [32, 33, 35, 37, 50, 53, 54, 60, 62, 71, 72, 75, 97, 100, 106, 107, 112, 113, 115–118, 120–122, 124] fail to consider the preservation of workers' *bid privacy*. Although the platform is usually considered to be trusted, there exist some *honest-but-curious* workers who strictly follow the protocol of the MCS system, but try to infer information about other workers' bids. A worker's bid usually contains her private and sensitive information. For example, a worker's bidding task set could imply her personal interests, knowledge base, etc. In geotagging MCS systems that provide accurate localization of physical objects (e.g., automated external defibrillator [8], pothole [30, 80]), bidding task sets contain the places a worker has visited or will visit, the disclosure of which breaches her location privacy. Similar to bidding task set, a worker's bidding price could also be utilized to infer her sensitive information. For example, bidding price could imply the type of mobile devices a worker uses for an MCS task, because usually workers tend to bid more if their mobile devices are more expensive.

Typically, the change in one worker’s bid has the potential to shift the overall payment profile (i.e., payments to all workers) significantly. It is possible that a curious worker could infer information about other workers’ bids from the different payments she receives in two rounds of the auction. To address this issue, we incorporate the notion of *differential privacy* [29, 77], which ensures that the change in any worker’s bid will not bring a significant change to the resulting payment profile. Therefore, different from all existing incentive mechanisms for MCS systems, we design a *differentially private incentive mechanism that protects workers’ bid privacy against honest-but-curious workers*.

Because of workers’ *selfish* and *strategic* behavior that aim to maximize their own utilities and the combinatorial nature of the tasks executed by each worker, we design an incentive mechanism based on the *single-minded reverse combinatorial auction*. In our mechanism, every worker bids on a set of tasks that she is interested to execute. The platform serves as the auctioneer and determines the winners and the payment profile that *minimize its total payment to all the winners*. In sum, this chapter has the following contributions.

- Different from all existing incentive mechanisms for MCS systems, we design a *differentially private* incentive mechanism that preserves the privacy of each worker’s bid against the other honest-but-curious workers.
- Apart from differential privacy, our mechanism also satisfies the desirable economic properties of approximate truthfulness and individual rationality.
- Algorithmically, our mechanism is computationally efficient and minimizes the platform’s total payment with a guaranteed approximation ratio.

The rest of the chapter is organized as follows. We introduce the preliminaries in Section 4.2, and present our formal mathematical formulation in Section 4.3. Our design details of the bid privacy-preserving incentive mechanism and the corresponding analyses are described in Section 4.4 and 4.5, respectively. In the next step, we present the results of our extensive simulation in Section 4.6, and summarize the related work in Section 4.7. Finally, we give the conclusion of this chapter in Section 4.8.

## 4.2 Preliminaries

In this section, we present an overview of MCS systems, the aggregation method, our auction model, and design objectives.

### 4.2.1 System Overview

The MCS system considered in this chapter consists of a cloud-based platform and a set of  $N$  participating workers denoted as  $\mathcal{N} = \{w_1, \dots, w_N\}$ .

In this chapter, we are particularly interested in MCS systems that host a set of  $K$  classification tasks, denoted as  $\mathcal{T} = \{\tau_1, \dots, \tau_K\}$ , namely ones that require workers to locally decide the classes of the objects or events she has observed, and report her local decisions (i.e., labels of the observed objects or events) to the platform. Here, we assume that all tasks in  $\mathcal{T}$  are binary classification tasks, which constitute a significant portion of the tasks posted on MCS platforms. Examples of such tasks include tagging whether or not a segment of road surface has potholes or bumps [30, 80], labeling whether or not traffic congestion happens at a specific road segment [99], etc. Each binary classification task  $\tau_j \in \mathcal{T}$  has a true class label  $l_j$ , unknown to the platform, which is either  $+1$  or  $-1$ . If worker  $w_i$  is selected to execute task  $\tau_j$ , she will provide a label  $l_{i,j}$  to the platform.

Currently, a major challenge in designing reliable MCS systems lies in the fact that the sensory data provided by individual workers are usually unreliable due to various reasons including carelessness, background noise, lack of sensor calibration, poor sensor quality, etc. To overcome this issue, the platform has to aggregate the labels provided by multiple workers, as this will likely cancel out the errors of individual workers and infer the true label. We describe the workflow<sup>1</sup> of the MCS system as follows.

- The platform firstly announces the set of binary classification tasks,  $\mathcal{T}$ , to the workers.
- Then, the workers and the platform start the auctioning stage, where the platform acts as the *auctioneer* purchasing the labels provided by the workers. Every worker  $w_i$  submits her bid  $b_i = (\Gamma_i, \rho_i)$ , which is a tuple consisting of the set of tasks  $\Gamma_i$  she wants to execute and her

---

<sup>1</sup>Note that we are specifically interested in the scenario where all workers and tasks arrive at same time. We leave the investigation of the online scenario where workers and tasks arrive sequentially in an online manner in our future work.

bidding price  $\rho_i$  for providing labels about these tasks. We use  $\mathbf{b} = (b_1, \dots, b_N)$  to denote workers' bid profile.

- Based on workers' bids, the platform determines the set of winners (denoted as  $\mathcal{S} \subseteq \mathcal{N}$ ) and the payment  $p_i$  paid to each worker  $w_i$ . We use  $\mathbf{p} = (p_1, \dots, p_N)$  to denote workers' payment profile.
- After the platform aggregates workers' labels to infer the true label of every task, it gives the payment to the corresponding winners.

Every worker  $w_i$  has a *reliability level*  $\theta_{i,j} \in [0, 1]$  for task  $\tau_j$ , which is the probability that the label  $l_{i,j}$  provided by worker  $w_i$  about task  $\tau_j$  equals to the true label  $l_j$ , i.e.,  $\Pr[l_{i,j} = l_j] = \theta_{i,j}$ . We use the matrix  $\boldsymbol{\theta} = [\theta_{i,j}] \in [0, 1]^{N \times K}$  to denote the reliability level matrix of all workers. We assume that the platform maintains a historical record of the reliability level matrix  $\boldsymbol{\theta}$  utilized as one of the inputs for winner and payment determination. There are many methods that the platform could use to estimate  $\boldsymbol{\theta}$ . In the cases where the platform has access to the true labels of some tasks *a priori*, it can assign these tasks to workers in order to estimate  $\boldsymbol{\theta}$  as in [87]. When ground truth labels are not available,  $\boldsymbol{\theta}$  can still be effectively estimated from workers' previously submitted data using algorithms such as those in [68–70, 78, 96, 101]. Alternatively, in many applications  $\boldsymbol{\theta}$  can be inferred from some explicit characteristics of the workers (e.g., a worker's reputation and experience of executing certain types of sensing tasks, the type and price of a worker's sensors) using the methods proposed in [63]. The issue of exactly which method is used by the platform to calculate  $\boldsymbol{\theta}$  is application dependent and out of the scope of this chapter.

## 4.2.2 Aggregation Method

In this chapter, we reasonably assume that the platform utilizes a weighted aggregation method to calculate the aggregated label  $\hat{l}_j$  for each task  $\tau_j$  based on the labels collected from workers. That is,

$$\hat{l}_j = \text{sign} \left( \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \alpha_{i,j} l_{i,j} \right), \quad (4.1)$$

where  $\alpha_{i,j}$  is the weight corresponding to the label  $l_{i,j}$ . In fact, many sophisticated state-of-the-art data aggregation mechanisms, such as those proposed in [68–70, 78, 96], also adopt the

weighted aggregation method to calculate the aggregation results. Given the aggregation method, the platform selects winners so that the aggregation error of each task  $\tau_j$ 's label is upper bounded by a predefined threshold  $\delta_j$ . That is, the platform aims to ensure that  $\Pr[\widehat{L}_j \neq l_j] \leq \delta_j$  holds for every task  $\tau_j \in \mathcal{T}$ , where  $\widehat{L}_j$  denotes the random variable corresponding to  $\widehat{l}_j$ . We directly apply with minor adaptation in this chapter the results derived in [54] (Theorem 1 and Corollary 1), formally summarized in Lemma 5, regarding the relationship between the selected winners' reliability levels and the upper bounds of tasks' aggregation error.

**Lemma 5.** *If the platform utilizes a weighted aggregation method that calculates each task  $\tau_j$ 's aggregated label  $\widehat{l}_j$  according to Equation (4.1) with  $\alpha_{i,j} = 2\theta_{i,j} - 1$ , and*

$$\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 \geq 2 \ln \left( \frac{1}{\delta_j} \right), \quad (4.2)$$

where  $\delta_j \in (0, 1)$ , then we have that  $\Pr[\widehat{L}_j \neq l_j] \leq \delta_j$ .

We refer to Equation (4.2) as the *error bound constraint* in the rest of this chapter. Essentially, Lemma 5 presents a necessary and sufficient condition for  $\Pr[\widehat{L}_j \neq l_j] \leq \delta_j$  to hold for each task  $\tau_j \in \mathcal{T}$  for a weighted aggregation algorithm. That is, the aggregated label  $\widehat{l}_j$  should be calculated as  $\widehat{l}_j = \text{sign}(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)l_{i,j})$  and the sum of the value  $(2\theta_{i,j} - 1)^2$ 's for all winner  $w_i$ 's that execute task  $\tau_j$  should not be smaller than the threshold  $2 \ln(\frac{1}{\delta_j})$ . Intuitively, the larger the value  $(2\theta_{i,j} - 1)^2$  is, the more informative the label  $l_{i,j}$  will be to the platform. When the value  $(2\theta_{i,j} - 1)^2$  approaches 0, or equivalently  $\theta_{i,j}$  approaches 0.5, the label  $l_{i,j}$  will be closer to a random noise.

### 4.2.3 Auction Model

In the rest of the chapter, we will refer to any subset of tasks of  $\mathcal{T}$  as a *bundle*. Since in the MCS system considered in this chapter every worker bids on one bundle of tasks, we use *single-minded reverse combinatorial auction with heterogeneous cost (hSRC auction)*, formally defined in Definition 18, to model the problem.

**Definition 18** (hSRC Auction). *We define the single-minded reverse combinatorial auction with heterogeneous cost, namely hSRC auction, as follows. In the hSRC auction, any worker  $w_i$  has a*

set of  $K_i$  possible bidding bundles denoted as  $\mathcal{T}_i = \{\Gamma_{i,1}, \dots, \Gamma_{i,K_i}\}$ . For providing labels about all the tasks in each bundle  $\Gamma_{i,k} \in \mathcal{T}_i$ , the worker has a cost  $c_{i,k}$ . Furthermore, every worker  $w_i$  is only interested in one of the bundles in  $\mathcal{T}_i$ , denoted as  $\Gamma_i^*$  with cost  $c_i^*$ .

Noted that the hSRC auction defined in Definition 18 is a generalization of traditional single-minded combinatorial auctions, such as those in [13, 15]. Typically, in traditional single-minded combinatorial auctions, all the possible bidding bundles of a worker have the same cost. However, in our hSRC auction, the cost  $c_{i,k}$ 's for every bundle  $\Gamma_{i,k} \in \mathcal{T}_i$  do not necessarily have to be the same. In MCS systems, workers usually have different costs for executing different bundles, which makes our definition of hSRC auction more suitable to the problem studied in this chapter. In Definition 19, we define a worker's truthful bid.

**Definition 19** (Truthful Bid). *We define bid  $b_i^* = (\Gamma_i^*, c_i^*)$  which contains worker  $w_i$ 's true interested bundle  $\Gamma_i^*$  and the corresponding cost  $c_i^*$  as her truthful bid.*

In Definition 20 and 21, we present the formal definitions of a worker's utility and the platform's total payment.

**Definition 20** (Worker's Utility). *Suppose a worker  $w_i$  bids  $\Gamma_{i,k} \in \mathcal{T}_i$  in the hSRC auction. If she is a winner, she will be paid  $p_i$  by the platform. Otherwise, she will not be allocated any task and receives zero payment. Therefore, the utility of the worker  $w_i$  is*

$$u_i = \begin{cases} p_i - c_{i,k}, & \text{if } w_i \in \mathcal{S} \\ 0, & \text{otherwise} \end{cases}. \quad (4.3)$$

**Definition 21** (Platform's Payment). *The platform's total payment to all workers given the payment profile  $\mathbf{p}$  and the winner set  $\mathcal{S}$  is*

$$R(\mathbf{p}, \mathcal{S}) = \sum_{i:w_i \in \mathcal{S}} p_i. \quad (4.4)$$

#### 4.2.4 Design Objective

Since workers are strategic in our hSRC auction, it is possible that a worker could submit a bid different from the truthful bid defined in Definition 19 in order to obtain more utility. To address

this problem, one of our goals is to design a *truthful* mechanism, where every worker maximizes her utility by bidding her truthful bid regardless of other workers' bids. In practice, ensuring exact truthfulness for the hSRC auction is too restrictive. Therefore, we turn to a weaker but more practical notion of  $\gamma$ -*truthfulness in expectation* [42, 77], formally defined in Definition 22.

**Definition 22** ( $\gamma$ -truthfulness). *An hSRC auction is  $\gamma$ -truthful in expectation, or  $\gamma$ -truthful for short, if and only if for any bid  $b_i \neq b_i^*$  and any bid profile of other workers  $\mathbf{b}_{-i}$ , there is*

$$\mathbb{E}[u_i(b_i^*, \mathbf{b}_{-i})] \geq \mathbb{E}[u_i(b_i, \mathbf{b}_{-i})] - \gamma, \quad (4.5)$$

where  $\gamma$  is a small positive constant.

$\gamma$ -truthfulness ensures that no worker is able to make more than a slight  $\gamma$  gain in her expected utility by bidding untruthfully. Therefore, we reasonably assume that each worker  $w_i$  would bid her truthful bid  $b_i^*$ , if our hSRC auction satisfies  $\gamma$ -truthfulness. Apart from  $\gamma$ -truthfulness, another desirable property of our hSRC auction is *individual rationality*, which implies that no worker has negative utility. This property is crucial in that it prevents workers from being disincentivized by receiving negative utilities. We formally define this property in the following Definition 23.

**Definition 23** (Individual Rationality). *An hSRC auction is individual rational if and only if  $u_i \geq 0$  holds for every worker  $w_i \in \mathcal{N}$ .*

Simply paying workers according to the output payment profile of the auction poses threats to the privacy of workers' bids. Because the change in one worker's bid has the potential to shift the payment profile significantly, it is possible for a curious worker to infer other workers' bids from the different payments she receives in two rounds of auction. Therefore, we aim to design a differentially private mechanism [29, 77], formally defined in Definition 24.

**Definition 24** (Differential Privacy). *We denote the proposed hSRC auction as a function  $M(\cdot)$  that maps an input bid profile  $\mathbf{b}$  to a payment profile  $\mathbf{p}$ . Then,  $M(\cdot)$  is  $\epsilon$ -differentially private if and only if for any possible set of payment profiles  $\mathcal{A}$  and any two bid profiles  $\mathbf{b}$  and  $\mathbf{b}'$  that differ in only one bid, we have*

$$\Pr[M(\mathbf{b}) \in \mathcal{A}] \leq \exp(\epsilon) \Pr[M(\mathbf{b}') \in \mathcal{A}], \quad (4.6)$$

where  $\epsilon$  is a small positive constant usually referred to as privacy budget.

Differential privacy ensures that the change in any worker’s bid will not bring a significant change to the resulting payment profile. Hence, it is difficult for the curious workers to infer information about other workers’ bids from the outcome (i.e., payment profile) of the mechanism. In this chapter, to achieve differential privacy we introduce randomization to the outcome of our mechanism, similar to [77, 127, 128].

In short, we aim to design a  $\gamma$ -truthful, individual rational and  $\epsilon$ -differentially private incentive mechanism in this chapter.

### 4.3 Mathematical Formulation

In this section, we present our formal mathematical problem formulation.

In this chapter, we adopt the natural and commonly used *optimal single-price payment*, as in [40, 48, 127], as our optimal payment benchmark, because it is within a constant factor of the payment of any mechanism with price differentiation, as proved in [48]. More detailed discussion on the optimal benchmark will be provided in Section 4.5. In this chapter, therefore, we aim to design a single-price mechanism that pays every winner in  $\mathcal{S}$  according to the same price  $p$ .

To simplify our analysis, we assume that the possible values of the cost  $c_{i,k}$  for a worker  $w_i$  to execute a bundle of tasks  $\Gamma_{i,k} \in \mathcal{T}_i$  forms a *finite set*  $\mathcal{C}$ . The smallest and largest element in  $\mathcal{C}$  is  $c_{\min}$  and  $c_{\max}$  respectively. Given the winner set  $\mathcal{S}$ , for an individual rational single-price mechanism, the platform’s total payment is minimized if and only if the price  $p$  equals to the largest cost of the workers in  $\mathcal{S}$ , that is  $p = \max_{w_i \in \mathcal{S}} c_{i,k}$ . This is because otherwise the platform can always let  $p = \max_{w_i \in \mathcal{S}} c_{i,k}$  and obtain a smaller total payment while maintaining individual rationality. Therefore, the set  $\mathcal{P}$  containing all possible prices should satisfy that  $\mathcal{P} \subseteq \mathcal{C}$ . Furthermore, we define that a price  $p$  is feasible if and only if it is possible to select a set of winners  $\mathcal{S}$  among the workers with bidding prices  $\rho_i \leq p$  such that the error bound constraint defined in Equation (4.2) is satisfied for every task. Then, we define the price set  $\mathcal{P}$  as the set containing all values in the set  $\mathcal{C}$  that are feasible. Thus, obviously we have  $c_{\max} \in \mathcal{P} \subseteq \mathcal{C}$ .

Next, we formulate the total payment minimization (TPM) problem as the following optimization program.

**TPM Problem:**

$$\min \sum_{i:w_i \in \mathcal{N}} px_i \quad (4.7)$$

$$\text{s.t.} \quad \sum_{i:w_i \in \mathcal{N}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 x_i \geq 2 \ln \left( \frac{1}{\delta_j} \right), \quad \forall \tau_j \in \mathcal{T} \quad (4.8)$$

$$p - \rho_i x_i \geq 0, \quad \forall w_i \in \mathcal{N} \quad (4.9)$$

$$x_i \in \{0, 1\}, \quad \forall w_i \in \mathcal{N}, \quad p \in \mathcal{P} \quad (4.10)$$

**Constants.** The TPM problem takes as inputs the price set  $\mathcal{P}$ , workers' bid profile  $\mathbf{b}$ , the reliability level matrix  $\boldsymbol{\theta}$ , the vector  $\boldsymbol{\delta} = (\delta_1, \dots, \delta_K)$ , as well as the task and worker set  $\mathcal{T}$  and  $\mathcal{N}$ .

**Variables.** In the TPM problem, we have a vector of  $N$  binary variables  $\mathbf{x} = (x_1, \dots, x_N)$ . For every worker  $w_i \in \mathcal{N}$ , there is a binary variable  $x_i$  indicating whether this worker is selected as a winner. That is,

$$x_i = \begin{cases} 1, & \text{if } w_i \in \mathcal{S} \\ 0, & \text{otherwise} \end{cases}.$$

Furthermore, another variable in the TPM problem is the price  $p$ , which could take any value in the price set  $\mathcal{P}$ .

**Objective function.** Based on the definition of variables  $\mathbf{x}$  and  $p$ ,  $\sum_{i:w_i \in \mathcal{N}} px_i$  represents the platform's total payment to all the winners. Hence, the TPM problem aims to find the vector  $\mathbf{x}$  and price  $p$  that minimize the platform's total payment.

**Constraints.** Constraint (4.8) is exactly the error bound constraint represented by Inequality (4.2) in Lemma 5, which ensures that the aggregation error of every task  $\tau_j \in \mathcal{T}$  is no larger than the predefined threshold  $\delta_j$ . To simplify presentation, we introduce the following notations.  $q_{i,j} = (2\theta_{i,j} - 1)^2$ ,  $Q_j = 2 \ln \left( \frac{1}{\delta_j} \right)$ ,  $\mathbf{q} = [q_{i,j}] \in [0, 1]^{N \times K}$  and  $\mathbf{Q} = (Q_1, \dots, Q_K)$ , and thus, Constraint (4.8) can be simplified as the following Inequality (4.11).

$$\sum_{i:w_i \in \mathcal{N}, \tau_j \in \Gamma_i} q_{i,j} x_i \geq Q_j, \quad \forall \tau_j \in \mathcal{T}. \quad (4.11)$$

Furthermore, Constraint (4.9) ensures that for each worker  $w_i$ , we have  $p \geq \rho_i$ , if the worker is a

winner. This means that any feasible solution to the TPM problem satisfies individual rationality, if workers submit truthful bids. Apart from Constraint (4.8) and (4.9), we also consider two other inherent constraints, namely approximate truthfulness and differential privacy for workers' bids, which means that the hSRC auction that corresponds to any feasible solution is approximately truthful and differentially private. Due to the difficulty in mathematically formulating the two constraints, we take them into consideration without explicitly formulating them, in the TPM problem.

Next, in Theorem 18, we prove the NP-hardness of the TPM problem.

**Theorem 18.** *The TPM problem is NP-hard.*

*Proof.* Firstly, we transform the TPM problem into a modified TPM problem by fixing the price  $p = 1$  and relaxing Constraint (4.9), as well as the inherent approximate truthfulness and differential privacy constraints. Clearly, the modified TPM problem is a special case of the TPM problem. Thus, we turn to proving the NP-hardness of the modified TPM problem, instead.

We start our proof by introducing an instance of the minimum set cover (MSC) problem with a universe of  $K$  elements  $\mathcal{U} = \{\tau_1, \dots, \tau_K\}$  and a set of  $N$  sets  $\mathcal{H} = \{\Gamma_1, \dots, \Gamma_N\}$ . The objective of the MSC problem is to find the minimum-cardinality subset of  $\mathcal{H}$  whose union contains all the elements in  $\mathcal{U}$ . We construct an instance of the modified TPM problem based on this instance of the MSC problem. Firstly, we construct  $\Gamma'_i$  from  $\Gamma_i$  where every  $\tau_j \in \Gamma_i$  has  $h_{i,j} \in \mathbb{Z}^+$  copies in  $\Gamma'_i$ . Furthermore, we require that the selected sets cover every  $\tau_j \in \mathcal{U}$  for at least  $H_j$  times. Therefore, we get an instance of the modified TPM problem where  $\mathbf{q} = [h_{i,j}] \in (\mathbb{Z}^+)^{N \times K}$ ,  $\mathbf{Q} = (H_1, \dots, H_K)$  and the bidding bundle profile  $\mathbf{\Gamma} = (\Gamma'_1, \dots, \Gamma'_N)$ . In fact, the modified TPM problem represents a richer family of problems where elements in  $\mathbf{q}$  and  $\mathbf{Q}$  can be positive real values. Therefore, every instance of the NP-complete MSC problem is polynomial-time reducible to the modified TPM problem. The modified TPM problem, and thus, the original TPM problem, is NP-hard.  $\square$

## 4.4 Mechanism Design

Because of the NP-hardness of the TPM problem shown in Theorem 18, even given the price  $p$ , it is impossible to calculate in polynomial time the set of winners that minimize the platform's total

payment unless  $P = NP$ . Let alone we eventually need to select an optimal price from the price set  $\mathcal{P}$ . Therefore, we aim to design a *polynomial-time* mechanism that gives us *an approximately optimal total payment* with a *guaranteed approximation ratio* to the optimal total payment  $R_{\text{OPT}}$ . In addition, we also take into consideration the bid privacy preserving objective when designing the mechanism. We present our mechanism in Algorithm 7, namely *differentially private hSRC (DP-hSRC) auction*, that satisfies all our design objectives.

Algorithm 7 takes as inputs the privacy budget  $\epsilon$ , the cost upper bound  $c_{\text{max}}$ , the worker set  $\mathcal{N}$ , the task set  $\mathcal{T}$ , the price set  $\mathcal{P}$ , workers' bid profile  $\mathbf{b}$ , the  $\mathbf{q}$  matrix and the  $\mathbf{Q}$  vector. It outputs the winner set  $\mathcal{S}$  and the payment  $p$  paid to each winner. Firstly, it sorts workers according to the ascending order of their bidding prices such that  $\rho_1 \leq \rho_2 \leq \dots \leq \rho_N$  (line 1). Then, it initializes several parameters (line 2-5). It finds the minimum price  $p_{\text{min}}$  in  $\mathcal{P}$  (line 2) and the index  $i_{\text{min}}$  of the largest bidding price that does not exceed  $p_{\text{min}}$  (line 3). The algorithm constructs an index set  $\mathcal{I}$  containing all the integers from  $i_{\text{min}}$  to  $N$  (line 4). Set  $\mathcal{I}$  contains every worker index  $i$  such that a winner set  $\mathcal{S}_i$  is calculated among the workers with bidding prices that are not larger than  $\rho_i$ . In the last step of the initialization, the algorithm creates an extra bidding price  $\rho_{N+1}$  by adding a small positive constant  $\delta$  to  $c_{\text{max}}$  (line 5) to ensure that  $\rho_{N+1}$  is greater than  $\forall p \in \mathcal{P}$ . The purpose of creating  $\rho_{N+1}$  is to make sure that every price  $p \in \mathcal{P}$  is considered by line 14 and 15 in the main loop (line 6-15) for exactly once.

After the initialization phase, Algorithm 7 calculates the winner set for every possible price  $p \in \mathcal{P}$  (line 6-15). Intuitively, we need to calculate the winner set for every given price  $p \in \mathcal{P}$ . However, for all possible prices between two consecutive bidding prices, that is  $\forall p \in \mathcal{P} \cap [\rho_i, \rho_{i+1})$ , the winner sets are the same. Therefore, to reduce the computational complexity and remove its dependency on the number of possible prices (i.e.,  $|\mathcal{P}|$ ), we only need to calculate the winner set for every price  $p \in \{\rho_{i_{\text{min}}}, \rho_{i_{\text{min}}+1}, \dots, \rho_N\}$ . At the beginning of every iteration of the main loop (line 6-15), Algorithm 7 initializes the winner set  $\mathcal{S}_i$  as  $\emptyset$ , the residual  $\mathbf{Q}'$  vector as  $\mathbf{Q}$  and the candidate winner set  $\mathcal{N}'$  as every worker  $w_k$  with bidding price  $\rho_k$  that is not larger than  $\rho_i$  (line 7). The inner loop (line 8-13) is executed until the error bound constraints for all tasks are satisfied, or equivalently until  $\mathbf{Q}' = \mathbf{0}^{K \times 1}$ . In every iteration of the inner loop (line 8-13), the worker  $w_{i_{\text{max}}}$  that provides the most improvement to the feasibility of Constraint 4.8 is selected as the new winner

(line 9). Hence,  $w_{i_{\max}}$  is included in  $\mathcal{S}_i$  (line 10) and excluded from  $\mathcal{N}'$  (line 11). After  $w_{i_{\max}}$  is selected, the algorithm updates the residual  $\mathbf{Q}'$  vector (line 12-13).

---

**Algorithm 7:** DP-hSRC Auction

---

**Input:**  $\epsilon, c_{\max}, \mathbf{b}, \mathbf{q}, \mathbf{Q}, \mathcal{N}, \mathcal{T}, \mathcal{P}$ ;  
**Output:**  $\mathcal{S}, p$ ;

- 1 sort workers according to the ascending order of bidding prices such that  $\rho_1 \leq \rho_2 \leq \dots \leq \rho_N$ ;  
// Initialization
- 2  $p_{\min} \leftarrow \min_{p \in \mathcal{P}} p$ ;
- 3  $i_{\min} \leftarrow \arg \max_{i: \rho_i \leq p_{\min}} \rho_i$ ;
- 4  $\mathcal{I} \leftarrow \{i_{\min}, i_{\min} + 1, \dots, N\}$ ;  
// Add a small constant  $\delta > 0$  to  $c_{\max}$
- 5  $\rho_{N+1} \leftarrow c_{\max} + \delta$ ;  
// Calculates the winner sets
- 6 **foreach**  $i \in \mathcal{I}$  **do**
- 7      $\mathcal{S}_i \leftarrow \emptyset, \mathbf{Q}' \leftarrow \mathbf{Q}, \mathcal{N}' \leftarrow \{w_k | \rho_k \leq \rho_i\}$ ;
- 8     **while**  $\sum_{j: \tau_j \in \mathcal{T}} Q'_j \neq 0$  **do**
- 9          $i_{\max} = \arg \max_{i: w_i \in \mathcal{N}'} \sum_{j: \tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}$ ;
- 10          $\mathcal{S}_i \leftarrow \mathcal{S}_i \cup \{w_{i_{\max}}\}$ ;
- 11          $\mathcal{N}' \leftarrow \mathcal{N}' \setminus \{w_{i_{\max}}\}$ ;  
// Update the residual  $\mathbf{Q}'$  vector
- 12         **foreach**  $j$  s.t.  $\tau_j \in \mathcal{T}$  **do**
- 13              $Q'_j \leftarrow Q'_j - \min\{Q'_j, q_{i_{\max},j}\}$ ;
- // Assign the same winner set  $\mathcal{S}_i$  to every possible price in  $[\rho_i, \rho_{i+1})$
- 14         **foreach**  $p \in \mathcal{P} \cap [\rho_i, \rho_{i+1})$  **do**
- 15              $S(p) \leftarrow \mathcal{S}_i$ ;
- 16 randomly pick a price  $p$  according to the distribution

$$\Pr[p = x] = \frac{\exp\left(-\frac{\epsilon x |S(x)|}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y |S(y)|}{2Nc_{\max}}\right)}, \quad \forall x \in \mathcal{P};$$

// Obtain the corresponding winner set

- 17  $\mathcal{S} \leftarrow S(p)$ ;
- 18 **return**  $\{\mathcal{S}, p\}$ ;

---

To ensure differential privacy, we introduce randomization to the output price. We extend the exponential mechanism proposed in [77] and set the probability that the output price  $p$  of Algorithm 7 equals to a price  $x \in \mathcal{P}$  to be proportional to the value  $\exp\left(-\frac{\epsilon x |S(x)|}{2Nc_{\max}}\right)$ . That is,

$$\Pr[p = x] \propto \exp\left(-\frac{\epsilon x |S(x)|}{2Nc_{\max}}\right), \quad \forall x \in \mathcal{P}. \quad (4.12)$$

One important rationale of setting the probability of every possible price as the form in Equation (4.12) is that the price resulting in a smaller total payment will have a larger probability to be sampled. In fact, the probability increases exponentially with the decrease of the total payment and the distribution is substantially biased towards low total payment prices. Therefore, we can both achieve differential privacy and a guaranteed approximation to the optimal payment, as will be proved in Section 4.5. Algorithm 7 normalizes  $\exp\left(-\frac{\epsilon x|S(x)|}{2Nc_{\max}}\right)$  and randomly picks a price  $p$  according to the following distribution (line 16) defined in Equation (4.13).

$$\Pr[p = x] = \frac{\exp\left(-\frac{\epsilon x|S(x)|}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y|S(y)|}{2Nc_{\max}}\right)}, \quad \forall x \in \mathcal{P}. \quad (4.13)$$

After a price  $p$  is sampled, the winner set  $\mathcal{S}$  is set to be the one corresponding to  $p$ , namely  $S(p)$  (line 17). Finally, it returns the winner set  $\mathcal{S}$  and the price  $p$  (line 18).

## 4.5 Analysis

In this section, we provide formal theoretical analysis about the desirable properties of our DP-hSRC auction. First of all, we prove that the DP-hSRC auction is  $\epsilon$ -differentially private in Theorem 19.

**Theorem 19.** *The DP-hSRC auction is  $\epsilon$ -differentially private.*

*Proof.* We denote  $\mathbf{b}$  and  $\mathbf{b}'$  as two bid profiles that differ in only one worker's bid.  $\forall x \in \mathcal{P}$ , we have

$$\begin{aligned} \frac{\Pr[M(\mathbf{b}) = x]}{\Pr[M(\mathbf{b}') = x]} &= \frac{\exp\left(-\frac{\epsilon x|S(x)|}{2Nc_{\max}}\right)}{\exp\left(-\frac{\epsilon x|S'(x)|}{2Nc_{\max}}\right)} \cdot \frac{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y|S'(y)|}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y|S(y)|}{2Nc_{\max}}\right)} \leq \exp\left(\frac{\epsilon xN}{2Nc_{\max}}\right) \cdot \frac{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y(|S(y)|-N)}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y|S(y)|}{2Nc_{\max}}\right)} \\ &\leq \exp\left(\frac{\epsilon}{2}\right) \cdot \frac{\sum_{y \in \mathcal{P}} \exp\left(\frac{-\epsilon y|S(y)| + \epsilon c_{\max}N}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon y|S(y)|}{2Nc_{\max}}\right)} = \exp\left(\frac{\epsilon}{2}\right) \cdot \exp\left(\frac{\epsilon c_{\max}N}{2Nc_{\max}}\right) = \exp(\epsilon), \end{aligned}$$

which is equivalent to that

$$\Pr[M(\mathbf{b}) = x] \leq \exp(\epsilon)\Pr[M(\mathbf{b}') = x], \quad \forall x \in \mathcal{P}. \quad (4.14)$$

Therefore, based on Inequality (4.14), we have that

$$\Pr[M(\mathbf{b}) \in \mathcal{A}] \leq \exp(\epsilon)\Pr[M(\mathbf{b}') \in \mathcal{A}], \forall \mathcal{A} \subseteq \mathcal{P},$$

and we arrive at the conclusion that the DP-hSRC auction is  $\epsilon$ -differentially private.  $\square$

We introduce the notation that  $\Delta c = c_{\max} - c_{\min}$ . Based on Theorem 19, we prove in Theorem 20 that the DP-hSRC auction is  $\epsilon\Delta c$ -truthful.

**Theorem 20.** *The DP-hSRC auction is  $\epsilon\Delta c$ -truthful.*

*Proof.* Similar to the proof of Theorem 19, we use  $\mathbf{b}$  and  $\mathbf{b}'$  to denote two bid profiles that differ in only one worker's bid. An equivalent form of Equation (4.14) proved in Theorem 19 is

$$\Pr[M(\mathbf{b}) = x] \geq \exp(-\epsilon)\Pr[M(\mathbf{b}') = x], \forall x \in \mathcal{P}.$$

Therefore, the expectation of any worker  $w_i$ 's utility taken over the output price distribution of the DP-hSRC auction mechanism  $M(\cdot)$  given in Algorithm 7 satisfies that

$$\begin{aligned} \mathbb{E}_{x \sim M(\mathbf{b})}[u_i(x)] &= \sum_{x \in \mathcal{P}} u_i(x)\Pr[M(\mathbf{b}) = x] \geq \sum_{x \in \mathcal{P}} u_i(x)\exp(-\epsilon)\Pr[M(\mathbf{b}') = x] \\ &= \exp(-\epsilon)\mathbb{E}_{x \sim M(\mathbf{b}')}[u_i(x)] \geq (1 - \epsilon)\mathbb{E}_{x \sim M(\mathbf{b}')}[u_i(x)] \\ &= \mathbb{E}_{x \sim M(\mathbf{b}')}[u_i(x)] - \epsilon\mathbb{E}_{x \sim M(\mathbf{b}')}[u_i(x)]. \end{aligned}$$

Since the maximum price in  $\mathcal{P}$  is  $c_{\max}$  and the minimum possible cost for a worker is  $c_{\min}$ , we have that  $u_i(x) \leq c_{\max} - c_{\min}, \forall x \in \mathcal{P}$ . Therefore, we have

$$\mathbb{E}_{x \sim M(\mathbf{b}')}[u_i(x)] \leq c_{\max} - c_{\min} = \Delta c,$$

and thus,

$$\mathbb{E}_{x \sim M(\mathbf{b})}[u_i(x)] \geq \mathbb{E}_{x \sim M(\mathbf{b}')}[u_i(x)] - \epsilon\Delta c.$$

Therefore, we conclude that the DP-hSRC auction is  $\epsilon\Delta c$ -truthful.  $\square$

Theorem 20 basically states that the proposed DP-hSRC auction upper bounds a worker's gain in her expected utility to bid untruthfully by  $\epsilon\Delta c$ . Therefore, we reasonably assume that each worker would bid truthfully in our DP-hSRC auction. Note that our DP-hSRC auction is  $\epsilon\Delta c$ -truthful in both the bidding bundle and price, namely any worker  $w_i$  bids her truthful bid  $b_i^* = (\Gamma_i^*, c_i^*)$ . In Theorem 21, we prove that our DP-hSRC auction is individual rational.

**Theorem 21.** *The DP-hSRC auction is individual rational.*

*Proof.* In every iteration of the main loop in Algorithm 7 (line 6-15), the candidate winner set  $\mathcal{N}'$  is initialized as those workers whose bidding prices (i.e.,  $\rho_k$ ) are not larger than the given price  $p = \rho_i$  (line 7). Furthermore, we have proved in Theorem 20 that every worker  $w_k$  bids truthfully, i.e.,  $\rho_k = c_k$ . It means that for any given price  $p$  the winners are selected among the workers (i.e.,  $w_k$ ) such that  $c_k \leq p$ . As a consequence, any winner  $w_k$ 's utility satisfies  $u_k = p - c_k \geq 0$  and any loser's utility equals to 0. Therefore, we conclude that the DP-hSRC auction is individual rational.  $\square$

Next, we provide our analysis about the algorithmic properties of the proposed DP-hSRC auction regarding the computational complexity and its approximation ratio to the optimal total payment in Theorem 22 and 23. Firstly, we analyze the computational complexity of our DP-hSRC auction in the following Theorem 22.

**Theorem 22.** *The computational complexity of the proposed DP-hSRC auction is  $O(N^2K)$ .*

*Proof.* The computational complexity of Algorithm 7 is dominated by the main loop (line 6-15), which terminates in worst case after  $N$  iterations. Furthermore, in every iteration of the inner loop (line 8-13), one worker is selected as a new winner. Thus, the inner loop also terminates in worst case after  $N$  iterations. Besides, within the inner loop, after a winner is selected the algorithm updates the  $Q'_j$  value for every task  $\tau_j \in \mathcal{T}$  in the worst case. Therefore, the overall computational complexity of the DP-hSRC auction is  $O(N^2K)$ .  $\square$

As proved in Theorem 22, our DP-hSRC auction described in Algorithm 7 has polynomial-time computational complexity depending on the number of workers  $N$  and the number of tasks  $K$ . Furthermore, the computational complexity provided in Theorem 22 does not depend on the cardinality of the possible price set  $\mathcal{P}$ , namely  $|\mathcal{P}|$ .

Given a price  $p$  and all the other parameters, we use  $S_{\text{OPT}}(p)$  to denote the winner set with the minimum-cardinality such that Constraint (4.8) and (4.9) are satisfied. Thus, the optimal total payment which we use as the optimal benchmark to compare Algorithm 7's total payment with in this chapter, denoted as  $R_{\text{OPT}}$ , is then calculated as

$$R_{\text{OPT}} = \min_{p \in \mathcal{P}} p |S_{\text{OPT}}(p)|. \quad (4.15)$$

Note that  $R_{\text{OPT}}$  is in fact the optimal total payment after we relax the approximate truthfulness and differential privacy constraints, which is clearly smaller than the actual optimal total payment of the TPM problem. Thus, it is fair to compare the total payment generated by Algorithm 7 with  $R_{\text{OPT}}$ .

Before we analyze the approximation ratio between Algorithm 7's total payment and the optimal total payment  $R_{\text{OPT}}$  in Theorem 23, we introduce Lemma 6 which is borrowed from [50] (Theorem 5 in [50]). We define the unit measure of every element in  $\mathbf{q}$  and  $\mathbf{Q}$  as  $\Delta q$  and introduce additionally the following two notations, i.e.,  $\beta = \max_{i:w_i \in \mathcal{N}} \sum_{j:\tau_j \in \Gamma_i} q_{i,j}$  and  $m = \frac{1}{\Delta q} \sum_{j:\tau_j \in \mathcal{T}} Q_j$ .

**Lemma 6.** *Given any price  $p \in \mathcal{P}$ , we have that the cardinality of the winner set returned by the proposed DP-hSRC auction  $S(p)$  and that of the minimum-cardinality winner set  $S_{\text{OPT}}(p)$  satisfies that*

$$|S(p)| \leq 2\beta H_m |S_{\text{OPT}}(p)|. \quad (4.16)$$

The relationship between the cardinality of the two sets  $S(p)$  and  $S_{\text{OPT}}(p)$  given in Lemma 6 is an important intermediary result that will be utilized in the proof of the following Theorem 23, which shows the approximation ratio of the total payment generated by the DP-hSRC auction to the optimal total payment.

**Theorem 23.** *We use  $R(x)$  to denote the total payment given by Algorithm 7 for any price  $x \in \mathcal{P}$ . The expected total payment generated by the DP-hSRC auction, denoted by  $\mathbb{E}_{x \in \mathcal{P}}[R(x)]$ , and the optimal payment  $R_{\text{OPT}}$  satisfies that*

$$\mathbb{E}_{x \in \mathcal{P}}[R(x)] \leq 2\beta H_m R_{\text{OPT}} + \frac{6Nc_{\max}}{\epsilon} \ln \left( e + \frac{\epsilon |\mathcal{P}| \beta H_m R_{\text{OPT}}}{c_{\min}} \right).$$

*Proof.* We use  $R_{\min}$  and  $R_{\max}$  to denote the minimum and maximum total payment generated by Algorithm 7 and we define the following sets  $\mathcal{B}_t = \{x | R(x) > R_{\min} + t\}$ ,  $\bar{\mathcal{B}}_t = \{x | R(x) \leq R_{\min} + t\}$  and  $\mathcal{B}_{2t} = \{x | R(x) > R_{\min} + 2t\}$  for some constant  $t > 0$ . Then, we have

$$\begin{aligned}
\Pr[x \in \mathcal{B}_{2t}] &\leq \frac{\Pr[x \in \mathcal{B}_{2t}]}{\Pr[x \in \bar{\mathcal{B}}_t]} \\
&= \frac{\sum_{x \in \mathcal{B}_{2t}} \frac{\exp\left(-\frac{\epsilon R(x)}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon R(y)}{2Nc_{\max}}\right)}}{\sum_{x \in \bar{\mathcal{B}}_t} \frac{\exp\left(-\frac{\epsilon R(x)}{2Nc_{\max}}\right)}{\sum_{y \in \mathcal{P}} \exp\left(-\frac{\epsilon R(y)}{2Nc_{\max}}\right)}} = \frac{\sum_{x \in \mathcal{B}_{2t}} \exp\left(-\frac{\epsilon R(x)}{2Nc_{\max}}\right)}{\sum_{x \in \bar{\mathcal{B}}_t} \exp\left(-\frac{\epsilon R(x)}{2Nc_{\max}}\right)} \leq \frac{|\mathcal{B}_{2t}| \exp\left(-\frac{\epsilon(R_{\min}+2t)}{2Nc_{\max}}\right)}{|\bar{\mathcal{B}}_t| \exp\left(-\frac{\epsilon(R_{\min}+t)}{2Nc_{\max}}\right)} \\
&= \frac{|\mathcal{B}_{2t}|}{|\bar{\mathcal{B}}_t|} \exp\left(-\frac{\epsilon t}{2Nc_{\max}}\right).
\end{aligned}$$

Then, we can calculate  $\mathbb{E}_{x \in \mathcal{P}}[R(x)]$  as follows.

$$\begin{aligned}
\mathbb{E}_{x \in \mathcal{P}}[R(x)] &= \sum_{x \in \bar{\mathcal{B}}_{2t}} R(x) \Pr[p = x] + \sum_{x \in \mathcal{B}_{2t}} R(x) \Pr[p = x] \leq R_{\min} + 2t + R_{\max} \frac{|\mathcal{B}_{2t}|}{|\bar{\mathcal{B}}_t|} \exp\left(-\frac{\epsilon t}{2Nc_{\max}}\right) \\
&\leq R_{\min} + 2t + R_{\max} |\mathcal{P}| \exp\left(-\frac{\epsilon t}{2Nc_{\max}}\right).
\end{aligned}$$

Therefore, for any  $t \geq \ln\left(\frac{R_{\max} |\mathcal{P}|}{t}\right) \cdot \frac{2Nc_{\max}}{\epsilon}$ , we have

$$\mathbb{E}_{x \in \mathcal{P}}[R(x)] \leq R_{\min} + 3t. \quad (4.17)$$

If we let  $t = \ln\left(e + \frac{\epsilon |\mathcal{P}| R_{\max}}{2Nc_{\max}}\right) \cdot \frac{2Nc_{\max}}{\epsilon} \geq \frac{2Nc_{\max}}{\epsilon}$ , we have that

$$\ln\left(\frac{R_{\max} |\mathcal{P}|}{t}\right) \cdot \frac{2Nc_{\max}}{\epsilon} \leq \ln\left(e + \frac{R_{\max} |\mathcal{P}| \epsilon}{2Nc_{\max}}\right) \cdot \frac{2Nc_{\max}}{\epsilon} = t.$$

Therefore, we can simply let  $t = \ln\left(e + \frac{\epsilon |\mathcal{P}| R_{\max}}{2Nc_{\max}}\right) \cdot \frac{2Nc_{\max}}{\epsilon}$ , and substitute  $t$  into Equation (4.17).

Then, we have that

$$\mathbb{E}_{x \in \mathcal{P}}[R(x)] \leq R_{\min} + \ln\left(e + \frac{\epsilon |\mathcal{P}| R_{\max}}{2Nc_{\max}}\right) \cdot \frac{6Nc_{\max}}{\epsilon}.$$

Furthermore, since  $R_{\max} \leq \frac{c_{\max}}{c_{\min}} NR_{\min}$ , we have that

$$\mathbb{E}_{x \in \mathcal{P}} [R(x)] \leq R_{\min} + \ln \left( e + \frac{\epsilon |\mathcal{P}| R_{\min}}{2c_{\min}} \right) \cdot \frac{6Nc_{\max}}{\epsilon}.$$

Suppose the optimal total payment  $R_{\text{OPT}}$  is achieved when the price  $p = p^*$ , i.e.,  $R_{\text{OPT}} = p^* |S_{\text{OPT}}(p^*)|$ . Then, we have

$$R_{\min} \leq p^* |S(p^*)| \leq 2\beta H_m p^* |S_{\text{OPT}}(p^*)| = 2\beta H_m R_{\text{OPT}}.$$

Finally, we arrive at the conclusion that

$$\mathbb{E}_{x \in \mathcal{P}} [R(x)] \leq 2\beta H_m R_{\text{OPT}} + \frac{6Nc_{\max}}{\epsilon} \ln \left( e + \frac{\epsilon |\mathcal{P}| \beta H_m R_{\text{OPT}}}{c_{\min}} \right)$$

and we finish the proof of Theorem 23. □

Thus far, the approximation ratio derived in Theorem 23 is the best one we have found. We leave the proof of its tightness or the derivation of a better one, as well as the calculation of a lower bound for the approximation ratio, in our future work.

## 4.6 Performance Evaluation

In this section, we present the baseline methods that we use in the simulation, as well as the simulation settings and results.

### 4.6.1 Baseline Method

Firstly, we compare the expected total payment of the DP-hSRC auction with the optimal total payment  $R_{\text{OPT}}$ . Instead of solving the TPM problem approximately using the method in Algorithm 7 (line 6-15), the exact optimal solution  $S_{\text{OPT}}(p)$  to the TPM problem given any fixed price  $p \in \mathcal{P}$  is calculated. Then, the optimal total payment  $R_{\text{OPT}} = \min_{p \in \mathcal{P}} p |S_{\text{OPT}}(p)|$  is derived by iterating over every possible price  $p \in \mathcal{P}$ .

Furthermore, we compare our DP-hSRC auction with a baseline auction mechanism. For any fixed price  $p \in \mathcal{P}$ , the baseline auction selects the workers in  $\mathcal{N}' = \{w_i | \rho_i \leq p\}$  as winners according

to the descending order of the value  $\sum_{j:\tau_j \in \Gamma_i} q_{i,j}$  until the error bound constraints of all tasks are satisfied. Then, a price  $p$  is picked randomly using the same method in Algorithm 7 (line 16). It is easily verifiable that the baseline auction is also  $\epsilon$ -differentially private,  $\epsilon\Delta c$ -truthful and individual rational.

### 4.6.2 Simulation Settings

Setting	$\epsilon$	$c_{\min}$	$c_{\max}$	$ \Gamma_i^* $	$\theta_{i,j}$	$\delta_j$	$N$	$K$
4.I	0.1	10	60	[10, 20]	[0.1, 0.9]	[0.1, 0.2]	[80, 140]	30
4.II	0.1	10	60	[10, 20]	[0.1, 0.9]	[0.1, 0.2]	120	[20, 50]
4.III	0.1	10	60	[50, 150]	[0.1, 0.9]	[0.1, 0.2]	[800, 1400]	200
4.IV	0.1	10	60	[50, 150]	[0.1, 0.9]	[0.1, 0.2]	1000	[200, 500]

Table 4.1: Simulation setting 4.I-4.IV

In Table 4.1, we present the simulation settings. In setting 4.I, we fix the number of tasks as 30 and vary the number of workers from 80 to 140. The privacy budget  $\epsilon$  is set to be 0.1 and  $c_{\min}$  and  $c_{\max}$  is 10 and 60 respectively. Every worker  $w_i$ 's cost  $c_i^*$  for her interested bundle  $\Gamma_i^*$  is chosen uniformly at random from the numbers spaced at the interval of 0.1 in the range [10, 60].  $|\Gamma_i^*|$ ,  $\theta_{i,j}$ , and  $\delta_j$  are generated uniformly at random from the intervals given in Table 4.1. Furthermore, the price set  $\mathcal{P}$  consists of all numbers spaced at the interval of 0.1 in the range [35, 60]. In setting 4.II, we fix the number of workers as 120 and vary the number of tasks from 20 to 50. All the other parameters are the same as those in setting 4.I. In setting 4.III and 4.IV, the parameter  $\epsilon$ ,  $c_{\min}$ ,  $c_{\max}$ ,  $|\Gamma_i^*|$ ,  $\theta_{i,j}$ ,  $\delta_j$ ,  $c_i^*$ , and  $\mathcal{P}$  are generated using the same method as in the previous two settings. The difference is that we increase the input size of the settings. In setting 4.III, we fix the number of tasks as 200 and vary the number of workers from 800 to 1400, whereas in setting 4.IV, we fix the number of workers as 1000 and vary the number of tasks from 200 to 500. Note that we leave the study of the values of these parameters in real-world scenarios in our future work. Moreover, all the optimal solutions to the TPM problem are calculated using the GUROBI optimization solver [6].

### 4.6.3 Simulation Results

In Figure 4.1 and 4.2, for every given worker and task number, we sample a price from the price distribution derived by the DP-hSRC auction and the baseline auction, respectively, for 10000 times. The corresponding mean and standard deviation of the platform's total payment calculated

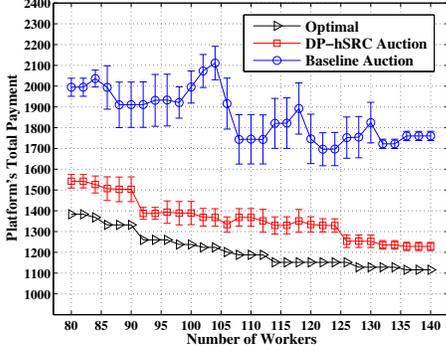


Figure 4.1: Platform’s total payment (setting 4.I)

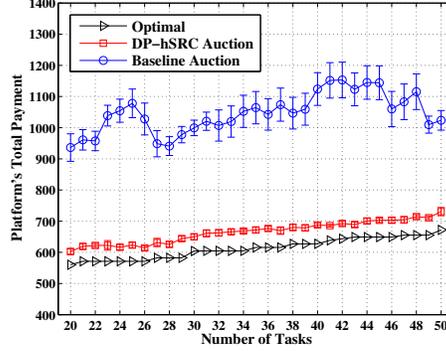


Figure 4.2: Platform’s total payment (setting 4.II)

using these price samples are plotted in Figure 4.1 and 4.2. From these two figures, we observe that the platform’s average total payment of the DP-hSRC auction is far better than that of the baseline auction and fairly close to the optimal total payment  $R_{\text{OPT}}$ . Note that the nonsmoothness of the curves in Figure 4.1 and 4.2, as well as those in the forthcoming Figure 4.3 and 4.4, is due to the randomness in generating the problem instances.

$N$	80	88	96	104	112	120	128	136
DP-hSRC	0.156	0.158	0.157	0.161	0.161	0.156	0.165	0.159
Optimal	6.479	11.86	30.83	410.7	897.1	2337	2310	6139
$K$	20	24	28	32	36	40	44	48
DP-hSRC	0.152	0.153	0.153	0.158	0.157	0.157	0.160	0.162
Optimal	13.33	44.04	396.4	395.9	539.7	735.5	1188	2661

Table 4.2: Execution time (seconds) for setting 4.I and 4.II

In Table 4.2, we compare the execution time of the DP-hSRC auction and the algorithm that computes the optimal total payment  $R_{\text{OPT}}$ . From this table, we can observe that the DP-hSRC auction executes in significantly less time than the optimal algorithm. Furthermore, the execution time of the optimal algorithm becomes excessively long with large numbers of tasks and workers so that it is infeasible in practice. In contrast, regardless of the growth of the number of users and tasks, the DP-hSRC auction keeps low execution time. Hence, the DP-hSRC auction is much more computationally efficient than the optimal algorithm.

In Figure 4.3 and 4.4, we consider setting 4.III and IV given in Table 4.1. Setting 4.III and 4.IV have much more numbers of workers and tasks than setting 4.I and 4.II. Under setting 4.III and 4.IV, the scales of the problem have become so large that make it infeasible for the optimal algorithm to return the optimal results in reasonable time. In contrast, in Figure 4.3 and 4.4, we

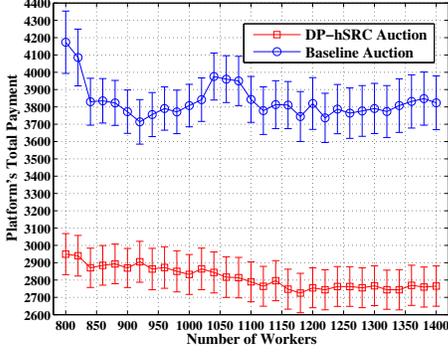


Figure 4.3: Platform's total payment (setting 4.III)

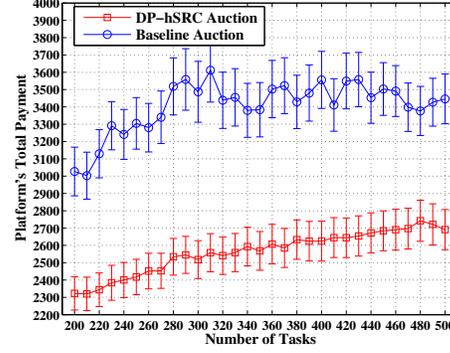


Figure 4.4: Platform's total payment (setting 4.IV)

demonstrate that our DP-hSRC auction is still able to generate total payment far better than the baseline auction under setting 4.III and 4.IV.

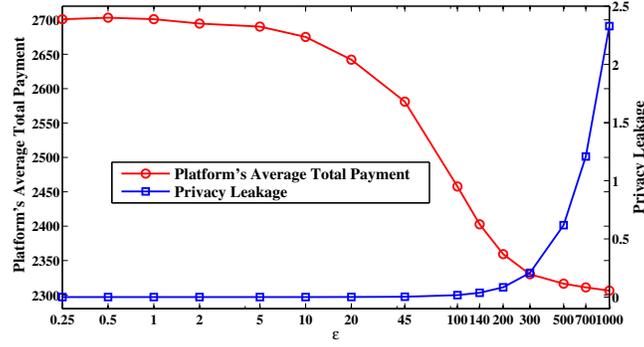


Figure 4.5: Trade-off between the platform's total payment and privacy leakage

In Figure 4.5, we plot the platform's average total payment and the privacy leakage of the DP-hSRC auction with the increasing of the privacy budget  $\epsilon$ . For any fixed  $\epsilon$ , we define the privacy leakage of the DP-hSRC auction as follows in Definition 25.

**Definition 25** (Privacy Leakage). *Suppose the two bid profiles  $\mathbf{b}$  and  $\mathbf{b}'$  that differ in only one worker's bid result in price distributions with probability mass functions (PMFs)  $P$  and  $P'$ . The privacy leakage of the two bid profiles is defined as the Kullback-Leibler (KL) divergence [61] of the two distributions represented as follows.*

$$\text{Privacy Leakage} = D_{\text{KL}}(P||P') = \sum_{x \in \mathcal{P}} P(x) \ln \left( \frac{P(x)}{P'(x)} \right).$$

The KL divergence captures the statistical difference of the two distributions  $P$  and  $P'$ . The

larger the statistical difference is, the easier the two bid profiles  $\mathbf{b}$  and  $\mathbf{b}'$  will be distinguished and thus, the more the privacy leakage is. From Figure 4.5, we observe that as the decrease of  $\epsilon$ , the privacy leakage decreases. Also, such improvement in privacy protection comes at a cost of the increased total payment of the platform shown in Figure 4.5. Therefore, Figure 4.5 illustrates the trade-off between the platform’s total payment and the privacy leakage of the DP-hSRC auction.

## 4.7 Related Work

As previously mentioned, most of the existing auction-based incentive mechanisms [32,33,35,37,50,53,54,60,62,71,72,75,97,100,106,107,112,113,115–118,120–122,124] fail to consider the preservation of workers’ bid privacy. In this chapter, different from most of these past literature, we incorporate the notion of *differential privacy* [29,77] and design a differentially private incentive mechanism for MCS systems that protects *workers’ bid privacy*.

Although several prior work [53,66,67,71,86,97,103–105] design, as well, privacy-preserving incentive mechanisms, our work is different from them in various aspects. Similar to our work, authors in [71] also design a bid privacy-preserving incentive mechanism. However, they consider to alleviate privacy leakage to workers bids from the public winner set, other than from payments considered in this chapter. Instead of bid privacy, [53,103–105] focus on protecting workers’ data privacy. [66,67,86] do not adopt game-theoretic methods, and thus cannot tackle workers’ strategic behavior. Instead, they adopt credit systems [66,67] and untraceable electronic currency [86]. Furthermore, the method of encrypting workers’ bids in [97] cannot address the problem of preventing curious workers from inferring information about other workers’ bids from the payments they receive. Note that existing models on privacy-preserving mechanism design [34,39,42,59,77,84,85,109,127,128] cannot be readily applied in our scenario, as they cannot tackle the coverage (like) constraints in our problem setting.

## 4.8 Conclusion

In this chapter, motivated by the need for the protection of workers’ privacy in MCS systems, we develop a differentially private incentive mechanism to incentivize worker participation without

disclosing their sensitive bid information. The proposed mechanism is based on a novel design of single-minded reverse combinatorial auction with heterogeneous cost, and thus bears several advantageous properties including approximate truthfulness, individual rationality, and computational efficiency. We conduct both theoretical analysis and extensive simulations to show that the proposed mechanism minimizes the expected total payment with a guaranteed approximation ratio to the optimal total payment.

## Chapter 5

# Incentivize Privacy-Preserving Data Aggregation in MCS Systems

### 5.1 Introduction

In real practice, apart from an incentive mechanism, an MCS system usually contains some other components which interact with the incentive mechanism and thus may affect its performance, such as *data aggregation* component that aggregates workers' data and *data perturbation* component that protects workers' privacy. Therefore, different from the isolated design of the incentive mechanism in most of the past literature [20, 22, 27, 28, 31–33, 35, 37, 43–45, 50, 51, 54, 57, 58, 60, 62, 66, 67, 71–75, 82, 86, 88, 89, 91, 92, 95, 97, 98, 100, 106, 107, 110–113, 115–124], we capture such interactive effect, and propose INCEPTION<sup>1</sup>, a novel MCS system framework with an *integrated design* of the incentive, data aggregation, and data perturbation mechanism. Below, we would like to shed some light on our design philosophy.

On one hand, the design of the incentive mechanism highly depends on how the platform aggregates workers' data. The sensory data provided by individual workers are usually not reliable due to various factors (e.g., poor sensor quality, environment noise, lack of sensor calibration). Therefore, the platform (i.e., a cloud-based central server) has to properly aggregate workers' noisy and even conflicting data so as to cancel out the possible errors from individual workers. Intuitively, if workers' data are aggregated using naive methods (e.g., average and voting) that regard all workers equally, the incentive mechanism does not need to view them differently in terms of their reliability. However, a *weighted aggregation* scheme that assigns higher weights to workers with higher reliability is much more favorable in that it makes the aggregated results closer to the data provided by more reliable workers. Therefore, we propose a weighted data aggregation mechanism that incorporates workers' diverse *reliability* to calculate highly accurate aggregated

---

<sup>1</sup>The name INCEPTION comes from INCEtive, Privacy, and data aggregaTION.

results. Accordingly, we *jointly design* our incentive mechanism which selects workers who are more likely to provide *reliable* data.

On the other hand, the incentive mechanism also needs to consider the *leakage of workers' privacy*, because it incurs costs which should be compensated as well. In many MCS applications, the platform usually publishes the aggregated results, which are oftentimes beneficial to the community or society, but jeopardizes workers' privacy. Although the platform can be considered to be trusted, there exist adversaries highly motivated to infer workers' data, which contain their sensitive and private information, from the published results. For example, publishing aggregated health data, such as treatment outcomes, improves people's awareness about the effects of new drugs and medical devices, but poses threats to the privacy of participating patients. Geotagging campaigns provide timely and accurate localization of physical objects (e.g, automated external defibrillator, litter, pothole), however, at the risk of leaking workers' sensitive location information. A high possibility for excessively large privacy leakage will deter workers from participating in the first place, even though they are promised to be compensated for their privacy costs. Therefore, we propose a *data perturbation mechanism* that reduces workers' privacy leakage to a reasonable degree by adding carefully controlled random noises to the original aggregated results, and *jointly design* the incentive mechanism that compensates their costs for not only sensing but also the remaining privacy leakage.

In summary, this chapter makes the following contributions.

- In this chapter, we propose INCEPTION, a novel MCS system framework that integrates an incentive, a data aggregation, and a data perturbation mechanism. Such an integrated design, which captures the *interactive effects* among these mechanisms, is much more challenging than designing them separately.
- INCEPTION has a reverse auction-based incentive mechanism that selects *reliable* workers and compensates their costs for both *sensing* and *privacy leakage*, which also satisfies *truthfulness* and *individual rationality*, and *minimizes the platform's total payment* for worker recruiting with a guaranteed approximation ratio.
- The data aggregation mechanism of INCEPTION also incorporates workers' *reliability* and

generates *highly accurate* aggregated results.

- Its data perturbation mechanism ensures satisfactory guarantee for the protection of *workers' privacy*, as well as the *accuracy* of the final perturbed results.

The rest of the chapter is organized as follows. We introduce the preliminaries in Section 5.2, and present the design details of INCEPTION in Section 5.3. Next, we present the results of our extensive simulation in Section 5.4, and summarize the related work in Section 5.5. Finally, we conclude this chapter in Section 5.6.

## 5.2 Preliminaries

In this section, we give an overview of INCEPTION, and describe the task model, reliability level model, auction model, as well as design objectives.

### 5.2.1 System Overview

INCEPTION is an MCS system framework consisting of a cloud-based platform and a set of  $N$  participating workers, denoted as  $\mathcal{N} = \{w_1, \dots, w_N\}$ . The platform hosts a set of  $K$  sensing tasks, denoted as  $\mathcal{T} = \{\tau_1, \dots, \tau_K\}$ , where each task  $\tau_j \in \mathcal{T}$  requires workers to locally sense a specific object or phenomenon, and report the sensory data to the platform. If worker  $w_i$  is selected to execute task  $\tau_j$ , she will provide her data  $x_{i,j}$  to the platform. We define  $\mathbf{x} = [x_{i,j}] \in (\mathcal{X} \cup \{\perp\})^{N \times K}$  as the matrix containing all workers' data, where  $\mathcal{X}$  denotes the range of tasks' sensory data, and  $x_{i,j} = \perp$  means that task  $\tau_j$  is not executed by worker  $w_i$ . To cancel out the errors from individual workers, for every task  $\tau_j \in \mathcal{T}$ , the platform aggregates workers' data into an aggregated result, denoted as  $x_j$ , which is used as an estimate of the task's ground truth value  $x_j^*$ , unknown to both the platform and the workers.

In our model, the platform publishes the aggregated results (e.g., locations of automated external defibrillators, litter, potholes) to the community or society. However, directly publishing them impairs workers' privacy. Therefore, the platform publishes the perturbed results after adding random noises to the original ones, and ensures  $\epsilon$ -differential privacy defined in Definition 26 (adapted from [29]).

**Definition 26** (Differential Privacy). We denote  $M : (\mathcal{X} \cup \{\perp\})^{N \times K} \rightarrow \mathbb{R}^{K \times 1}$  as a mechanism that maps any input data matrix to a perturbed result vector. Then, the mechanism  $M$  is  $\epsilon$ -differentially private if and only if for any two data matrices  $\mathbf{x}$  and  $\mathbf{x}'$  that differ in only one entry and any  $\mathcal{A} \subseteq \mathbb{R}^{K \times 1}$ , we have

$$\Pr[M(\mathbf{x}) \in \mathcal{A}] \leq \exp(\epsilon) \Pr[M(\mathbf{x}') \in \mathcal{A}], \quad (5.1)$$

where  $\epsilon$  is a small positive number usually referred to as privacy budget.

The framework of INCEPTION is illustrated in Figure 5.1, and its workflow<sup>2</sup> is described as follows.

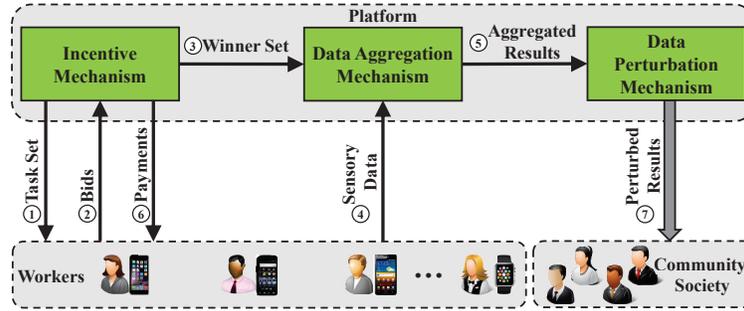


Figure 5.1: Framework of INCEPTION<sup>3</sup>

- Firstly, the platform announces the set of sensing tasks  $\mathcal{T}$  and an upper bound of the privacy budget  $\epsilon$ , such as  $\epsilon \leq 0.5$ , to workers (step ①).
- **Incentive Mechanism.** Then, the platform starts the reverse auction-based incentive mechanism, where it acts as the *auctioneer*, to purchase data from participating workers, who act as *bidders*. Every worker  $w_i$  submits to the platform her bid  $b_i = (\Gamma_i, b_i^s, b_i^p)$  which is a triple containing the set of sensing tasks  $\Gamma_i$  she wants to execute, as well as her bidding prices for executing them  $b_i^s$  and unit privacy loss  $b_i^p$  (step ②). Based on workers' bids, the platform determines the set of winners  $\mathcal{S} \subseteq \mathcal{N}$  and the payment  $p_i$  to every winner  $w_i$  (step ③). Losers of the auction do not execute tasks and receive no payments. We denote workers' bid and payment profile as  $\mathbf{b} = (b_1, \dots, b_N)$  and  $\mathbf{p} = (p_1, \dots, p_N)$ , respectively.

<sup>2</sup>Note that we are specifically interested in the scenario where all workers and tasks arrive at same time. We leave the investigation of the online scenario where workers and tasks arrive sequentially in an online manner in our future work.

<sup>3</sup>In this figure, circled numbers represent the order of the events.

- **Data Aggregation Mechanism.** Next, the platform collects winners’ sensory data (step ④) and calculates an aggregated result  $x_j$  for each task  $\tau_j$  (step ⑤).
- After collecting workers’ data, the platform pays workers according to  $\mathbf{p}$  and reveals to them the exact value of the privacy budget  $\epsilon$  (step ⑥), such as  $\epsilon = 0.25$ . The design rationale for keeping the exact value of  $\epsilon$  confidential to workers at the bidding stage and revealing it together with the payments is described in detail in Section 5.3.2.3.
- **Data Perturbation Mechanism.** Finally, the platform adds random noises to the original aggregated results and publishes the perturbed ones (step ⑦). We use  $\hat{x}_j$  to denote the perturbed result for task  $\tau_j$ .

### 5.2.2 Task Model

In this chapter, we are specifically interested in MCS systems that collect *heterogeneous* types of sensory data from participating workers, which are ubiquitous in practice. That is, some of the tasks held by the platform (e.g., environmental monitoring) require workers to submit *continuous data* (e.g., temperature, humidity), whereas others (e.g., geotagging) collect *categorical data* (e.g., whether or not potholes exist on a specific road segment). In the rest of this chapter, we refer to the former as *continuous tasks*, and the latter as *categorical tasks*. Furthermore, we denote  $\mathcal{T}_{\text{con}}$  and  $\mathcal{T}_{\text{cat}}$  as the set of continuous tasks and categorical tasks, respectively. Obviously, we have that  $\mathcal{T} = \mathcal{T}_{\text{con}} \cup \mathcal{T}_{\text{cat}}$ .

Without loss of generality, we assume that, for each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , the ground truth  $x_j^*$  and any worker  $w_i$ ’s data  $x_{i,j}$  are normalized values within the range  $[0, 1]$ . Furthermore, we assume that all categorical tasks in  $\mathcal{T}_{\text{cat}}$  are binary classification tasks with ground truths  $x_j^*$ ’s taking values from the set  $\{+1, -1\}$ , which collect binary labels, either  $+1$  or  $-1$ , from participating workers.

### 5.2.3 Reliability Level Model

Before task  $\tau_j$  is executed by worker  $w_i$ , her data about this task can be regarded as a random variable  $X_{i,j}$ . Then, we define a worker’s *reliability level* for a continuous and categorical task, respectively, in Definition 27 and 28.

**Definition 27** (Reliability Level for Continuous Task). *Worker  $w_i$ 's reliability level  $\theta_{i,j}$  for a continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  is defined as the expected absolute difference between her data and the ground truth, i.e.,*

$$\theta_{i,j} = \mathbb{E}[|X_{i,j} - x_j^*|] \in [0, 1], \quad (5.2)$$

where the expectation is taken over the randomness of  $X_{i,j}$ .

**Definition 28** (Reliability Level for Categorical Task). *Worker  $w_i$ 's reliability level  $\theta_{i,j}$  for a categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$  is defined as the probability that she provides a correct label about this task, i.e.,*

$$\theta_{i,j} = \Pr[X_{i,j} = x_j^*] \in [0, 1]. \quad (5.3)$$

We use  $\boldsymbol{\theta} = [\theta_{i,j}] \in [0, 1]^{N \times K}$  to denote the reliability level matrix of all workers. We assume that the reliability level matrix  $\boldsymbol{\theta}$  is *a priori* known to the platform. In practice, the platform can keep a historical record of  $\boldsymbol{\theta}$ , which can be obtained by many methods. For example, since a worker's reliability levels for similar tasks typically tend to be similar, the platform could assign some tasks with known ground truths to workers and utilize workers' sensory data about these tasks to estimate their reliability levels for similar tasks as in [87]. In scenarios where ground truths are not available,  $\boldsymbol{\theta}$  can still be effectively estimated utilizing workers' previously submitted sensory data about similar tasks by algorithms proposed in [68–70, 78, 96, 101] or inferred from some of workers' characteristics (e.g., a worker's reputation and experience for similar tasks, the price of a worker's sensors) using the methods in [63].

#### 5.2.4 Auction Model

In this chapter, as in most prior work, we assume that workers are *selfish* and *strategic* that aim to maximize their own utilities. We use the term *bundle* to refer to any subset of the overall task set  $\mathcal{T}$  in the rest of this chapter. Since every worker bids on one bundle of tasks in the INCEPTION framework, we model the incentive mechanism as a single-minded reverse combinatorial auction. However, different from the traditional combinatorial auction [13, 15], we study the scenario where workers explicitly consider privacy leakage as one of the sources for their costs. Therefore, we propose the *single-minded reverse combinatorial auction with privacy cost (pSRC auction)*, formally

defined in Definition 29, as the incentive mechanism.

**Definition 29** (pSRC Auction). *In a single-minded reverse combinatorial auction with privacy cost (pSRC auction), each worker  $w_i$  has only one interested bundle  $\Gamma_i^*$ . Her cost of executing the bundle of tasks, namely sensing cost, is denoted as  $c_i^s$  (unknown to the platform). Additionally, she has a cost for privacy leakage, namely privacy cost, denoted as  $C_i^p(\epsilon)$ , if  $\epsilon$ -differential privacy is guaranteed. Hence, worker  $w_i$ 's cost function is defined as in Equation (5.4).*

$$C_i(\Gamma, \epsilon) = \begin{cases} c_i^s + C_i^p(\epsilon), & \text{if } \Gamma \subseteq \Gamma_i^* \\ +\infty, & \text{otherwise} \end{cases}. \quad (5.4)$$

For the tasks that do not belong to worker  $w_i$ 's interested bundle  $\Gamma_i^*$ , either she is not able to execute them or executing these tasks incurs a large cost. Therefore, we assign a  $+\infty$  cost to these tasks in Equation (5.4).

A major difference between the cost function defined in Equation (5.4) and those in prior work [20, 22, 27, 28, 31–33, 35, 37, 43–45, 50, 51, 54, 57, 58, 60, 62, 66, 67, 71–75, 82, 86, 88, 89, 91, 92, 95, 97, 98, 100, 106, 107, 110–113, 115–124] is that the privacy cost  $C_i^p(\epsilon)$  is explicitly integrated into it. Such integration is reasonable and necessary. In an MCS system where the platform utilizes a worker's private and sensitive data in a way that incurs privacy leakage, the worker will not be effectively incentivized to participate unless both her sensing and privacy cost are compensated. For any worker  $w_i$  the privacy cost  $C_i^p(\epsilon)$  is positively correlated with the privacy budget  $\epsilon$ , because  $\epsilon$  in fact captures the amount of privacy leakage of the MCS system. Therefore, we adopt the natural linear model for privacy cost as in [34, 39] where  $C_i^p(\epsilon) = c_i^p \epsilon$  with  $c_i^p$  representing worker  $w_i$ 's cost for unit privacy leakage. Similar to  $c_i^s$ ,  $c_i^p$  is also unknown to the platform. Next, we define a worker's utility in Definition 30.

**Definition 30** (Worker's Utility). *Any worker  $w_i$ 's utility  $u_i$  is defined as*

$$u_i = \begin{cases} p_i - c_i^s - c_i^p \epsilon, & \text{if } w_i \in \mathcal{S} \\ 0, & \text{otherwise} \end{cases}. \quad (5.5)$$

Apart from workers' utilities, we are also interested in the platform's total payment defined in

Definition 31.

**Definition 31** (Platform’s Total Payment). *Given the payment profile  $\mathbf{p}$  and the winner set  $\mathcal{S}$ , the platform’s total payment is  $P = \sum_{i:w_i \in \mathcal{S}} p_i$ .*

### 5.2.5 Design Objective

In this chapter, we aim to ensure that INCEPTION bears the following desirable properties.

Since workers are strategic in our model, it is possible that any worker  $w_i$  submits a bid  $(\Gamma_i, b_i^s, b_i^p)$  that deviates from the true value  $(\Gamma_i^*, c_i^s, c_i^p)$ . However, one of our objectives is to design a *truthful* incentive mechanism defined in Definition 32.

**Definition 32** (Truthfulness). *A pSRC auction is truthful if and only if bidding the true value  $(\Gamma_i^*, c_i^s, c_i^p)$  is the dominant strategy for each worker  $w_i$ , i.e., bidding  $(\Gamma_i^*, c_i^s, c_i^p)$  maximizes each worker  $w_i$ ’s utility for all possible values of other workers’ bids and the privacy budget  $\epsilon$ .*

By Definition 32, we aim to ensure the truthful bidding of the interested bundle  $\Gamma_i^*$ , the sensing cost  $c_i^s$ , and the cost for unit privacy leakage  $c_i^p$  for every worker  $w_i$ . Apart from truthfulness, another desirable and necessary property is *individual rationality* defined in Definition 33.

**Definition 33** (Individual Rationality). *A pSRC auction is individual rational if and only if no worker receives negative utility, i.e., we have  $u_i \geq 0$  for each worker  $w_i$ .*

Individual rationality in our pSRC auction means that a worker’s sensing and privacy cost are both compensated, which is crucial to effectively incentivize worker participation. As mentioned in Section 5.2.1, we aim to design an MCS system that ensures  $\epsilon$ -*differential privacy*. However, the perturbation added to the aggregated results inevitably impairs their accuracy. Next, we formally define the concept of  $(\alpha, \beta)$ -accuracy for continuous tasks in Definition 34.

**Definition 34** ( $(\alpha, \beta)$ -Accuracy). *For two random variables  $Y_1$  and  $Y_2$  within the range  $[0, 1]$ ,  $Y_1$  is  $(\alpha, \beta)$ -accurate to  $Y_2$ , if and only if  $\Pr[|Y_1 - Y_2| \geq \alpha] \leq \beta$ , where  $\alpha, \beta \in (0, 1)$ . Note that  $Y_2$  could also be a constant.*

We use  $\widehat{X}_j$  to denote the random variable corresponding to  $\widehat{x}_j$  (i.e., the perturbed result for task  $\tau_j$ ). Facing the trade-off between privacy and accuracy, we need to carefully control the amount of

noises added to the aggregated results and ensure that, for each continuous task  $\tau_j$ ,  $\widehat{X}_j$  is  $(\alpha, \beta)$ -accurate to the ground truth  $x_j^*$  with sufficiently small  $\alpha$  and  $\beta$  within  $(0, 1)$ . That is, we aim to ensure that the perturbed results of all continuous tasks are fairly close to the ground truths with high probability. For categorical tasks, we adopt the notion of  $\gamma$ -accuracy, which is formally defined in Definition 35.

**Definition 35** ( $\gamma$ -Accuracy). *For two random variables  $Z_1$  and  $Z_2$  that take values from the set  $\{+1, -1\}$ ,  $Z_1$  is  $\gamma$ -accurate to  $Z_2$ , if and only if  $\Pr[Z_1 \neq Z_2] \leq \gamma$ , where  $\gamma \in (0, 1)$ . Note that  $Z_2$  could also be a constant.*

For each categorical task  $\tau_j$ , we aim to ensure that the perturbed result  $\widehat{X}_j$  is  $\gamma$ -accurate to the ground truth  $x_j^*$  with a sufficiently small  $\gamma \in (0, 1)$ , which means that the perturbed results of all categorical tasks are equal to the ground truths with high probability.

In short, our objective is to design a *differentially private* MCS system that provides satisfactory *accuracy guarantee* for the final perturbed results, and incentivizes worker participation in a *truthful* and *individual rational* manner.

## 5.3 Design Details

In this section, we provide our design details for the incentive, data aggregation, and data perturbation mechanism.

### 5.3.1 Data Aggregation Mechanism

In this section, we introduce the design details of INCEPTION’s data aggregation mechanism, as well as the corresponding analyses.

#### 5.3.1.1 Proposed Mechanism

Although the data aggregation mechanism comes after the incentive mechanism in the workflow of INCEPTION, we introduce it first, as it affects the design of the incentive mechanism.

To guarantee that the perturbed results have satisfactory accuracy, the original aggregated results before perturbation need to be accurate enough in the first place. Therefore, we reasonably

assume that the platform uses a *weighted aggregation* method to calculate the aggregated result  $x_j$  for each task  $\tau_j$  based on workers' data. That is, given the winner set  $\mathcal{S}$  determined by the incentive mechanism, the aggregated result  $x_j$  of each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  is calculated as

$$x_j = \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} x_{i,j}, \quad (5.6)$$

where  $\lambda_{i,j} > 0$  is the weight of worker  $w_i$  on this task with  $\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} = 1$  for every continuous task  $\tau_j$ . Similarly, for each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , we calculate the aggregated result  $x_j$  as

$$x_j = \text{sign} \left( \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} x_{i,j} \right), \quad (5.7)$$

where, similar to continuous tasks,  $\lambda_{i,j} > 0$  is worker  $w_i$ 's weight on this task  $\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} = 1$ . Furthermore, the function  $\text{sign}(z)$  takes the value  $+1$ , when  $z \geq 0$ , and  $-1$  otherwise.

The motivation for utilizing weighted aggregation is to capture the effect of workers' diverse reliability levels on the calculation of the aggregated results. Intuitively, we should assign higher weights to workers whose sensory data are more likely to be close to the ground truths, which makes the aggregated results closer to the data provided by more reliable workers. In fact, many state-of-the-art data aggregation methods [69,78] utilize such weighted aggregation to calculate the aggregated results. Since the accuracy of the aggregated results highly depends on how exactly the weight  $\lambda_{i,j}$ 's are chosen, we propose the following data aggregation mechanism in Algorithm 8.

---

**Algorithm 8:** Data Aggregation Mechanism

---

**Input:**  $\alpha, \theta, \mathbf{b}, \mathbf{x}, \mathcal{S}, \mathcal{T}, \mathcal{T}_{\text{con}}$ ;  
**Output:**  $(x_1, \dots, x_K)$ ;  
**1** **foreach**  $j$  *s.t.*  $\tau_j \in \mathcal{T}$  **do**  
**2**     **if**  $\tau_j \in \mathcal{T}_{\text{con}}$  **then**  
       // Calculate the aggregated result of a continuous task  
**3**      $x_j \leftarrow \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \frac{(\alpha_j - \theta_{i,j}) x_{i,j}}{\sum_{k:w_k \in \mathcal{S}, \tau_j \in \Gamma_k} (\alpha_j - \theta_{k,j})}$ ;  
**4**     **else**  
       // Calculate the aggregated result of a categorical task  
**5**      $x_j \leftarrow \text{sign} \left( \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \frac{(2\theta_{i,j} - 1) x_{i,j}}{\sum_{k:w_k \in \mathcal{S}, \tau_j \in \Gamma_k} (2\theta_{k,j} - 1)} \right)$ ;  
**6** **return**  $(x_1, \dots, x_K)$ ;

---

Besides the reliability level matrix  $\boldsymbol{\theta}$ , the bid profile  $\mathbf{b}$ , workers' data  $\mathbf{x}$ , the winner set  $\mathcal{S}$ , as well as the task and continuous task set,  $\mathcal{T}$  and  $\mathcal{T}_{\text{con}}$ , Algorithm 8 also takes as input a vector of positive real numbers  $\boldsymbol{\alpha}$ , where each element  $\alpha_j$  corresponds to one continuous task  $\tau_j$ . These  $\alpha_j$ 's are parameters chosen by the platform, such that  $\max_{i:\tau_j \in \Gamma_i} \theta_{i,j} < \alpha_j < 0.5$ . Note that, for a continuous task  $\tau_j$ , large  $\theta_{i,j}$  indicates low reliability, and any worker  $w_i$  with  $\theta_{i,j} \geq 0.5$  will not be selected by the incentive mechanism to execute this task. The aggregated result  $x_j$  of every continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  is calculated (line 3) using Equation (5.6) with the weight

$$\lambda_{i,j} = \frac{\alpha_j - \theta_{i,j}}{\sum_{k:w_k \in \mathcal{S}, \tau_j \in \Gamma_k} (\alpha_j - \theta_{k,j})}, \quad \forall w_i \in \mathcal{S}, \tau_j \in \Gamma_i. \quad (5.8)$$

By Equation (5.8), worker  $w_i$ 's weight for a continuous task  $\tau_j$ , namely  $\lambda_{i,j}$ , increases with the decrease of  $\theta_{i,j}$ . Such a design choice conforms to our intuition that the less the expected deviation of worker  $w_i$ 's data compared to the ground truth  $x_j^*$ , the more  $x_{i,j}$  should be counted in the calculation of the aggregated result  $x_j$ .

For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , we calculate its aggregated result  $x_j$  (line 5) using Equation (5.7) with the weight,

$$\lambda_{i,j} = \frac{2\theta_{i,j} - 1}{\sum_{k:w_k \in \mathcal{S}, \tau_j \in \Gamma_k} (2\theta_{k,j} - 1)}, \quad \forall w_i \in \mathcal{S}, \tau_j \in \Gamma_i. \quad (5.9)$$

Note that large  $\theta_{i,j}$  for a categorical task implies high reliability, and the incentive mechanism will not select any worker  $w_i$  with  $\theta_{i,j} \leq 0.5$  to execute this task. Following a similar philosophy as calculating the aggregated result of a continuous task, the data from workers with higher reliability are counted more in the calculation of a categorical task's aggregated result, as well. Formal analysis about the data aggregation mechanism is provided in Section 5.3.1.2.

### 5.3.1.2 Analysis

In this section, we firstly analyze Algorithm 8's guarantee of aggregation accuracy for continuous tasks. In the following Lemma 7, we establish an upper bound for the accuracy of the aggregated result  $x_j$  of each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  compared to its ground truth  $x_j^*$ . In the rest of our analyses, we use  $X_j$  to denote the random variable representing any task  $\tau_j$ 's aggregated result  $x_j$ .

**Lemma 7.** For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , given the winner set  $\mathcal{S}$ , the reliability level matrix  $\boldsymbol{\theta}$ , the vector of platform-chosen parameter  $\boldsymbol{\alpha}$ , as well as workers' weights  $\lambda_{i,j}$ 's on this task, we have that

$$\Pr[|X_j - x_j^*| \geq \alpha_j] \leq \exp\left(-\frac{2\left(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}(\alpha_j - \theta_{i,j})\right)^2}{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right) \quad (5.10)$$

by aggregating workers' data according to Equation (5.6).

*Proof.* From Equation (5.6), for each continuous task  $\tau_j$ , we have that

$$\begin{aligned} |X_j - x_j^*| &= \left| \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} X_{i,j} - x_j^* \right| = \left| \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} (X_{i,j} - x_j^*) \right| \\ &\leq \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \left| \lambda_{i,j} (X_{i,j} - x_j^*) \right|. \end{aligned}$$

We define a random variable  $L_j$  for every continuous task  $\tau_j$  as  $L_j = \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} |\lambda_{i,j} (X_{i,j} - x_j^*)|$ , which is the sum of random variables  $L_{i,j}$ 's with  $L_{i,j} = |\lambda_{i,j} (X_{i,j} - x_j^*)| \in [0, \lambda_{i,j}]$ . Thus,

$$\mathbb{E}[L_j] = \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} \mathbb{E}[|X_{i,j} - x_j^*|] = \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} \theta_{i,j}.$$

Therefore, from the Hoeffding bound, we have

$$\begin{aligned} \Pr[|X_j - x_j^*| \geq \alpha_j] &\leq \Pr\left[\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} |\lambda_{i,j} (X_{i,j} - x_j^*)| \geq \alpha_j\right] \\ &= \Pr[Y_j \geq \alpha_j] = \Pr[Y_j - \mathbb{E}[Y_j] > \alpha_j - \mathbb{E}[Y_j]] \\ &\stackrel{\text{(Hoeffding bound)}}{\leq} \exp\left(-\frac{2(\alpha_j - \mathbb{E}[Y_j])^2}{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right) \\ &= \exp\left(-\frac{2\left(\alpha_j - \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} \theta_{i,j}\right)^2}{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right) \\ &= \exp\left(-\frac{2\left(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}(\alpha_j - \theta_{i,j})\right)^2}{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}\right), \end{aligned}$$

which exactly proves this lemma.  $\square$

Clearly, Lemma 7 gives us an upper bound for the probability  $\Pr[|X_j - x_j^*| \geq \alpha_j]$  for each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ . Then, in the following Theorem 24, we will prove that this upper bound

is minimized by our proposed Algorithm 8.

**Theorem 24.** *For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , the data aggregation mechanism proposed in Algorithm 8 minimizes the upper bound of the probability  $\Pr[|X_j - x_j^*| \geq \alpha_j]$  established in Lemma 7, and ensures that*

$$\Pr[|X_j - x_j^*| \geq \alpha_j] \leq \exp\left(-2 \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2\right). \quad (5.11)$$

*Proof.* For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , we denote  $\boldsymbol{\lambda}_j = [\lambda_{i,j}]$  as the vector that contains every  $\lambda_{i,j}$  such that  $w_i \in \mathcal{S}$  and  $\tau_j \in \Gamma_i$ . Therefore, minimizing the upper bound of  $\Pr[|X_j - x_j^*| \geq \alpha_j]$  established in Lemma 7 is equivalent to maximizing the function  $\varphi(\boldsymbol{\lambda}_j)$  defined as

$$\varphi(\boldsymbol{\lambda}_j) = \frac{\left(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} (\alpha_j - \theta_{i,j})\right)^2}{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2}. \quad (5.12)$$

From the Cauchy-Schwarz inequality, we have that

$$\begin{aligned} \varphi(\boldsymbol{\lambda}_j) &\leq \frac{\left(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2\right) \left(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2\right)}{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2} \\ &= \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2 \end{aligned}$$

and  $\varphi(\boldsymbol{\lambda}_j) = \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2$  is achieved when  $\lambda_{i,j} \propto \alpha_j - \theta_{i,j}$ .

Using the fact that  $\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} = 1$ , we have

$$\lambda_{i,j} = \frac{\alpha_j - \theta_{i,j}}{\sum_{k:w_k \in \mathcal{S}, \tau_j \in \Gamma_k} (\alpha_j - \theta_{k,j})}. \quad (5.13)$$

Therefore, by substituting the expression of  $\lambda_{i,j}$  given in Equation (5.13) into Equation (5.12), we have that

$$\Pr[|X_j - x_j^*| \geq \alpha_j] \leq \exp\left(-2 \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2\right),$$

which is exactly the Equation (5.11) in Theorem 24.  $\square$

By Theorem 24, for each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , the data aggregation mechanism proposed in

Algorithm 8 upper bounds the probability of  $\Pr[|X_j - x_j^*| \geq \alpha_j]$  by  $\exp(-2 \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2)$  which is in fact the minimum value of the upper bound established in Lemma 7 for this probability. Then, we introduce Corollary 2 which is directly utilized in the design of the incentive mechanism in Section 5.3.2.

**Corollary 2.** *For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , if*

$$\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2 \geq \frac{1}{2} \ln \left( \frac{1}{\beta_j} \right), \quad (5.14)$$

*then the data aggregation mechanism proposed in Algorithm 8 ensures that  $\Pr[|X_j - x_j^*| \geq \alpha_j] \leq \beta_j$ , where  $\beta_j \in (0, 1)$  is a parameter chosen by the platform for this task. We use  $\boldsymbol{\beta}$  to denote the vector, where each element  $\beta_j$  corresponds to one continuous task  $\tau_j$ .*

*Proof.* Corollary 2 directly follows from Theorem 24. If we let the upper bound of  $\Pr[|X_j - x_j^*| \geq \alpha_j]$  guaranteed by Algorithm 8 to be no greater than  $\beta_j \in (0, 1)$ , we have

$$\exp \left( -2 \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2 \right) \leq \beta_j,$$

which is equivalent to exactly

$$\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2 \geq \frac{1}{2} \ln \left( \frac{1}{\beta_j} \right). \quad (5.15)$$

Therefore, together with Theorem 24, we know that Inequality (5.15) implies  $\Pr[|X_j - x_j^*| \geq \alpha_j] \leq \beta_j$ .  $\square$

Corollary 2 states that  $(\alpha_j, \beta_j)$ -accuracy is guaranteed for the aggregated result of every continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  compared to the corresponding ground truth  $x_j^*$ , if the condition specified by Inequality (5.14) is satisfied by the set of selected winners  $\mathcal{S}$  in the incentive mechanism proposed in Section 5.3.2.

Next, we introduce the results on Algorithm 8's aggregation accuracy for categorical tasks in the following Lemma 8, Theorem 25, and Corollary 3, which are adapted from the Theorem 1 and Corollary 1 in [54].

**Lemma 8.** For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , given the winner set  $\mathcal{S}$ , the reliability level matrix  $\theta$ , as well as workers' weights  $\lambda_{i,j}$ 's on this task, we have that

$$\Pr[X_j \neq x_j^*] \leq \exp \left( - \frac{(\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j} (2\theta_{i,j} - 1))^2}{2 \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} \lambda_{i,j}^2} \right) \quad (5.16)$$

by aggregating workers' data according to Equation (5.7).

**Theorem 25.** For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , the data aggregation mechanism proposed in Algorithm 8 minimizes the upper bound of the probability  $\Pr[X_j \neq x_j^*]$  established in Lemma 8, and ensures that

$$\Pr[X_j \neq x_j^*] \leq \exp \left( - \frac{\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2}{2} \right). \quad (5.17)$$

**Corollary 3.** For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , if

$$\sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 \geq 2 \ln \left( \frac{1}{\gamma_j} \right), \quad (5.18)$$

then the data aggregation mechanism proposed in Algorithm 8 ensures that  $\Pr[X_j \neq x_j^*] \leq \gamma_j$ , where  $\gamma_j \in (0, 1)$  is a parameter chosen by the platform for this task. We use  $\boldsymbol{\gamma}$  to denote the vector, where each element  $\gamma_j$  corresponds to one categorical task  $\tau_j$ .

The proofs of Lemma 8, Theorem 25, and Corollary 3 are omitted in this chapter, because they can be adapted from those of the Theorem 1 and Corollary 1 in [54] with minor changes. Clearly, they are counterparts of Lemma 7, Theorem 24, and Corollary 2 for categorical tasks, and collectively ensure that  $\gamma_j$ -accuracy is guaranteed for the aggregated result of each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , as long as Inequality (5.18) is satisfied by the winners selected by the incentive mechanism. Next, in Section 5.3.2, we introduce the design of INCEPTION's incentive mechanism, which is based on the data aggregation mechanism proposed in Algorithm 8.

### 5.3.2 Incentive Mechanism

In this section, we introduce the mathematical formulation, design details and the analysis of the proposed incentive mechanism.

### 5.3.2.1 Mathematical Formulation

As mentioned in Section 5.2.4, our incentive mechanism is based on the pSRC auction defined in Definition 29. In this chapter, we aim to design a pSRC auction that *minimizes the platform's total payment* with *satisfactory data aggregation accuracy*. Such a design choice exactly captures the objective of most MCS systems, that is to collect *high quality* data from the crowd with *minimum total expense*. The formal mathematical formulation is given in the following *pSRC auction total payment minimization (pSRC-TPM)* problem.

**pSRC-TPM Problem:**

$$\min \sum_{i:w_i \in \mathcal{N}} p_i y_i \quad (5.19)$$

$$\text{s.t.} \quad \sum_{i:w_i \in \mathcal{N}, \tau_j \in \Gamma_i} (\alpha_j - \theta_{i,j})^2 y_i \geq \frac{1}{2} \ln \left( \frac{1}{\beta_j} \right), \quad \forall \tau_j \in \mathcal{T}_{\text{con}} \quad (5.20)$$

$$\sum_{i:w_i \in \mathcal{N}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)^2 y_i \geq 2 \ln \left( \frac{1}{\gamma_j} \right), \quad \forall \tau_j \in \mathcal{T}_{\text{cat}} \quad (5.21)$$

$$y_i \in \{0, 1\}, \quad p_i \in [0, +\infty), \quad \forall w_i \in \mathcal{N} \quad (5.22)$$

**Constants.** The pSRC-TPM problem takes as inputs the worker set  $\mathcal{N}$ , the continuous and categorical task set  $\mathcal{T}_{\text{con}}$  and  $\mathcal{T}_{\text{cat}}$ , workers' bid profile  $\mathbf{b}$ , the reliability level matrix  $\boldsymbol{\theta}$ , and the  $\boldsymbol{\beta}$ ,  $\boldsymbol{\alpha}$ , and  $\boldsymbol{\gamma}$  vector.

**Variables.** The pSRC-TPM problem has a vector of  $N$  binary variables, denoted as  $\mathbf{y} = (y_1, \dots, y_N)$ . The variable  $y_i = 1$  indicates that the worker  $w_i$  is selected as a winner (i.e.,  $w_i \in \mathcal{S}$ ); otherwise  $w_i \notin \mathcal{S}$ . The second vector of variables is the payment profile  $\mathbf{p} = (p_1, \dots, p_N)$ , where every element takes a non-negative real value.

**Objective function.** The objective function given by  $\sum_{i:w_i \in \mathcal{N}} p_i y_i = \sum_{i:w_i \in \mathcal{S}} p_i$  is exactly the total payment made by the platform to all winners.

**Constraints.** Constraint (5.20) is equivalent to Inequality (5.14) given in Corollary 2, which specifies the condition that the selected winners should satisfy. By Corollary 2, any feasible solution  $\mathbf{y}$  to the pSRC-TPM problem gives a winner set  $\mathcal{S}$  which ensures that the aggregated result of every continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  is  $(\alpha_j, \beta_j)$ -accurate to the ground truth  $x_j^*$ . Similarly, Constraint (5.21)

is equivalent to Inequality (5.7), which ensures that  $\gamma_j$ -accuracy for each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$  is satisfied by the winner set  $\mathcal{S}$  given by any feasible solution  $\mathbf{y}$  to the pSRC-TPM problem. To simplify presentation, we introduce the following extra notations. For each worker  $w_i \in \mathcal{N}$  and task  $\tau_j \in \mathcal{T}$ , we define

$$q_{i,j} = \begin{cases} (\alpha_j - \theta_{i,j})^2, & \text{if } \tau_j \in \mathcal{T}_{\text{con}} \\ (2\theta_{i,j} - 1)^2, & \text{if } \tau_j \in \mathcal{T}_{\text{cat}} \end{cases}, \quad (5.23)$$

and for each task  $\tau_j \in \mathcal{T}$ , we define

$$Q_j = \begin{cases} \frac{1}{2} \ln \left( \frac{1}{\beta_j} \right), & \text{if } \tau_j \in \mathcal{T}_{\text{con}} \\ 2 \ln \left( \frac{1}{\gamma_j} \right), & \text{if } \tau_j \in \mathcal{T}_{\text{cat}} \end{cases}. \quad (5.24)$$

Furthermore, we define  $\mathbf{q} = [q_{i,j}] \in [0, +\infty)^{N \times K}$  and  $\mathbf{Q} = [Q_j] \in [0, +\infty)^{K \times 1}$ . Therefore, Constraint (5.20) and (5.21) can be simplified and merged into the following Constraint (5.25).

$$\sum_{i:w_i \in \mathcal{N}, \tau_j \in \Gamma_i} q_{i,j} y_i \geq Q_j, \quad \forall \tau_j \in \mathcal{T}. \quad (5.25)$$

Besides Constraint (5.20) and (5.21), any feasible solution to the pSRC-TPM problem should also satisfy two other inherent constraints, namely truthfulness and individual rationality, which means that the pSRC auction corresponding to the solution is truthful and individual rational. Because of the difficulty in mathematically formulating the two constraints, we take them into consideration without explicitly formulating them, in the pSRC-TPM problem.

In Theorem 26, we prove the NP-hardness of the pSRC-TPM problem.

**Theorem 26.** *The pSRC-TPM problem is NP-hard.*

*Proof.* We consider a special case of the pSRC-TPM problem with a constant payment profile  $\mathbf{p}$  and the truthfulness and individual rationality constraints relaxed. With constant  $p_i$ 's, it becomes a binary linear program (BLP). We prove the NP-hardness of the BLP by a polynomial-time reduction from the minimum weight set cover (MWSC) problem.

The reduction starts from an instance of the NP-complete MWSC problem with a universe  $\mathcal{T} = \{\tau_1, \dots, \tau_K\}$  and a set of subsets of  $\mathcal{T}$  defined as  $\mathcal{R} = \{\Gamma_1, \dots, \Gamma_N\}$ . Each set  $\Gamma_i \in \mathcal{R}$  has

---

**Algorithm 9:** pSRC Auction Winner Determination

---

**Input:**  $\epsilon, \mathbf{b}, \mathbf{q}, \mathbf{Q}, \mathcal{N}, \mathcal{T}$ ;  
**Output:**  $\mathcal{S}$ ;  
// Initialization  
1  $\mathcal{S} \leftarrow \emptyset, \mathbf{Q}' \leftarrow \mathbf{Q}$ ;  
// Calculate the winner set  $\mathcal{S}$   
2 **while**  $\sum_{j:\tau_j \in \mathcal{T}} Q'_j \neq 0$  **do**  
    // Find the worker with the minimum bidding price effectiveness  
3  $l = \arg \min_{i:w_i \in \mathcal{N}} \frac{b_i^s + b_i^p \epsilon}{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}}$ ;  
4  $\mathcal{S} \leftarrow \mathcal{S} \cup \{w_l\}$ ;  
5  $\mathcal{N} \leftarrow \mathcal{N} \setminus \{w_l\}$ ;  
    // Update the  $\mathbf{Q}'$  vector  
6 **foreach**  $j$  *s.t.*  $\tau_j \in \mathcal{T}$  **do**  
7      $Q'_j \leftarrow Q'_j - \min\{Q'_j, q_{l,j}\}$ ;  
8 **return**  $\mathcal{S}$ ;

---

a non-negative weight  $p_i$ . The objective of the MWSC problem is to find the subset of  $\mathcal{R}$  with the minimum total weight whose union equals to  $\mathcal{T}$ . We transform  $\Gamma_i$  to  $\Gamma'_i$  where each element  $\tau_j \in \Gamma_i$  has  $a_{i,j} \in \mathbb{Z}^+$  copies and require each  $\tau_j$  to be covered for exactly  $A_j \in \mathbb{Z}^+$  times. By now, an instance of the BLP with  $\mathbf{q} = [a_{i,j}] \in (\mathbb{Z}^+)^{N \times K}$ ,  $\mathbf{Q} = [A_j] \in (\mathbb{Z}^+)^{K \times 1}$ , and payment profile  $\mathbf{p}$  has been constructed. Actually, a richer family of problems can be represented by the BLP because elements in  $\mathbf{q}$  and  $\mathbf{Q}$  can be any positive real numbers besides positive integers. Hence, every instance of the MWSC problem is polynomial-time reducible to the BLP, which proves its NP-hardness. Furthermore, because the BLP is only a special case of the pSRC-TPM problem, the pSRC-TPM problem is also NP-hard.  $\square$

### 5.3.2.2 Proposed Mechanism

Because of the NP-hardness of the pSRC-TPM problem proved in Theorem 26, directly solving it to obtain the winner set  $\mathcal{S}$  and the payment profile  $\mathbf{p}$  is computationally intractable when the cardinality of  $\mathcal{N}$  and  $\mathcal{T}$  become large. Therefore, we propose our own winner determination and pricing algorithm for the pSRC auction in Algorithm 9 and 10, respectively. The proposed algorithms are *computationally efficient* and approximately minimize the platform's total payment with a *guaranteed approximation ratio*.

The inputs of the winner determination algorithm given in Algorithm 9 include the privacy

---

**Algorithm 10:** pSRC Auction Pricing
 

---

**Input:**  $\epsilon, \mathbf{b}, \mathbf{q}, \mathbf{Q}, \mathcal{N}, \mathcal{T}, \mathcal{S}$ ;  
**Output:**  $\mathbf{p}$ ;  
 // Initialization  
 1  $\mathbf{p} \leftarrow (0, \dots, 0)$ ;  
 2 **foreach**  $i$  *s.t.*  $w_i \in \mathcal{S}$  **do**  
 3     run Algorithm 9 on  $\mathcal{N} \setminus \{w_i\}$  until  $\sum_{j:\tau_j \in \Gamma_i} Q'_j = 0$ ;  
 4      $\mathcal{S}' \leftarrow$  the winner set when step 3 stops;  
    // Calculate payment  
 5     **foreach**  $k$  *s.t.*  $w_k \in \mathcal{S}'$  **do**  
 6          $\mathbf{Q}' \leftarrow$  tasks'  $\mathbf{Q}'$  vector when  $w_k$  is selected;  
 7          $p_i \leftarrow \max \left\{ p_i, (b_k^s + b_k^p \epsilon) \cdot \frac{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}}{\sum_{j:\tau_j \in \Gamma_k} \min\{Q'_j, q_{k,j}\}} \right\}$ ;  
 8 **return**  $\mathbf{p}$ ;  


---

budget  $\epsilon$ , bid profile  $\mathbf{b}$ ,  $\mathbf{q}$  matrix,  $\mathbf{Q}$  vector, worker set  $\mathcal{N}$ , and task set  $\mathcal{T}$ . Firstly, it initializes the winner set  $\mathcal{S}$  as  $\emptyset$  and the residual vector of  $\mathbf{Q}$ , namely  $\mathbf{Q}'$ , as  $\mathbf{Q}$  (line 1). Then, the main loop (line 2-7) calculates the winner set  $\mathcal{S}$ . It is executed until the winner set  $\mathcal{S}$  makes the pSRC-TPM problem feasible (line 2). We define worker  $w_i$ 's *virtual bidding price* as  $b_i^v = b_i^s + b_i^p \epsilon$ . In each iteration, Algorithm 9 finds the worker  $w_l$  with the *minimum bidding price effectiveness* (line 3) defined as the ratio between her virtual bidding price and her contribution to the improvement of the feasibility of Constraint (5.20). Next,  $w_l$  is included into the winner set  $\mathcal{S}$  (line 4) and excluded from the worker set  $\mathcal{N}$  (line 5). Finally, the  $\mathbf{Q}'$  vector is updated (line 6-7).

Apart from the same inputs taken by Algorithm 9, the pricing algorithm given in Algorithm 10 also uses the winner set  $\mathcal{S}$  calculated by Algorithm 9. Firstly, it initializes the payment profile  $\mathbf{p}$  as a vector of  $N$  zeros (line 1). Then, the main loop (line 2-7) calculates the payment to each winner. For each winner  $w_i \in \mathcal{S}$ , Algorithm 9 is executed on the worker set containing all workers except  $w_i$  until the point after which  $w_i$  will never be selected as a winner (line 3). The winner set at this point is recorded as  $\mathcal{S}'$  (line 4). For each worker  $w_k \in \mathcal{S}'$ , Algorithm 10 calculates worker  $w_i$ 's maximum virtual bidding price  $b_{i,k}^v$  that makes her substitute  $w_k$  as the winner. To achieve this,  $b_{i,k}^v$  should satisfy

$$\frac{b_{i,k}^v}{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}} = \frac{b_k^s + b_k^p \epsilon}{\sum_{j:\tau_j \in \Gamma_k} \min\{Q'_j, q_{k,j}\}},$$

which is equivalent to

$$b_{i,k}^v = (b_k^s + b_k^p \epsilon) \cdot \frac{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}}{\sum_{j:\tau_j \in \Gamma_k} \min\{Q'_j, q_{k,j}\}}.$$

Then, the maximum value among these  $b_{i,k}^v$ 's is chosen as the payment  $p_i$  to worker  $w_i$  (line 7).

### 5.3.2.3 Analysis

Firstly, we analyze the truthfulness of the proposed pSRC auction in Theorem 27.

**Theorem 27.** *The proposed pSRC auction is truthful.*

*Proof.* Firstly, we fix the privacy budget  $\epsilon$  and assume a worker  $w_i$  wins the auction by bidding  $b_i = (\Gamma_i, b_i^s, b_i^p)$ . We show that the pSRC auction satisfies the property of *monotonicity* and *critical payment* in terms of the bidding bundle  $\Gamma_i$  and virtual bidding price  $b_i^v = b_i^s + b_i^p \epsilon$ .

- **Monotonicity.** Consider worker  $w_i$ 's bid  $\tilde{b}_i = (\tilde{\Gamma}_i, \tilde{b}_i^s, \tilde{b}_i^p)$  with  $\tilde{\Gamma}_i \supset \Gamma_i$  and  $\tilde{b}_i^v = \tilde{b}_i^s + \tilde{b}_i^p \epsilon < b_i^v$ . Algorithm 9 selects winners in an increasing order of the bidding price effectiveness. Hence,  $\tilde{b}_i$  will also make worker  $w_i$  a winner, as it increases her priority of winning compared to  $b_i$ .
- **Critical payment.** Algorithm 10 in fact pays every winner the supremum of all virtual bidding prices that can still make her a winner, namely critical payment.

As proved in [15, 50], the monotonicity and critical payment property make the pSRC auction truthful in terms of the bidding bundle and the virtual bidding price. That is worker  $w_i$  maximizes her utility by bidding  $\Gamma_i^*$  and  $(b_i^s, b_i^p)$  such that  $b_i^s + b_i^p \epsilon = c_i^s + c_i^p \epsilon$ . For a fixed  $\epsilon$ , the worker still has incentive to bid  $(b_i^s, b_i^p) \neq (c_i^s, c_i^p)$ . However, since the exact value of  $\epsilon$  is not revealed to workers in the bidding process, the only strategy that maximizes her utility under all possible values of  $\epsilon$  is to bid  $b_i^s = c_i^s$  and  $b_i^p = c_i^p$ . Therefore, the pSRC auction is truthful.  $\square$

The proposed pSRC auction ensures that truthful bidding is a dominant strategy for every worker under any possible value of  $\epsilon$ . As stated in the proof of Theorem 27, it is crucial to keep the exact value of the privacy budget  $\epsilon$  confidential to workers in the bidding process to ensure the truthfulness of a worker's bidding prices for the costs of sensing and unit privacy leakage, i.e., to achieve  $b_i^s = c_i^s$  and  $b_i^p = c_i^p$  for every worker  $w_i$ . The reason that the platform firstly announces

to workers an upper bound of  $\epsilon$  is to avoid their concerns of the possibility for excessively large privacy leakage. Next, we analyze the individual rationality of the pSRC auction.

**Theorem 28.** *The pSRC auction is individual rational.*

*Proof.* By Definition 30, losers of the auction receive zero utilities. From Theorem 27, every winner  $w_i$  bids to the platform the true value  $(c_i^s, c_i^p)$  and the payment  $p_i$  to this winner is exactly the supremum of all virtual bidding prices for her to win the auction. Therefore, it is guaranteed that  $p_i \geq c_i^s + c_i^p \epsilon$ , which is equivalent to  $u_i \geq 0$ . Hence, the proposed pSRC auction is individual rational.  $\square$

In our INCEPTION framework, the platform reveals the exact value of the privacy budget  $\epsilon$  when workers receive their payments so that they can evaluate their utilities after participating and confirm that their utilities are in fact non-negative. Next, we analyze the algorithmic properties of the pSRC auction.

**Theorem 29.** *The computational complexity of the proposed pSRC auction is  $O(N^3 + N^2K)$ .*

*Proof.* The main loop (line 2-7) of Algorithm 9 terminates in worst case after  $N$  iterations. In every iteration, it takes  $O(N)$  time to find the worker with the minimum bidding price effectiveness (line 3), and at most  $K$  other iterations are needed to update the  $\mathbf{Q}'$  vector (line 6-7). Therefore, the computational complexity of Algorithm 9 is  $O(N^2 + NK)$ .

Furthermore, the computational complexity of Algorithm 10 is  $O(N^3 + N^2K)$ , because there is one more layer of loop that executes for  $N$  iterations in worst case. In conclusion, the computational complexity of the pSRC auction is  $O(N^3 + N^2K)$ .  $\square$

Before analyzing the approximation ratio of the platform's total payment generated by the pSRC auction to the optimal total payment, we introduce Lemma 9 and 10 that are utilized in the analysis. The two lemmas are directly related to the *pSRC auction social cost minimization (pSRC-SCM) problem* defined as follows.

**pSRC-SCM Problem:**

$$\min \sum_{i:w_i \in \mathcal{N}} (c_i^s + c_i^p \epsilon) y_i \tag{5.26}$$

$$\text{s.t.} \quad \sum_{i:w_i \in \mathcal{N}, \tau_j \in \Gamma_i} q_{i,j} y_i \geq Q_j, \quad \forall \tau_j \in \mathcal{T} \quad (5.27)$$

$$y_i \in \{0, 1\}, \quad \forall w_i \in \mathcal{N} \quad (5.28)$$

The pSRC-SCM problem has the same set of inputs, constraints (including the inherent truthfulness and individual rationality constraints), and variables  $\mathbf{y} = \{y_1, \dots, y_N\}$  as the pSRC-TPM problem. Instead of the platform's total payment, it minimizes the social cost, i.e.,  $\sum_{i:w_i \in \mathcal{S}} (c_i^s + c_i^p \epsilon)$ , which is the sum of all winners' costs.

**Lemma 9.** *The optimal social cost of the pSRC-SCM problem, denoted as  $C_{\text{OPT}}$ , is a lower bound of the optimal total payment of the pSRC-TPM problem, denoted as  $P_{\text{OPT}}$ .*

*Proof.* Suppose  $(\mathbf{y}^*, \mathbf{p}^*)$  is the optimal solution to the pSRC-TPM problem. We have  $P_{\text{OPT}} = \sum_{i:w_i \in \mathcal{N}} p_i^* y_i^*$ .

Since the pSRC-TPM problem and the pSRC-SCM problem have the same set of constraints,  $(\mathbf{y}^*, \mathbf{p}^*)$  is also feasible to the pSRC-SCM problem. Furthermore, from individual rationality, we have  $p_i^* \geq (c_i^s + c_i^p \epsilon) y_i^*$  for every worker  $w_i$ . Therefore, we have

$$C_{\text{OPT}} \leq \sum_{i:w_i \in \mathcal{N}} (c_i^s + c_i^p \epsilon) y_i^* \leq \sum_{i:w_i \in \mathcal{N}} p_i^* y_i^* = P_{\text{OPT}},$$

which means that  $C_{\text{OPT}}$  is a lower bound of  $P_{\text{OPT}}$ .  $\square$

Then, we introduce Lemma 10 which is borrowed from [50] (Theorem 5 in [50]) with some minor adaptations. Similar to [50], we introduce the following notations including  $\eta = \max_{i,j:w_i \in \mathcal{N}, \tau_j \in \mathcal{T}} (c_i^s + c_i^p \epsilon) q_{i,j} |\Gamma_i|$  and  $m = \frac{1}{\Delta q} \sum_{j:\tau_j \in \mathcal{T}} Q_j$  where  $\Delta q$  is the unit measure of elements in  $\mathbf{q}$  and  $\mathbf{Q}$ .

**Lemma 10.** *The social cost generated by Algorithm 9 satisfies  $2\eta H_m$ -approximation to the optimal social cost, i.e.,*

$$\sum_{i:w_i \in \mathcal{S}} (c_i^s + c_i^p \epsilon) \leq 2\eta H_m C_{\text{OPT}},$$

where  $H_m = 1 + \frac{1}{2} + \dots + \frac{1}{m}$ .

The proof to Lemma 10, which can be found in [50] is omitted in this chapter. We define

$\nu = \max_{i,k:w_i,w_k \in \mathcal{N}} \frac{c_i^s + c_i^p \epsilon}{c_k^s + c_k^p \epsilon}$ ,  $\rho = \frac{1}{\Delta q} \max_{i,j:w_i \in \mathcal{N}, \tau_j \in \mathcal{T}} q_{i,j} |\Gamma_i|$ , and introduce the following Theorem 30 regarding the approximation ratio of the proposed pSRC auction in terms of the platform's total payment.

**Theorem 30.** *The platform's total payment generated by the proposed pSRC auction satisfies  $2\rho\nu\eta H_m$ -approximation to the optimal total payment, i.e.,*

$$\sum_{i:w_i \in \mathcal{S}} p_i \leq 2\rho\nu\eta H_m P_{\text{OPT}}.$$

*Proof.* Based on Algorithm 10, for every winner  $w_i$  there exists some worker  $w_{k_i}$  such that

$$p_i = (c_{k_i}^s + c_{k_i}^p \epsilon) \cdot \frac{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}}{\sum_{j:\tau_j \in \Gamma_{k_i}} \min\{Q'_j, q_{k_i,j}\}},$$

where  $Q'_j$  denotes the element corresponding to task  $\tau_j$  in the  $\mathbf{Q}'$  vector determined on line 6 of Algorithm 10 when the worker  $w_{k_i}$  is selected as a winner. Therefore, we have

$$\begin{aligned} \sum_{i:w_i \in \mathcal{S}} p_i &= \sum_{i:w_i \in \mathcal{S}} (c_{k_i}^s + c_{k_i}^p \epsilon) \cdot \frac{\sum_{j:\tau_j \in \Gamma_i} \min\{Q'_j, q_{i,j}\}}{\sum_{j:\tau_j \in \Gamma_{k_i}} \min\{Q'_j, q_{k_i,j}\}} \\ &\leq \max_{i:w_i \in \mathcal{N}} (c_i^s + c_i^p \epsilon) \cdot \left( \frac{1}{\Delta q} \sum_{i:w_i \in \mathcal{S}} \sum_{j:\tau_j \in \Gamma_i} q_{i,j} \right) \\ &\leq |\mathcal{S}| \max_{i:w_i \in \mathcal{N}} (c_i^s + c_i^p \epsilon) \cdot \left( \frac{1}{\Delta q} \max_{i,j:w_i \in \mathcal{N}, \tau_j \in \mathcal{T}} q_{i,j} |\Gamma_i| \right) \\ &= \rho |\mathcal{S}| \max_{i:w_i \in \mathcal{N}} (c_i^s + c_i^p \epsilon). \end{aligned} \tag{5.29}$$

Furthermore, the social cost satisfies that

$$\sum_{i:w_i \in \mathcal{S}} (c_i^s + c_i^p \epsilon) \geq |\mathcal{S}| \min_{i:w_i \in \mathcal{N}} (c_i^s + c_i^p \epsilon). \tag{5.30}$$

From Inequality (5.29) and (5.30), and Lemma 9 and 10, we have that

$$\begin{aligned} \sum_{i:w_i \in \mathcal{S}} p_i &\leq \rho \left( \max_{i,k:w_i,w_k \in \mathcal{N}} \frac{c_i^s + c_i^p \epsilon}{c_k^s + c_k^p \epsilon} \right) \sum_{i:w_i \in \mathcal{S}} (c_i^s + c_i^p \epsilon) \\ &= \rho \nu \sum_{i:w_i \in \mathcal{S}} (c_i^s + c_i^p \epsilon) \leq 2\rho\nu\eta H_m P_{\text{OPT}}. \end{aligned}$$

Therefore, the proposed pSRC auction satisfies  $2\rho\nu\eta H_m$ -approximation to the optimal total payment.  $\square$

Note that there is a  $\max_{i \in \mathcal{N}} |\Gamma_i|$  factor in  $\rho$  and  $\eta$ , which could be large theoretically, and in worst case equals to the number of tasks  $K$ . However, practically, as a worker  $w_i$  typically has a limited capability and interest in terms of the number of tasks she can and wants to execute,  $\max_{i \in \mathcal{N}} |\Gamma_i|$  will be far less than  $K$ , which prevents the  $2\rho\nu\eta H_m$  approximation ratio proved in Theorem 30 from growing excessively large, in practice, as  $K$  increases. Thus far, this the best approximation ratio we have found, and we leave the proof of its tightness or the derivation of a better one, as well as the calculation of a lower bound for the approximation ratio, in our future work.

### 5.3.3 Data Perturbation Mechanism

In this section, we introduce the design details of INCEPTION’s data perturbation mechanism, as well as the corresponding analyses.

#### 5.3.3.1 Proposed Mechanism

As previously mentioned, any adversary curious about workers’ data could try to infer them utilizing the aggregated results if they are published directly. One example of such an adversary could be another competing platform hosting similar sensing tasks. The portion of workers’ data inferred with reasonable accuracy could be utilized by the adversary platform to calculate the results of its own tasks. In this way, it could reduce the number of workers recruited by itself, and thus its financial expense for worker recruiting.

To enable such inference, the adversary needs the information about workers’ weights, namely  $\lambda_{i,j}$ ’s, defined in Equation (5.8). That is, it has to know  $\alpha$  and  $\theta$ , which is usually feasible for the adversary platform. For similar sensing tasks,  $\alpha$  is typically a common and standard design choice across different platforms, and workers’ reliability levels for similar tasks tend to be similar as well. Therefore,  $\theta$  can also be effectively estimated or inferred by the adversary platform using the methods mentioned in Section 5.2.3, such as utilizing workers’ sensory data about similar tasks collected during its past interactions with them as in [69,78], using some of workers’ characteristics

(e.g., reputation and experience for similar tasks) as in [63], and many others. To tackle such inference attack, we propose a novel data perturbation mechanism in Algorithm 11 by tailoring the Laplace mechanism in [29, 39] to our problem setting.

---

**Algorithm 11:** Data Perturbation Mechanism

---

**Input:**  $(x_1, \dots, x_N)$ ,  $\alpha$ ,  $\beta$ ,  $\mathcal{T}$ ,  $\mathcal{T}_{\text{con}}$ ,  $\tilde{\mathbf{x}}$ ,  $\delta$ ;  
**Output:**  $(\hat{x}_1, \dots, \hat{x}_N)$ ;

- 1 **foreach**  $j$  *s.t.*  $\tau_j \in \mathcal{T}$  **do**
- 2     **if**  $\tau_j \in \mathcal{T}_{\text{con}}$  **then**
- 3         randomly sample a noise  $n_j$  from  $\text{Lap}\left(0, -\frac{\alpha_j}{\ln \beta_j}\right)$ ;
- 4          $\hat{x}_j \leftarrow x_j + n_j$ ;
- 5     **else**
- 6         randomly sample a noise  $n_j$  from  $\text{Lap}\left(0, \frac{1}{\delta_j}\right)$ ;
- 7          $\hat{x}_j \leftarrow \text{sign}(\tilde{x}_j + n_j)$ ;

8 **return**  $(\hat{x}_1, \dots, \hat{x}_N)$ ;

---

Apart from the the vector of the aggregated results  $(x_1, \dots, x_N)$  output by the data aggregation mechanism, the task and continuous task set  $\mathcal{T}$  and  $\mathcal{T}_{\text{con}}$ , the same  $\alpha$  and  $\beta$  vector as in Algorithm 8, 9, and 10, Algorithm 11 also takes as input the vector  $\tilde{\mathbf{x}}$ , where each element  $\tilde{x}_j$  corresponds to one categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$  with

$$\tilde{x}_j = \sum_{i:w_i \in \mathcal{S}, \tau_j \in \Gamma_i} (2\theta_{i,j} - 1)x_{i,j}. \quad (5.31)$$

Clearly, for each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ ,  $\tilde{x}_j$  is its intermediate aggregated result before we convert it to the binary label  $x_j$ . Although not explicitly described, Algorithm 8 keeps track of these intermediate results  $\tilde{x}_j$ 's so that they can be utilized by Algorithm 11. Additionally, the last input parameter to Algorithm 11 is the vector  $\delta$ , where each element  $\delta_j \in (0, 1)$  is a platform-chosen parameter corresponding to the privacy guarantee of a categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ . For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , Algorithm 11 independently samples a random noise  $n_j$  from the Laplacian distribution with mean 0 and scaling  $-\frac{\alpha_j}{\ln \beta_j}$ , denoted as  $\text{Lap}\left(0, -\frac{\alpha_j}{\ln \beta_j}\right)$  (line 3), and adds it to the aggregated result  $x_j$  (line 4). For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , the algorithm randomly samples a noise from the Laplacian distribution with mean 0 and scaling  $\frac{1}{\delta_j}$ , denoted as  $\text{Lap}\left(0, \frac{1}{\delta_j}\right)$  (line 6), and the perturbed result  $\hat{x}_j$  of this task is calculated as  $\text{sign}(\tilde{x}_j + n_j)$  (line 7). Although

adding Laplacian noise as in [29, 39] is a well-established approach to achieve differential privacy, the scaling of the Laplacian distribution is application specific and has to be carefully designed to achieve a desirable trade-off between privacy and data accuracy.

### 5.3.3.2 Analysis

We firstly analyze Algorithm 11's accuracy guarantee for continuous tasks.

**Theorem 31.** *For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , the data perturbation mechanism given in Algorithm 11 satisfies*

$$\Pr[|\widehat{X}_j - X_j| \geq \alpha_j] = \beta_j. \quad (5.32)$$

*Proof.* For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , we use  $N_j$  to denote the random variable representing the random noise sampled from the Laplacian distribution  $\text{Lap}(0, -\frac{\alpha_j}{\ln \beta_j})$ , i.e.,  $N_j \sim \text{Lap}(0, -\frac{\alpha_j}{\ln \beta_j})$ .

Thus,

$$\begin{aligned} \Pr[|\widehat{X}_j - X_j| \geq \alpha_j] &= \Pr[|N_j| \geq \alpha_j] = 2\Pr[N_j \geq \alpha_j] \\ &= 2 \int_{\alpha_j}^{+\infty} -\frac{\ln \beta_j}{2\alpha_j} \exp\left(\frac{z \ln \beta_j}{\alpha_j}\right) dz = \beta_j, \end{aligned}$$

which gives us  $\Pr[|\widehat{X}_j - X_j| \geq \alpha_j] = \beta_j$ . □

Theorem 31 states that  $(\alpha_j, \beta_j)$ -accuracy is guaranteed for the perturbed result compared to the original one before perturbation for every continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ . However, our ultimate goal is to achieve that the perturbed results has satisfactory accuracy compared to ground truths, which is proved in the following Theorem 32.

**Theorem 32.** *For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , the data perturbation mechanism given in Algorithm 11 satisfies*

$$\Pr[|\widehat{X}_j - x_j^*| \geq 2\alpha_j] \leq 1 - (1 - \beta_j)^2. \quad (5.33)$$

*Proof.* As discussed in Section 5.3.1 and 5.3.2, the aggregated result for every continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  satisfies that  $\Pr[|X_j - x_j^*| \geq \alpha_j] \leq \beta_j$ . From Theorem 31 and the fact that  $X_j - x_j^*$  and  $\widehat{X}_j - X_j = N_j$  are two independent random variables, we have

$$\Pr[|\widehat{X}_j - x_j^*| > 2\alpha_j] \leq \Pr[|\widehat{X}_j - X_j| + |X_j - x_j^*| > 2\alpha_j] \leq 1 - (1 - \beta_j)^2,$$

which gives us  $\Pr[|\widehat{X}_j - x_j^*| \geq 2\alpha_j] \leq 1 - (1 - \beta_j)^2$ .  $\square$

Therefore, Theorem 32 gives us that  $(2\alpha_j, 1 - (1 - \beta_j)^2)$ -accuracy is satisfied for the perturbed result of every continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  compared to its ground truth. Next, we analyze Algorithm 11's accuracy guarantee for categorical tasks.

**Theorem 33.** *For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , the data perturbation mechanism given in Algorithm 11 satisfies*

$$\Pr[\widehat{X}_j \neq x_j^*] \leq \frac{\gamma_j + 1}{2}. \quad (5.34)$$

*Proof.* For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , we have that

$$\Pr[\widehat{X}_j \neq x_j^*] = \Pr[\widetilde{X}_j + N_j \geq 0 | x_j^* = -1] \Pr[x_j^* = -1] + \Pr[\widetilde{X}_j + N_j < 0 | x_j^* = +1] \Pr[x_j^* = +1],$$

where  $\widehat{X}_j$  denotes the random variable corresponding to  $\widetilde{x}_j$ , and  $N_j$  denotes the random variable that represents the random noise sampled from the Laplacian distribution  $\text{Lap}(0, \frac{1}{\gamma_j})$ . Then, we have that

$$\begin{aligned} \Pr[\widetilde{X}_j + N_j \geq 0 | x_j^* = -1] &\leq 1 - \Pr[\widetilde{X}_j < 0 | x_j^* = -1] \Pr[N_j < 0] \\ &< 1 - \frac{1 - \gamma_j}{2} = \frac{1 + \gamma_j}{2}, \end{aligned}$$

where the last inequality is because of  $\Pr[\widetilde{X}_j < 0 | x_j^* = -1] > 1 - \gamma_j$  which is an intermediate result in the proof of Theorem 1 in [54]. Similar, we have that  $\Pr[\widetilde{X}_j + N_j < 0 | x_j^* = +1] < \frac{1 + \gamma_j}{2}$ . Therefore, we have that

$$\Pr[\widehat{X}_j \neq x_j^*] \leq \frac{\gamma_j + 1}{2}, \quad (5.35)$$

which exactly proves this Theorem.  $\square$

By Theorem 33, we have that the final perturbed result of each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$  satisfies  $\gamma_j$ -accuracy compared to its ground truth with  $\gamma_j \in (0, 1)$ . Next, in Theorem 34, we analyze the privacy guarantee of the data perturbation mechanism.

**Theorem 34.** *The data perturbation mechanism given in Algorithm 11 satisfies  $\epsilon$ -differential privacy, where the privacy budget  $\epsilon = \max \left\{ \max_{j: \tau_j \in \mathcal{T}_{\text{con}}} \left( -\frac{\ln \beta_j}{\alpha_j} \right), \max_{j: \tau_j \in \mathcal{T}_{\text{cat}}} 2\delta_j \right\}$ .*

*Proof.* Similar to the proof of Theorem 31 and 33, we use  $N_j$  to denote the random variable corresponding to the random noise  $n_j$  sampled by Algorithm 11 for each task  $\tau_j$ . For any  $\mathcal{O} \subseteq \mathbb{R}$  and  $r \in \mathbb{R}$ , we use  $\mathcal{O} - r$  to denote the set  $\{x' = x - r | x \in \mathcal{O}\}$ , and  $x_j^{(i)}$  and  $\hat{x}_j^{(i)}$  to denote the aggregated result for task  $\tau_j$  before and after perturbation when one worker  $w_i$ 's data  $x_{i,j}$  changes. For each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$ , we have  $|x_j - x_j^{(i)}| \leq 1$ , and

$$\begin{aligned} \Pr[\hat{X}_j \in \mathcal{O}] &= \Pr[N_j \in \mathcal{O} - X_j] \\ &= \int_{z \in \mathcal{O} - X_j} -\frac{\ln \beta_j}{2\alpha_j} \exp\left(\frac{|z| \ln \beta_j}{\alpha_j}\right) dz \\ &\leq \exp\left(-\frac{\ln \beta_j}{\alpha_j}\right) \int_{z \in \mathcal{O} - X_j^{(i)}} -\frac{\ln \beta_j}{2\alpha_j} \exp\left(\frac{|z| \ln \beta_j}{\alpha_j}\right) dz \\ &= \exp\left(-\frac{\ln \beta_j}{\alpha_j}\right) \Pr[\hat{X}_j^{(i)} \in \mathcal{O}]. \end{aligned}$$

For each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$ , we use  $\tilde{x}_j^{(i)}$  to denote the value of  $\tilde{x}_j$  when one worker  $w_i$ 's data  $x_{i,j}$  changes, and clearly  $|x_j - x_j^{(i)}| \leq 2$ . Thus, we have that

$$\begin{aligned} \Pr[\tilde{X}_j + N_j \in \mathcal{O}] &= \Pr[N_j \in \mathcal{O} - \tilde{X}_j] \\ &= \int_{z \in \mathcal{O} - \tilde{X}_j} \frac{\delta_j}{2} \exp(-\delta_j |z|) dz \\ &\leq \exp(2\delta_j) \int_{z \in \mathcal{O} - \tilde{X}_j^{(i)}} \frac{\delta_j}{2} \exp(-\delta_j |z|) dz \\ &= \exp(2\delta_j) \Pr[\tilde{X}_j^{(i)} + N_j \in \mathcal{O}]. \end{aligned}$$

As  $\mathcal{O}$  could be any subset of  $\mathbb{R}$ , we let  $\mathcal{O} = [0, +\infty)$ , and get  $\Pr[\tilde{X}_j + N_j \geq 0] \leq \exp(2\delta_j) \Pr[\tilde{X}_j^{(i)} + N_j \geq 0]$ . Thus, we have that

$$\frac{\Pr[\hat{X}_j = +1]}{\Pr[\hat{X}_j^{(i)} = +1]} = \frac{\Pr[\tilde{X}_j + N_j \geq 0]}{\Pr[\tilde{X}_j^{(i)} + N_j \geq 0]} \leq \exp(2\delta_j).$$

Similarly, by letting  $\mathcal{O} = (-\infty, 0)$ , we have that

$$\frac{\Pr[\hat{X}_j = -1]}{\Pr[\hat{X}_j^{(i)} = -1]} = \frac{\Pr[\tilde{X}_j + N_j < 0]}{\Pr[\tilde{X}_j^{(i)} + N_j < 0]} \leq \exp(2\delta_j).$$

Note that the previous analysis focuses on a specific task  $\tau_j$ . The overall privacy budget considering all tasks in  $\mathcal{T}$  is thus  $\epsilon = \max \left\{ \max_{j:\tau_j \in \mathcal{T}_{\text{con}}} \left( -\frac{\ln \beta_j}{\alpha_j} \right), \max_{j:\tau_j \in \mathcal{T}_{\text{cat}}} 2\delta_j \right\}$ .  $\square$

### 5.3.4 Summary of Design Details

Thus far, we have finished the description of the design details of INCEPTION. Its incentive mechanism (Section 5.3.2) selects a set of winners that are more likely to provide reliable data and determines the payments to compensate their sensing and privacy costs. Meanwhile, it approximately minimizes the platform’s total payment (Theorem 30), and satisfies computational efficiency (Theorem 29), truthfulness (Theorem 27), and individual rationality (Theorem 28). Incorporating workers’ reliability levels, the data aggregation mechanism (Section 5.3.1) provides aggregated results with high accuracy (Corollary 2 and 3), and the data perturbation mechanism (Section 5.3.3) adds carefully controlled noises to the aggregated results to achieve differential privacy (Theorem 34), and small degradation of aggregation accuracy (Theorem 31, 32, and 33).

Overall, INCEPTION guarantees  $\max \left\{ \max_{j:\tau_j \in \mathcal{T}_{\text{con}}} \left( -\frac{\ln \beta_j}{\alpha_j} \right), \max_{j:\tau_j \in \mathcal{T}_{\text{cat}}} 2\delta_j \right\}$ -differential privacy,  $(2\alpha_j, 1 - (1 - \beta_j)^2)$ -accuracy for each continuous task  $\tau_j \in \mathcal{T}_{\text{con}}$  (Theorem 32), and  $\frac{\gamma_j+1}{2}$ -accuracy for each categorical task  $\tau_j \in \mathcal{T}_{\text{cat}}$  (Theorem 33). The platform could carefully select the parameter  $\alpha_j, \beta_j, \gamma_j, \delta_j \in (0, 1)$  for every task  $\tau_j$  to achieve satisfactory guarantee for aggregation accuracy and workers’ privacy.

## 5.4 Performance Evaluation

In this section, we introduce the baseline methods, and simulation settings, as well as results.

### 5.4.1 Baseline Methods

Ideally we need to compare the proposed pSRC auction with a truthful and individual rational auction that returns exact optimal solutions to the pSRC-TPM problem. However, because solving the pSRC-TPM problem is notoriously challenging, we instead use the following VCG auction [23, 41] as one of the baseline methods. The VCG auction solves the pSRC-SCM problem optimally and pays every winner according to the VCG payment. This choice is reasonable as the optimal social cost offers a lower bound to the optimal total payment as proved in Lemma 9. Hence, a

good approximation to the optimal social cost indicates a better approximation to the optimal total payment.

Another baseline method is the bidding price effectiveness greedy (BPE-Greedy) auction. Initially, it sorts workers according to an increasing order of their bidding price effectiveness. Winners are selected in this order until the feasibility of the pSRC-TPM problem is satisfied. Its pricing mechanism pays every winner her critical payment as Algorithm 10 does. It is easily provable that the BPE-Greedy auction also satisfies truthfulness and individual rationality.

Furthermore, we compare our weighted data aggregation mechanism with two other baseline aggregation methods, namely the mean and median aggregation. For each continuous task, the mean and median aggregation method simply utilizes, respectively, the mean and median of workers' data as its aggregated result. For each categorical task, the median aggregation method also uses the median of workers' data as the tasks' aggregated result, but the mean aggregation method firstly calculates the mean of workers' data about this task, and then takes the sign of the mean as the aggregated result.

### 5.4.2 Simulation Settings

For simplicity of presenting our simulation results, in this chapter, we consider setting 5.I-5.IV in Table 5.1 where the platform hosts only continuous tasks, and setting 5.V-5.VIII where the platform hosts only categorical tasks. Note that, clearly, our INCEPTION framework is applicable in the scenario where both continuous and categorical tasks are hosted by the platform.

For each continuous task  $\tau_j$ , we generate worker  $w_i$ 's data about this task, i.e.,  $x_{i,j}$ , from a normal distribution with mean  $\mu_{i,j}$  and standard deviation  $\sigma_{i,j}$ , truncated within the range  $[0, 1]$ . The value of  $\theta_{i,j}$  for each continuous task  $\tau_j$  is calculated by platform as

$$\theta_{i,j} = \frac{c_{i,j}\sigma_{i,j}}{\sqrt{2\pi}} \left( 2 \exp\left(\frac{-b_{i,j}^2}{2\sigma_{i,j}^2}\right) - \exp\left(\frac{-a_{i,j}^2}{2\sigma_{i,j}^2}\right) - \exp\left(\frac{-(1-a_{i,j})^2}{2\sigma_{i,j}^2}\right) \right) + c_{i,j}b_{i,j} \left( \Phi\left(\frac{-a_{i,j}}{\sigma_{i,j}}\right) + \Phi\left(\frac{1-a_{i,j}}{\sigma_{i,j}}\right) - 2\Phi\left(\frac{-b_{i,j}}{\sigma_{i,j}}\right) \right),$$

where  $c_{i,j} = \left(\Phi\left(\frac{1-\mu_{i,j}}{\sigma_{i,j}}\right) - \Phi\left(-\frac{\mu_{i,j}}{\sigma_{i,j}}\right)\right)^{-1}$ ,  $b_{i,j} = \mu_{i,j} - x_j^*$ ,  $a_{i,j} = x_j^* + b_{i,j}$ , and  $\Phi(\cdot)$  denotes the c.d.f. of the standard normal distribution. We omit the derivation for  $\theta_{i,j}$  due to space limit. The

parameter settings for the scenarios with only continuous tasks are given in Table 5.1.

Setting	$\alpha_j, \beta_j$	$c_i^s, c_i^p$	$\mu_{i,j}, x_j^*$	$\sigma_{i,j}$	$ \Gamma_i^* $	$N$	$K$
5.I	(0, 0.1]	[1, 2]	[0, 1]	[1, 2]	[15, 20]	[91, 120]	40
5.II	(0, 0.1]	[1, 2]	[0, 1]	[1, 2]	[15, 20]	100	[21, 50]
5.III	(0, 0.1]	[1, 2]	[0, 1]	[1, 2]	[25, 35]	[2100, 5000]	500
5.IV	(0, 0.1]	[1, 2]	[0, 1]	[1, 2]	[25, 35]	1000	[710, 1000]

Table 5.1: Simulation setting 5.I-5.IV (continuous tasks only)

In setting 5.I and 5.II,  $\alpha_j, \beta_j, c_i^s, c_i^p, x_j^*, \mu_{i,j}, \sigma_{i,j}$ , and  $|\Gamma_i^*|$  are generated uniformly at random from the intervals given in Table 5.1. The bundle  $\Gamma_i^*$  contains  $|\Gamma_i^*|$  tasks randomly chosen from  $\mathcal{T}$ . In setting 5.I, we fix the number of tasks as 40 and vary the number of workers from 91 to 120. In contrast, we fix the number of workers as 100 and vary the number of tasks from 21 to 50 in setting 5.II. In setting 5.III and 5.IV,  $\alpha_j, \beta_j, c_i^s, c_i^p, x_j^*, \mu_{i,j}, \sigma_{i,j}$ , and  $|\Gamma_i^*|$  are generated in the same way as in setting 5.I and 5.II from the intervals given in Table 5.1. Different from the previous two settings, setting 5.III and 5.IV take instances with larger sizes, given in Table 5.1, as inputs. Next, we give our parameter settings for the scenarios with only categorical tasks in Table 5.2.

Setting	$\gamma_j, \delta_j$	$c_i^s, c_i^p$	$x_j^*$	$\theta_{i,j}$	$ \Gamma_i^* $	$N$	$K$
5.V	(0, 0.1]	[1, 2]	$\{-1, +1\}$	(0, 1)	[15, 20]	[91, 120]	40
5.VI	(0, 0.1]	[1, 2]	$\{-1, +1\}$	(0, 1)	[15, 20]	100	[21, 50]
5.VII	(0, 0.1]	[1, 2]	$\{-1, +1\}$	(0, 1)	[25, 35]	[2100, 5000]	500
5.VIII	(0, 0.1]	[1, 2]	$\{-1, +1\}$	(0, 1)	[25, 35]	1000	[710, 1000]

Table 5.2: Simulation setting 5.V-5.VIII (categorical tasks only)

In setting 5.V and 5.VI,  $\gamma_j, \delta_j, c_i^s, c_i^p, x_j^*, \theta_{i,j}$ , and  $|\Gamma_i^*|$  are generated uniformly at random from the intervals given in Table 5.2. The bundle  $\Gamma_i^*$  contains  $|\Gamma_i^*|$  tasks randomly chosen from  $\mathcal{T}$ . In setting 5.V, we fix the number of tasks as 40 and vary the number of workers from 91 to 120. In contrast, we fix the number of workers as 100 and vary the number of tasks from 21 to 50 in setting 5.VI. In setting 5.VII and 5.VIII, the parameters  $\gamma_j, \delta_j, c_i^s, c_i^p, x_j^*, \theta_{i,j}$ , and  $|\Gamma_i^*|$  are generated in the same way as in setting 5.V and 5.VI from the intervals given in Table 5.2. Different from the previous two settings, setting 5.VII and 5.VIII take instances with larger sizes as inputs. Note that we leave the study of the values of these parameters in real-world scenarios in our future work. The optimal solutions to the pSRC-SCM problem are computed using the GUROBI solver [6].

### 5.4.3 Simulation Results

Figure 5.2-5.7 show our simulation results on setting 5.I-5.IV with only continuous tasks. Figure 5.2 and 5.3 show that the platform’s total payment of the pSRC auction is far less than that of the BPE-Greedy auction and close to the optimal social cost given by the VCG auction. As the optimal social cost lower bounds the optimal total payment, the pSRC auction gives us close-to-optimal total payment. Next, we compare the execution time of the VCG and the BPE-Greedy auction.

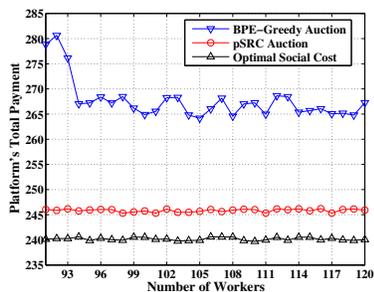


Figure 5.2: Platform’s total payment (setting 5.I)

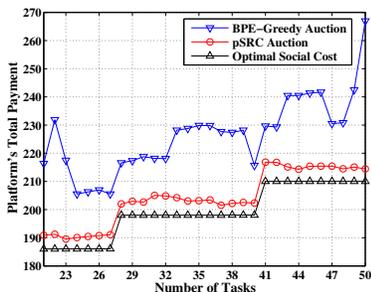


Figure 5.3: Platform’s total payment (setting 5.II)

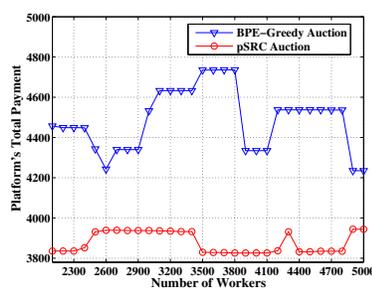


Figure 5.4: Platform’s total payment (setting 5.III)

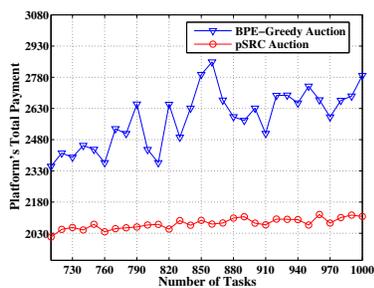


Figure 5.5: Platform’s total payment (setting 5.IV)

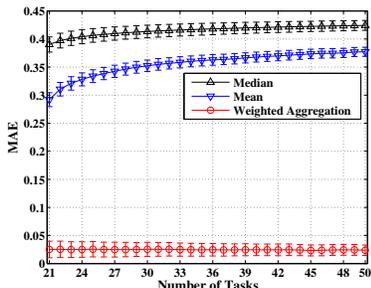


Figure 5.6: MAE of data aggregation (continuous tasks)

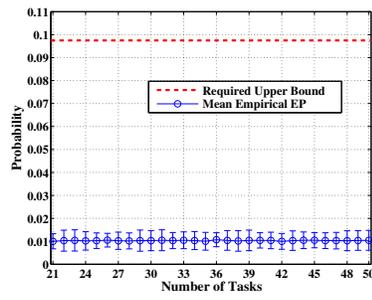


Figure 5.7: EP after perturbation (continuous tasks)

From Table 5.3, we observe that the VCG auction has excessively long running time so that it can hardly be utilized in practice. The running time of the VCG auction lower bounds that of the auction that gives us the optimal total payment, because solving the pSRC-SCM problem is in fact easier and faster than solving the pSRC-TPM problem. Hence, calculating the optimal total payment becomes computationally infeasible in practice. However, the execution time of the pSRC auction keeps in the order of microsecond, which is much less that of the VCG auction.

In Figure 5.4 and 5.5, we show our simulation results about the platform’s total payment for setting 5.III and 5.IV with larger-size problem instances where the VCG auction is not able to

$N$	91	95	99	103	107	111	115	119
VCG	20.23	79.11	227.5	257.7	308.7	836.4	1199	1537
pSRC	0.008	0.009	0.007	0.008	0.008	0.006	0.007	0.006
$K$	21	25	29	33	37	41	45	49
VCG	0.300	6.676	13.09	30.60	1063	1160	1330	1677
pSRC	0.003	0.005	0.003	0.007	0.009	0.009	0.003	0.003

Table 5.3: Execution time (seconds) for setting 5.I and 5.II

terminate in reasonable time. We can observe that the proposed pSRC auction still gives us a total payment far less than that of the BPE-Greedy auction.

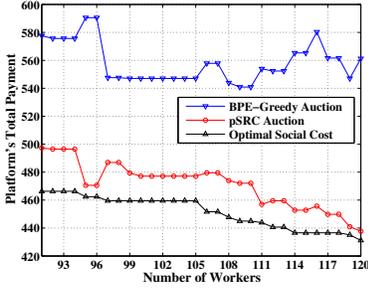


Figure 5.8: Platform's total payment (setting 5.V)

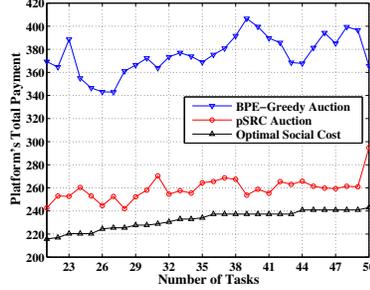


Figure 5.9: Platform's total payment (setting 5.VI)

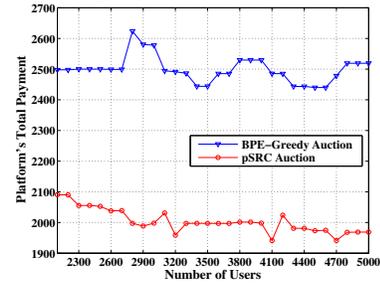


Figure 5.10: Platform's total payment (setting 5.VII)

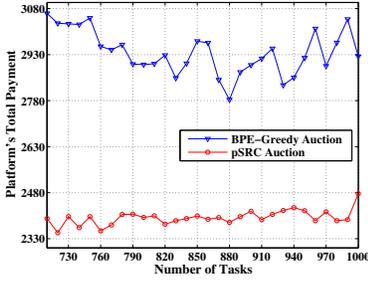


Figure 5.11: Platform's total payment (setting 5.VIII)

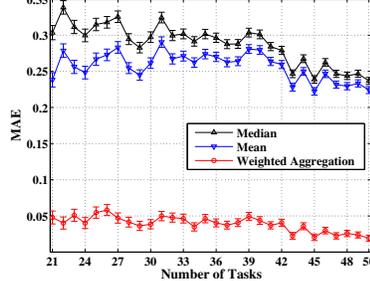


Figure 5.12: MAE of data aggregation (categorical tasks)

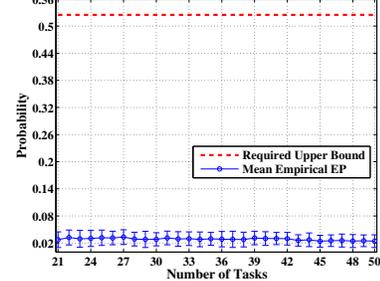


Figure 5.13: EP after perturbation (categorical tasks)

We evaluate the accuracy guarantee of INCEPTION in setting 5.II with a minor change of the parameter  $\beta_j$ , i.e.,  $\beta_j$  is fixed as 0.05 for every task  $\tau_j$  to simplify presentation. We compare the mean absolute error (MAE) for all tasks, defined as  $\text{MAE} = \frac{1}{K} \sum_{j:\tau_j \in \mathcal{T}} |x_j - x_j^*|$ , of the weighted aggregation mechanism given in Algorithm 8 with those of the mean and median aggregation. The simulation for each combination of worker and task number is repeated for 10000 times and the means and standard deviations of the MAEs are plotted. We observe from Figure 5.6 that the MAE of our weighted aggregation is far less than those of the mean and median aggregation. Then, we show simulation results regarding  $\Pr[|\hat{X}_j - x_j^*| \geq \alpha_j]$ , referred to as the error probability

(EP) of the perturbed results for task  $\tau_j$ . After 10000 repetitions of the simulation for any specific combination of worker and task number, empirical values for the EPs are calculated and we plot the means and standard deviations of the empirical EPs over all tasks.

Next, we show our simulation results for setting 5.V-5.VIII with only categorical tasks in Figure 5.8-5.13, which share similar trends as Figure 5.2-5.7. The simulation setting for Figure 5.12 and 5.13 is the same as setting 5.IV except that  $\gamma_j$  and  $\delta_j$  for each task  $\tau_j$  are fixed as 0.1. In Figure 5.13, EP is defined as  $\Pr[\widehat{X}_j \neq x_j^*]$ , whose empirical value is calculated in the same way as that for a continuous task in Figure 5.7.

$N$	91	95	99	103	107	111	115	119
VCG	7.397	25.31	115.5	225.6	312.4	517.4	1059	1105
pSRC	0.016	0.018	0.018	0.019	0.020	0.019	0.021	0.024
$K$	21	25	29	33	37	41	45	49
VCG	5.400	16.33	33.90	500.4	735.9	1050	1100	1507
pSRC	0.016	0.017	0.019	0.019	0.020	0.021	0.023	0.024

Table 5.4: Execution time (seconds) for setting 5.V and 5.VI

Furthermore, we show in Table 5.4 the comparison between the execution time of the VCG and the BPE-Greedy auction for setting 5.V and 5.VI. Clearly, similar to Table 5.3, Table 5.4 also shows that execution time of the pSRC auction is much less than that of the VCG auction.

## 5.5 Related Work

As previously mentioned, different from most of the prior work [20, 22, 27, 28, 31–33, 35, 37, 43–45, 50, 51, 54, 57, 58, 60, 62, 66, 67, 71–75, 82, 86, 88, 89, 91, 92, 95, 97, 98, 100, 106, 107, 110–113, 115–124], we explicitly incorporate workers’ *reliability* and *privacy costs* (motivated by [34, 39]) into the incentive mechanism and provide an integrated design of the incentive, data aggregation, and data perturbation mechanism.

One line of past literature [56, 64, 65, 79, 90, 103–105] investigate privacy-preserving data collection or aggregation in mobile sensing or other various application scenarios. Unlike this chapter, most of them [56, 64, 65, 79, 90] do not consider the issue of incentives. [103–105] protect workers’ privacy against an untrusted platform, whereas, in this chapter, we consider the platform as trusted, and preserve workers’ privacy from the adversaries outside the MCS system who try to infer workers’ data using the publicly available aggregated results. Another set of existing

work [51, 66, 67, 71, 86, 97], related but orthogonal to this chapter, studies privacy-preserving incentive mechanisms for mobile sensing systems. Instead of data privacy, they protect workers' anonymity [66, 67, 86] or bid privacy [51, 71, 97] within the incentive mechanisms.

## 5.6 Conclusion

In this chapter, we propose INCEPTION, a novel MCS system framework that integrates an incentive, a data aggregation, and a data perturbation mechanism. Its incentive mechanism selects reliable workers, and compensates their costs for sensing and privacy leakage, which meanwhile satisfies truthfulness and individual rationality. Its data aggregation mechanism incorporates workers' reliability to generate highly accurate aggregated results, and its data perturbation mechanism ensures satisfactory guarantee for workers' privacy, as well as the accuracy for the final perturbed results. The desirable properties of INCEPTION are validated through both theoretical analysis and extensive simulations.

## Chapter 6

# Conclusions and Future Work

In this section, I conclude the thesis by first summarizing the research findings, and then discussing future research directions.

### 6.1 Summary

In this thesis, I develop four incentive mechanisms to serve the objective of effectively stimulating worker participation in MCS systems, with each of them focusing on one or multiple crucial facets, such as QoI awareness and preservation of workers' bid or data privacy. Note that although I focus on MCS systems in this thesis, the various proposed incentive mechanisms could potentially be utilized to incentivize participation in general-purpose crowdsourcing systems (e.g., Amazon Mechanical Turk), as well, after minor adaptations.

#### 6.1.1 QoI Aware Incentive Mechanisms for MCS Systems

The first issue that I consider in the design of incentive mechanisms is QoI awareness. As low quality sensory data could possibly lead to inaccurate aggregated sensing results or false decisions by the platform, which could eventually result in invaluable loss, QoI is clearly an important metric that should be considered, but has been ignored by most of the prior work. Therefore, in Chapter 2, I propose QoI aware incentive mechanisms, which adopt reverse auction-based frameworks, and tackle workers' strategic behavior. Specifically, the proposed mechanisms yield close-to-optimal social welfare in a computationally efficient manner, which meanwhile satisfy other crucial desirable properties, namely truthfulness and individual rationality.

### **6.1.2 Incentivizing Multi-Requester Mobile Crowd Sensing**

The second issue addressed in this thesis is to effectively incentivize participation in MCS systems, where three parties, including the data requesters, a platform, as well as a crowd of participating workers co-exist, which is a different and, in fact, more practical scenario compared to the one considered in most of the past literature, where there is only one data requester who also serves as the platform in the MCS system. To achieve this end, in Chapter 3, I propose CENTURION, a double auction-based incentive mechanism, which involves auctions among not only the workers, but also the data requesters, and is able to incentivize the participation of both data requesters and workers. I show through rigorous theoretical analyses that the proposed mechanism bears many desirable properties, including truthfulness, individual rationality, computational efficiency, as well as non-negative social welfare.

### **6.1.3 Bid Privacy-Preserving Incentive Mechanism for MCS Systems**

In practice, although the platform is oftentimes considered to be trusted, there usually exist honest-but-curious workers who strictly follow the protocol of the system, but try to infer information about other workers' bids in auction-based incentive mechanisms. Therefore, the third issue considered in this thesis is to prevent workers from being disincentivized by excessive leakage of bid privacy. To address this issue, I propose, in Chapter 4, a bid privacy-preserving incentive mechanism for MCS systems. I incorporate the notion of differential privacy, and ensure that the change in any worker's bid will not bring a significant change to the mechanism's payments to participating workers. Apart from preserving the privacy of workers' bids, the proposed mechanism also bears a suite of other desirable properties, including approximate truthfulness, individual rationality, computational efficiency, as well as yielding a guaranteed approximation ratio to the platform's optimal total payment.

### **6.1.4 Incentivizing Privacy-Preserving Data Aggregation in MCS Systems**

Besides the issue of bid privacy discussed in Chapter 4, participating workers in MCS systems usually face, as well, another type of equally possible and severe privacy breach, which is the leakage of their data privacy. Therefore, it is entirely necessary for an MCS system to contain a data

perturbation module that preserves workers’ data privacy by carefully perturbing the aggregated results before they are published. In real practice, the various modules of an MCS system are far from isolated, but, in fact, interact with each other, and thus affect each other’s design.

Thus, with this point in mind, in Chapter 5, I propose INCEPTION, which is the first integrated framework for MCS systems with an incentive, a data aggregation, and a data perturbation mechanism. Specifically, INCEPTION has an auction-based incentive mechanism that selects reliable workers and compensates their costs for both sensing and privacy leakage, which meanwhile satisfies truthfulness and individual rationality, and minimizes the platforms total payment for worker recruiting with a guaranteed approximation ratio. The data aggregation mechanism of INCEPTION also incorporates workers’ reliability and generates highly accurate aggregated results. Its data perturbation mechanism ensures satisfactory guarantee for the protection of workers privacy, as well as the accuracy of the final perturbed results.

## 6.2 Future Research Directions

I envision that applications based on crowd sensing will continue to gain increasingly great popularity in the future, and eventually become integral parts of people’s everyday living and working. This will bring us great opportunities as well as new challenges. I will continue my exploration and research efforts, especially in regard to effectively incentivizing worker participation in MCS systems. Below, I list a few problems I am keenly interested in exploring next.

First, although I focus on designing mechanisms that provide monetary incentives in this thesis, there exist various types of non-monetary incentives (e.g., the aggregated results of interest to workers, enhanced quality of service), which could potentially be very useful to incentivize worker participation. Thus, the first research task that I aim to carry out is to explore the possibility of incentivizing worker participation in MCS systems more effectively by exploiting both monetary and non-monetary incentives.

Second, in order to identify truthful values from crowd workers’ noisy or even conflicting sensory data, truth discovery algorithms [68–70, 78, 96, 101], which jointly estimate workers’ data quality and the underlying truths through quality-aware data aggregation, have drawn significant attention. However, the power of truth discovery algorithms could not be fully unleashed in MCS systems,

unless the platform properly deals with workers' strategic reduction of their costly sensing effort (e.g., time, resources, attention), which inevitably deteriorates the quality of their sensory data, and further impairs the aggregation accuracy. To address this problem, a feasible solution is to utilize a carefully-designed payment mechanism that offers sufficient amount of payments to incentivize high-effort sensing from workers, which meanwhile keeps the overall payment below a budget.

Third, in practice, not only is a worker's QoI typically unknown *a priori*, but also may the workers, as well as the deadline-sensitive sensing tasks arrive sequentially in an online manner. In this case, workers' QoI has to be estimated over time, as the tasks are allocated, usually with the lack of ground truths to verify the quality of workers' sensory data. Therefore, the third research task that I aim to explore in the future is to integrate QoI estimation into the incentive mechanism, which enables the platform to learn workers' QoI over time, and further dynamically adjusts worker selection, as well as the payments to workers, according to the estimated QoI.

Additionally, there are many other research opportunities waiting for us to explore, such as boosting the performance of machine learning algorithms (e.g., regression, classification, deep learning) when data sources are strategic, the detection of and defense against malicious user in crowd sensing systems, and so forth.

I seek to resolve these challenges through not only independent but also collaborative work using all kinds of analytic tools such as game theory, algorithm, machine learning, and optimization. I believe these researches can give rise to and benefit tremendously a whole new range of crowd sensing-based applications, making our world safer, smarter and better.

# References

- [1] Amazon mechanical turk. <https://www.mturk.com/>.
- [2] Clickworker. <https://www.clickworker.com/>.
- [3] Crowdmag. <https://www.ngdc.noaa.gov/geomag/crowdmag.shtml>.
- [4] Crowdmed. <https://www.crowdmed.com/>.
- [5] Flunearyou. <https://flunearyou.org/>.
- [6] Gurobi solver. <http://www.gurobi.com/>.
- [7] Medwatcher. <https://medwatcher.org/>.
- [8] Myheartmap. <http://www.med.upenn.edu/myheartmap/>.
- [9] Noisemap. <http://www.noisemap.ltd.uk/wordpress/>.
- [10] Waze. <https://www.waze.com/>.
- [11] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998.
- [12] M. Babaioff, R. Lavi, and E. Pavlov. Mechanism design for single-value domains. In *Proceedings of the 20th AAAI National Conference on Artificial Intelligence (AAAI)*, pages 241–247. AAAI Press, 2005.
- [13] M. Babaioff, R. Lavi, and E. Pavlov. Single-value combinatorial auctions and algorithmic implementation in undominated strategies. *Journal of the ACM*, 56(1):4:1–4:32, 2009.
- [14] M. Babaioff and N. Nisan. Concurrent auctions across the supply chain. In *Proceedings of the 3rd ACM Conference on Electronic Commerce (EC)*, pages 1–10. ACM, 2001.
- [15] L. Blumrosen and N. Nisan. Combinatorial auctions. *Algorithmic game theory*, 267, 2007.
- [16] S. Chen, M. Li, K. Ren, X. Fu, and C. Qiao. Rise of the indoor crowd: Reconstruction of building interior view via mobile crowdsourcing. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 59–71. ACM, 2015.
- [17] S. Chen, M. Li, K. Ren, and C. Qiao. Crowd map: Accurate reconstruction of indoor floor plans from crowdsourced sensor-rich videos. In *Proceedings of the 35th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 1–10. IEEE, 2015.

- [18] X. Chen and J. Huang. Spatial spectrum access game: Nash equilibria and distributed learning. In *Proceedings of the 13th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 205–214. ACM, 2012.
- [19] X. Chen, B. Proulx, X. Gong, and J. Zhang. Social trust and social reciprocity based cooperative d2d communications. In *Proceedings of the 14th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 187–196. ACM, 2013.
- [20] Y. Chen, B. Li, and Q. Zhang. Incentivizing crowdsourcing systems with network effects. In *Proceedings of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–9. IEEE, 2016.
- [21] Y. Cheng, X. Li, Z. Li, S. Jiang, Y. Li, J. Jia, and X. Jiang. Aircloud: A cloud-based air-quality monitoring system for everyone. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 251–265. ACM, 2014.
- [22] M. H. Cheung, R. Southwell, F. Hou, and J. Huang. Distributed time-sensitive task selection in mobile crowdsensing. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 157–166. ACM, 2015.
- [23] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.
- [24] K. Deshmukh, A. V. Goldberg, J. D. Hartline, and A. R. Karlin. Truthful and competitive double auctions. In *Proceedings of the 10th Annual European Symposium on Algorithms (ESA)*, pages 361–373. Springer-Verlag, 2002.
- [25] M. Dong, H. Jin, G. Sun, X. Wang, W. Liu, and X. Wang. Non-cooperative game based social welfare maximizing bandwidth allocation in wsns. In *Proceedings of the 54th IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5. IEEE, 2011.
- [26] M. Dong, G. Sun, X. Wang, and Q. Zhang. Combinatorial auction with time-frequency flexibility in cognitive radio networks. In *Proceedings of the 31st Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2282–2290. IEEE, 2012.
- [27] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand. Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing. In *Proceedings of the 31st Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1701–1709. IEEE, 2012.
- [28] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand. Motivating smartphone collaboration in data acquisition and distributed computing. *IEEE Transactions on Mobile Computing*, 13(10):2320–2333, 2014.
- [29] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Theory of Cryptography Conference (TCC)*, pages 265–284. Springer, 2006.
- [30] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan. The pothole patrol: Using a mobile sensor network for road surface monitoring. In *Proceedings of the 6th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*, pages 29–39. ACM, 2008.

- [31] B. Faltings, J. J. Li, and R. Jurca. Incentive mechanisms for community sensing. *IEEE Transactions on Computers*, 63(1):115–128, 2014.
- [32] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos. Trac: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing. In *Proceedings of the 33rd Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1231–1239. IEEE, 2014.
- [33] Z. Feng, Y. Zhu, Q. Zhang, H. Zhu, J. Yu, J. Cao, and L. M. Ni. Towards truthful mechanisms for mobile crowdsourcing with dynamic smartphones. In *Proceedings of the 34th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 11–20. IEEE, 2014.
- [34] L. K. Fleischer and Y.-H. Lyu. Approximately optimal auctions for selling privacy when costs are correlated with data. In *Proceedings of the 13th ACM Conference on Electronic Commerce (EC)*, pages 568–585. ACM, 2012.
- [35] X. Gan, X. Wang, W. Niu, G. Hang, X. Tian, X. Wang, and J. Xu. Incentivize multi-class crowd labeling under budget constraint. *IEEE Journal on Selected Areas in Communications*, PP(99):1–1, 2017.
- [36] R. K. Ganti, N. Pham, H. Ahmadi, S. Nangia, and T. F. Abdelzaher. Greengps: A participatory sensing fuel-efficient maps application. In *Proceedings of the 8th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*, pages 151–164. ACM, 2010.
- [37] L. Gao, F. Hou, and J. Huang. Providing long-term participation incentive in participatory sensing. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2803–2811. IEEE, 2015.
- [38] R. Gao, M. Zhao, T. Ye, F. Ye, Y. Wang, K. Bian, T. Wang, and X. Li. Jigsaw: Indoor floor plan reconstruction via mobile crowdsensing. In *Proceedings of the 20th Annual ACM International Conference on Mobile Computing and Networking (Mobicom)*, pages 249–260. ACM, 2014.
- [39] A. Ghosh and A. Roth. Selling privacy at auction. In *Proceedings of the 12th ACM Conference on Electronic Commerce (EC)*, pages 199–208. ACM, 2011.
- [40] A. Gopinathan and Z. Li. A prior-free revenue maximizing auction for secondary spectrum access. In *Proceedings of the 30th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 86–90. IEEE, 2011.
- [41] T. Groves. Incentives in teams. *Econometrica*, 41(4):617–631, 1973.
- [42] A. Gupta, K. Ligett, F. McSherry, A. Roth, and K. Talwar. Differentially private combinatorial optimization. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1106–1125. ACM, 2010.
- [43] K. Han, H. Huang, and J. Luo. Posted pricing for robust crowdsensing. In *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 261–270. ACM, 2016.

- [44] K. Han, C. Zhang, and J. Luo. Taming the uncertainty: Budget limited robust crowdsensing through online learning. *IEEE/ACM Transactions on Networking*, 24(3):1462–1475, 2016.
- [45] S. He, D. H. Shin, J. Zhang, and J. Chen. Toward optimal allocation of location dependent tasks in crowdsensing. In *Proceedings of the 33rd Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 745–753. IEEE, 2014.
- [46] S. Hu, L. Su, H. Liu, H. Wang, and T. F. Abdelzaher. Smartroad: Smartphone-based crowd sensing for traffic regulator detection and identification. *ACM Transactions on Sensor Networks*, 11(4):55:1–55:27, 2015.
- [47] J. Huang, R. A. Berry, and M. L. Honig. Auction-based spectrum sharing. *Mobile Networks and Applications*, 11(3):405–418, 2006.
- [48] N. Immorlica, A. R. Karlin, M. Mahdian, and K. Talwar. Balloon popping with applications to ascending auctions. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 104–112. IEEE, 2007.
- [49] J. Jia, Q. Zhang, Q. Zhang, and M. Liu. Revenue generation for truthful spectrum auction in dynamic spectrum access. In *Proceedings of the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 3–12. ACM, 2009.
- [50] H. Jin, L. Su, D. Chen, K. Nahrstedt, and J. Xu. Quality of information aware incentive mechanisms for mobile crowd sensing systems. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 167–176. ACM, 2015.
- [51] H. Jin, L. Su, B. Ding, K. Nahrstedt, and N. Borisov. Enabling privacy-preserving incentives for mobile crowd sensing systems. In *Proceedings of the 36th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 344–353. IEEE, 2016.
- [52] H. Jin, L. Su, and K. Nahrstedt. Centurion: Incentivizing multi-requester mobile crowd sensing. *CoRR*, abs/1701.01533, 2017.
- [53] H. Jin, L. Su, H. Xiao, and K. Nahrstedt. Inception: Incentivizing privacy-preserving data aggregation for mobile crowd sensing systems. In *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 341–350. ACM, 2016.
- [54] H. Jin, L. Su, H. Xiao, and K. Nahrstedt. Centurion: Incentivizing multi-requester mobile crowd sensing. In *Proceedings of the 36th Annual IEEE International Conference on Computer Communications (INFOCOM)*. IEEE, 2017.
- [55] H. Jin, G. Sun, X. Wang, and Q. Zhang. Spectrum trading with insurance in cognitive radio networks. In *Proceedings of the 31st Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2041–2049. IEEE, 2012.
- [56] T. Jung, X. Mao, X. Y. Li, S. J. Tang, W. Gong, and L. Zhang. Privacy-preserving data aggregation without secure channel: Multivariate polynomial evaluation. In *Proceedings of the 32nd Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2634–2642. IEEE, 2013.

- [57] M. Karaliopoulos, I. Koutsopoulos, and M. Titsias. First learn then earn: Optimizing mobile crowdsensing campaigns through data-driven user profiling. In *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 271–280. ACM, 2016.
- [58] M. Karaliopoulos, O. Telelis, and I. Koutsopoulos. User recruitment for mobile crowdsensing over opportunistic networks. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2254–2262. IEEE, 2015.
- [59] M. Kearns, M. Pai, A. Roth, and J. Ullman. Mechanism design in large games: Incentives and privacy. In *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science (ITCS)*, pages 403–410. ACM, 2014.
- [60] I. Koutsopoulos. Optimal incentive-driven design of participatory sensing systems. In *Proceedings of the 32nd Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1402–1410. IEEE, 2013.
- [61] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.
- [62] J. S. Lee and B. Hoh. Sell your experiences: a market mechanism based incentive for participatory sensing. In *Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 60–68. IEEE, 2010.
- [63] H. Li, B. Zhao, and A. Fuxman. The wisdom of minority: Discovering and targeting the right group of workers for crowdsourcing. In *Proceedings of the 23rd ACM International Conference on World Wide Web (WWW)*, pages 165–176. ACM, 2014.
- [64] Q. Li and G. Cao. Efficient and privacy-preserving data aggregation in mobile sensing. In *Proceedings of the 20th IEEE International Conference on Network Protocols (ICNP)*, pages 1–10. IEEE, 2012.
- [65] Q. Li and G. Cao. Efficient privacy-preserving stream aggregation in mobile sensing with low aggregation error. In *Proceedings of the 33rd International Symposium on Privacy Enhancing Technologies Symposium (PETS)*, pages 60–81. Springer, 2013.
- [66] Q. Li and G. Cao. Providing privacy-aware incentives for mobile sensing. In *Proceedings of the 11th IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 76–84. IEEE, 2013.
- [67] Q. Li and G. Cao. Providing efficient privacy-aware incentives for mobile sensing. In *Proceedings of the 34th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 208–217. IEEE, 2014.
- [68] Q. Li, Y. Li, J. Gao, L. Su, B. Zhao, M. Demirbas, W. Fan, and J. Han. A confidence-aware approach for truth discovery on long-tail data. *Proceedings of the VLDB Endowment*, 8(4):425–436, 2014.
- [69] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 1187–1198. ACM, 2014.

- [70] Y. Li, J. Gao, C. Meng, Q. Li, L. Su, B. Zhao, W. Fan, and J. Han. A survey on truth discovery. *ACM SIGKDD Explorations Newsletter*, 17(2):1–16, 2016.
- [71] J. Lin, D. Yang, M. Li, J. Xu, and G. Xue. Bidguard: A framework for privacy-preserving crowdsensing incentive mechanisms. In *Proceedings of the 4th IEEE Conference on Communications and Network Security (CNS)*, pages 145–153. IEEE, 2016.
- [72] T. Luo, S. K. Das, H. P. Tan, and L. Xia. Incentive mechanism design for crowdsourcing: An all-pay auction approach. *ACM Transactions on Intelligent Systems and Technology*, 7(3):35:1–35:26, 2016.
- [73] T. Luo, S. S. Kanhere, S. K. Das, and H. P. Tan. Incentive mechanism design for heterogeneous crowdsourcing using all-pay contests. *IEEE Transactions on Mobile Computing*, 15(9):2234–2246, 2016.
- [74] T. Luo, S. S. Kanhere, H. P. Tan, F. Wu, and H. Wu. Crowdsourcing with tullock contests: A new perspective. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2515–2523. IEEE, 2015.
- [75] T. Luo, H. P. Tan, and L. Xia. Profit-maximizing incentive for participatory sensing. In *Proceedings of the 33rd Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 127–135. IEEE, 2014.
- [76] R. P. McAfee. A dominant strategy double auction. *Journal of Economic Theory*, 56(2):434–450, 1992.
- [77] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 94–103. IEEE, 2007.
- [78] C. Meng, W. Jiang, Y. Li, J. Gao, L. Su, H. Ding, and Y. Cheng. Truth discovery on crowd sensing of correlated entities. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 169–182. ACM, 2015.
- [79] C. Miao, W. Jiang, L. Su, Y. Li, S. Guo, Z. Qin, H. Xiao, J. Gao, and K. Ren. Cloud-enabled privacy-preserving truth discovery in crowd sensing systems. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 183–196. ACM, 2015.
- [80] P. Mohan, V. N. Padmanabhan, and R. Ramjee. Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 323–336. ACM, 2008.
- [81] R. B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.
- [82] P. Naghizadeh and M. Liu. Perceptions and truth: A mechanism design approach to crowdsourcing reputation. *IEEE/ACM Transactions on Networking*, 24(1):163–176, 2016.
- [83] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic game theory*, volume 1. Cambridge University Press Cambridge, 2007.
- [84] K. Nissim, C. Orlandi, and R. Smorodinsky. Privacy-aware mechanism design. In *Proceedings of the 13th ACM Conference on Electronic Commerce (EC)*, pages 774–789. ACM, 2012.

- [85] K. Nissim, R. Smorodinsky, and M. Tennenholtz. Approximately optimal mechanism design via differential privacy. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS)*, pages 203–213. ACM, 2012.
- [86] X. Niu, M. Li, Q. Chen, Q. Cao, and H. Wang. Eppi: An e-cent-based privacy-preserving incentive mechanism for participatory sensing systems. In *Proceedings of the 33rd IEEE International Performance Computing and Communications Conference (IPCCC)*, pages 1–8. IEEE, 2014.
- [87] D. Oleson, A. Sorokin, G. Laughlin, V. Hester, J. Le, and L. Biewald. Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. In *Proceedings of the 11th AAAI Conference on Human Computation (HCOMP)*, pages 43–48. AAAI Press, 2011.
- [88] D. Peng, F. Wu, and G. Chen. Pay as how well you do: A quality based incentive mechanism for crowdsensing. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 177–186. ACM, 2015.
- [89] L. Pu, X. Chen, J. Xu, and X. Fu. Crowdlet: Optimal worker recruitment for self-organized mobile crowdsourcing. In *Proceedings of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–9. IEEE, 2016.
- [90] F. Qiu, F. Wu, and G. Chen. Privacy and quality preserving multimedia data aggregation for participatory sensing systems. *IEEE Transactions on Mobile Computing*, 14(6):1287–1300, 2015.
- [91] G. Radanovic and B. Faltings. Incentive schemes for participatory sensing. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (IFAAMAS)*, pages 1081–1089. IFAAMAS, 2015.
- [92] G. Radanovic, B. Faltings, and R. Jurca. Incentives for effort in crowdsourcing using the peer truth serum. *ACM Transactions on Intelligent Systems and Technology*, 7(4):48:1–48:28, 2016.
- [93] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th Annual ACM International Conference on Mobile Computing and Networking (Mobicom)*, pages 293–304. ACM, 2012.
- [94] S. Rajagopalan and V. V. Vazirani. Primal-dual rnc approximation algorithms for (multi)-set (multi)-cover and covering integer programs. In *Proceedings of the 34th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 322–331. IEEE, 1993.
- [95] Z. Song, C. H. Liu, J. Wu, J. Ma, and W. Wang. Qoi-aware multitask-oriented dynamic participant selection with budget constraints. *IEEE Transactions on Vehicular Technology*, 63(9):4618–4632, 2014.
- [96] L. Su, Q. Li, S. Hu, S. Wang, J. Gao, H. Liu, T. F. Abdelzaher, J. Han, X. Liu, Y. Gao, and L. Kaplan. Generalized decision aggregation in distributed sensing systems. In *Proceedings of the 35th IEEE Real-Time Systems Symposium (RTSS)*, pages 1–10. IEEE, 2014.
- [97] J. Sun and H. Ma. Privacy-preserving verifiable incentive mechanism for online crowdsourcing markets. In *Proceedings of the 23rd International Conference on Computer Communication and Networks (ICCCN)*, pages 1–8. IEEE, 2014.

- [98] C. K. Tham and T. Luo. Quality of contributed service and market equilibrium for participatory sensing. *IEEE Transactions on Mobile Computing*, 14(4):829–842, 2015.
- [99] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. Vtrack: Accurate, energy-aware road traffic delay estimation using mobile phones. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 85–98. ACM, 2009.
- [100] J. Wang, J. Tang, D. Yang, E. Wang, and G. Xue. Quality-aware and fine-grained incentive mechanisms for mobile crowdsensing. In *Proceedings of the 36th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 354–363. IEEE, 2016.
- [101] S. Wang, D. Wang, L. Su, L. Kaplan, and T. F. Abdelzaher. Towards cyber-physical systems in social spaces: The data reliability challenge. In *Proceedings of the 35th IEEE Real-Time Systems Symposium (RTSS)*, pages 74–85. IEEE, 2014.
- [102] S. Wang, P. Xu, X. Xu, S. Tang, X. Li, and X. Liu. Toda: Truthful online double auction for spectrum allocation in wireless networks. In *Proceedings of the 6th IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*, pages 1–10. IEEE, 2010.
- [103] W. Wang, L. Ying, and J. Zhang. A game-theoretic approach to quality control for collecting privacy-preserving data. In *Proceedings of the 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 474–479, 2015.
- [104] W. Wang, L. Ying, and J. Zhang. Buying data from privacy-aware individuals: The effect of negative payments. In *Proceedings of the 12th Conference on Web and Internet Economics (WINE)*, 2016.
- [105] W. Wang, L. Ying, and J. Zhang. The value of privacy: Strategic data subjects, incentive mechanisms and fundamental limits. In *Proceedings of the 42nd ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science (SIGMETRICS)*, pages 249–260. ACM, 2016.
- [106] Y. Wei, Y. Zhu, H. Zhu, Q. Zhang, and G. Xue. Truthful online double auctions for dynamic mobile crowdsourcing. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2074–2082. IEEE, 2015.
- [107] Y. Wen, J. Shi, Q. Zhang, X. Tian, Z. Huang, H. Yu, Y. Cheng, and X. Shen. Quality-driven auction-based incentive mechanism for mobile crowd sensing. *IEEE Transactions on Vehicular Technology*, 64(9):4203–4214, 2015.
- [108] C. Wu, Z. Yang, and Y. Liu. Smartphones based crowdsourcing for indoor localization. *IEEE Transactions on Mobile Computing*, 14(2):444–457, 2015.
- [109] D. Xiao. Is privacy compatible with truthfulness? In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science (ITCS)*, pages 67–86. ACM, 2013.
- [110] H. Xie and J. C. Lui. Modeling crowdsourcing systems: Design and analysis of incentive mechanism and rating system. *ACM SIGMETRICS Performance Evaluation Review*, 42(2):52–54, 2014.

- [111] H. Xie, J. C. S. Lui, J. W. Jiang, and W. Chen. Incentive mechanism and protocol design for crowdsourcing systems. In *Proceedings of the 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 140–147, 2014.
- [112] D. Yang, G. Xue, X. Fang, and J. Tang. Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing. In *Proceedings of the 18th Annual ACM International Conference on Mobile Computing and Networking (Mobicom)*, pages 173–184. ACM, 2012.
- [113] D. Yang, G. Xue, X. Fang, and J. Tang. Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones. *IEEE/ACM Transactions on Networking*, 24(3):1732–1744, 2016.
- [114] Q. Yu, J. Chen, Y. Fan, X. Shen, and Y. Sun. Multi-channel assignment in wireless sensor networks: A game theoretic approach. In *Proceedings of the 29th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–9. IEEE, 2010.
- [115] H. Zhang, B. Liu, H. Susanto, G. Xue, and T. Sun. Incentive mechanism for proximity-based mobile crowd service systems. In *Proceedings of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–9. IEEE, 2016.
- [116] Q. Zhang, Y. Wen, X. Tian, X. Gan, and X. Wang. Incentivize crowd labeling under budget constraint. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2812–2820. IEEE, 2015.
- [117] X. Zhang, G. Xue, R. Yu, D. Yang, and J. Tang. Truthful incentive mechanisms for crowdsourcing. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2830–2838. IEEE, 2015.
- [118] X. Zhang, Z. Yang, Z. Zhou, H. Cai, L. Chen, and X. Li. Free market of crowdsourcing: Incentive mechanism design for mobile sensing. *IEEE Transactions on Parallel and Distributed Systems*, 25(12):3190–3200, 2014.
- [119] Y. Zhang and M. van der Schaar. Reputation-based incentive protocols in crowdsourcing applications. In *Proceedings of the 31st Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 2140–2148. IEEE, 2012.
- [120] D. Zhao, X. Y. Li, and H. Ma. How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint. In *Proceedings of the 33rd Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1213–1221. IEEE, 2014.
- [121] D. Zhao, X.-Y. Li, and H. Ma. Budget-feasible online incentive mechanisms for crowdsourcing tasks truthfully. *IEEE/ACM Transactions on Networking*, 24(2):647–661, 2016.
- [122] D. Zhao, H. Ma, and L. Liu. Frugal online incentive mechanisms for mobile crowd sensing. *IEEE Transactions on Vehicular Technology*, PP(99):1–1, 2016.
- [123] Z. Zheng, Y. Peng, F. Wu, S. Tang, and G. Chen. Trading data in the crowd: Profit-driven data acquisition for mobile crowdsensing. *IEEE Journal on Selected Areas in Communications*, PP(99):1–1, 2017.
- [124] Z. Zheng, F. Wu, X. Gao, H. Zhu, G. Chen, and S. Tang. A budget feasible incentive mechanism for weighted coverage maximization in mobile crowdsensing. *IEEE Transactions on Mobile Computing*, PP(99):1–1, 2016.

- [125] Z. Zheng, F. Wu, S. Tang, and G. Chen. Unknown combinatorial auction mechanisms for heterogeneous spectrum redistribution. In *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 3–12. ACM, 2014.
- [126] X. Zhou and H. Zheng. Trust: A general framework for truthful double spectrum auctions. In *Proceedings of the 28th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 999–1007. IEEE, 2009.
- [127] R. Zhu, Z. Li, F. Wu, K. Shin, and G. Chen. Differentially private spectrum auction with approximate revenue maximization. In *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 185–194. ACM, 2014.
- [128] R. Zhu and K. G. Shin. Differentially private and strategy-proof spectrum auction with approximate revenue maximization. In *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 918–926. IEEE, 2015.