



Provided by the author(s) and University of Galway in accordance with publisher policies. Please cite the published version when available.

Title	Adaptive task assignment in spatial crowdsourcing
Author(s)	Hassan, Umair ul
Publication Date	2016-08
Item record	http://hdl.handle.net/10379/6035

Downloaded 2024-04-20T01:35:34Z

Some rights reserved. For more information, please see the item record link above.





NUI Galway
OÉ Gaillimh

ADAPTIVE TASK ASSIGNMENT IN SPATIAL CROWDSOURCING

Umair ul Hassan

*Submitted in fulfillment of the requirements for the degree of
Doctor of Philosophy*

Advisor: **Dr. Edward Curry**

External Examiner: **Prof. Seng W. Loke**

Internal Examiner: **Dr. John Breslin**

The Insight Centre for Data Analytics
National University of Ireland, Galway
August, 2016

Abstract

Spatial crowdsourcing has emerged as a new paradigm for solving difficult problems in the physical world. It engages a large number of human workers in scenarios such as crisis management and smart cities. The utility of spatial crowdsourcing is generally dependent on who performs the task. On one hand, the workers may not perform assigned tasks within time. On the other hand, the performed tasks may not meet the predefined quality criteria. In the case of spatial crowdsourcing, the success of an assignment depends on factors such as task location, travel distance, or expected reward. This necessitates an appropriate task assignment process to optimize the utility of spatial crowdsourcing.

The design of the task assignment process faces three primary research challenges: dynamism, heterogeneity, and uncertainty. The goal of this thesis is to address these research challenges; therefore, it proposes a conceptual framework to analyze the dimensions of dynamic task assignment in spatial crowdsourcing. To formalize the research problem, this thesis defines dynamic task assignment as a repeated decision-making problem under uncertainty and heterogeneity. Uncertainty defines the limited knowledge about the reliability of workers for assigned tasks, and heterogeneity characterizes the differences in reliability and expertise of workers. The proposed formalization is referred to as the adaptive assignment problem in spatial crowdsourcing. The adaptive assignment problem combines online optimization with heuristic-based learning for balancing the exploration-exploitation trade-off during repeated assignments. The problem is instantiated in four different scenarios of spatial crowdsourcing to highlight specific requirements of assignment algorithms. An agent-based simulation methodology is employed to evaluate the algorithms proposed for each scenario. Empirical evaluation provides evidence of the performance of algorithms using synthetic and real-world data.

Declaration

I, Umair ul Hassan, declare that this thesis titled “Adaptive Task Assignment in Spatial Crowdsourcing” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

Dedication

To Ammi, Abu, Azka and Mahd

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor Dr. Edward Curry for the continuous support during my study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor. I am also thankful to Dr. Séan O’Riain for his support as my co-advisor during the early part of my research.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Seng W. Loke and Dr. John Berslin, for their encouragement, insightful comments, and hard questions. My sincere thanks also goes to the members of my graduate research committee: Prof. Dietrich Rebholz-Schuhmann, Prof. Stefan Decker, Dr. Adegboyega Ojo, and Dr. Connor Hayes, for their constructive feedback.

The period I spent in Insight Galway (formerly known as Digital Enterprise Research Institute) was a great experience, and I am happy to be a small part of the history of the research institute. I benefited a lot from the interaction with my colleagues and from their kindness. I was also greatly supported by the administrative and technical staff at Insight Galway. I was also helped by Dr. M. Intizar Ali, Dr. Aqeel H. Kazmi, Dr. Souleiman Hasan, Qaiser Mehmood, and Asra Aslam in proofreading this thesis.

On a personal level, I am grateful to my parents for being a constant source of strength for me throughout my life, as well as for their guidance and sacrifice. I am grateful to my wife Azka who partnered me in all walks of life, whose sacrificial care for me and our son made it possible for me to complete this work. Together, we have a beautiful son Mahd whose smile makes the world look pleasant.

Lastly, and most importantly, I am humbled by the blessings of the Creator and Sustainer of the universe, Allah s.w.t. Indeed, it is He who grants us what we do not deserve and none is worthy of praise except Him.

Table of Contents

Abstract	i
Declaration	iii
Dedication	v
Acknowledgments	vii
List of Figures	xv
List of Tables	xix
1 Introduction	1
1.1 Motivating Example	3
1.1.1 Repeated Interaction & Dynamism	4
1.1.2 Diversity of Tasks & Workers	7
1.1.3 The Uncertainty Challenge	8
1.2 Adaptive Task Assignment in Spatial Crowdsourcing	10
1.2.1 Research Requirements	11
1.2.2 Existing Approaches	13
1.3 Adaptive Task Assignment: A Combinatorial Contextual Bandit Approach	15
1.4 Research Methodology	20
1.5 Thesis Outline	21
1.6 Associated Publications	22
2 Background	25
2.1 Purpose of Crowdsourcing	27

TABLE OF CONTENTS

2.1.1	Human Computation	27
2.1.2	Participatory Sensing	29
2.1.3	Citizen Actuation	30
2.2	Crowdsourcing in the Physical World	31
2.2.1	Spatial Crowdsourcing	32
2.2.2	Mobile Crowdsourcing	32
2.2.3	Mobile Crowd Sensing	33
2.2.4	Other Perspectives	33
2.3	Task Routing: Pull versus Push	34
2.4	Evolution of the Assignment Problem	37
2.4.1	Deterministic & Partially Dynamic	39
2.4.2	Deterministic & Fully Dynamic	40
2.4.3	Probabilistic & Partially Dynamic	40
2.4.4	Probabilistic & Fully Dynamic	40
2.5	Chapter Summary	41
3	Related Literature	43
3.1	Dimensions of Adaptive Task Assignment	44
3.2	Non-spatial Crowdsourcing	46
3.2.1	Online Learning for Adaptive Task Assignment	47
3.2.2	Offline Learning for Adaptive Task Assignment	51
3.2.3	Comparative Analysis	52
3.3	Spatial Crowdsourcing	53
3.3.1	Deterministic Knowledge for Dynamic Task Assignment	54
3.3.2	Probabilistic Knowledge for Dynamic Task Assignment	57
3.3.3	Comparative Analysis	60
3.4	Chapter Summary	61
4	Adaptive Task Assignment in Spatial Crowdsourcing	63
4.1	Preliminaries	64
4.1.1	Assignment Protocol	66
4.1.2	Dynamic Task Assignment	68
4.1.3	The Bandit Problem	72
4.2	Adaptive Task Assignment: A Combinatorial Contextual Bandits Approach	74

4.2.1	The Adaptive Assignment Problem	75
4.2.2	Heuristics based Learning	77
4.3	Evaluation Methodology	79
4.3.1	Agent-based Simulation	79
4.3.2	Data-driven Initialization	82
4.4	Instantiating Adaptive Task Assignment	84
4.5	Chapter Summary	90
5	Adaptive Assignment with Distance-Reliability Optimization	93
5.1	Problem Description	95
5.2	Baseline Approaches	96
5.2.1	Maximum Weighted Bipartite Matching	96
5.2.2	Close Distance Priority	99
5.3	Distance-Reliability Ratio	101
5.4	Estimating Reliabilities	104
5.4.1	Greedy Exploration	104
5.4.2	Optimistic Exploration	106
5.5	Experimental Evaluation	108
5.5.1	Evaluation Methodology	109
5.5.2	Experiments on Real-world Data	113
5.5.3	Experiments on Uniform Data	115
5.5.4	Experiments on Skewed Data	124
5.6	Discussion	129
5.7	Chapter Summary	134
6	Adaptive Assignment with Spatial Contextual Learning	137
6.1	Problem Description	139
6.2	Baseline Approaches	141
6.3	Contextual Bandits	143
6.4	Estimating Reliabilities with Contextual Variables	144
6.5	Experimental Evaluation	148
6.5.1	Evaluation Methodology	148
6.5.2	Experiments without Spatial Context	153
6.5.3	Experiments with Spatial Context	155
6.6	Discussion	156

6.7	Chapter Summary	157
7	Adaptive Semi-Assignment with Location Diversity	159
7.1	Problem Description	161
7.2	Baseline Approaches	163
7.3	Location Diversity	164
7.4	Estimating Reliabilities	166
7.4.1	Thompson Sampling	166
7.4.2	DynTS Algorithm	167
7.5	Experimental Evaluation	170
7.5.1	Evaluation Methodology	171
7.5.2	Experiments on Task Completion	175
7.5.3	Experiments on Dynamic Assignment Size	176
7.5.4	Experiments on Task Completion over Time	178
7.5.5	Experiments on Worker Distribution	179
7.5.6	Experiments on Task Arrival Rate	180
7.6	Discussion	180
7.7	Chapter Summary	182
8	Adaptive Assignment using Spatial Expertise	183
8.1	Problem Description	185
8.2	Baseline Approaches	186
8.3	Estimating Expertise	187
8.4	Greedy Assignment Algorithm	188
8.5	Experimental Evaluation	189
8.5.1	Evaluation Methodology	191
8.5.2	Comparison of Accuracy & Response Rate	196
8.5.3	Comparison of Offline Learning Workload	196
8.5.4	Effects of Filtering on Workload	197
8.6	Discussion	198
8.7	Chapter Summary	200
9	Conclusion	201
9.1	Thesis Summary	201
9.2	Thesis Conclusion	203

TABLE OF CONTENTS

9.3	Core Contributions	205
9.4	Limitations	206
9.5	Future Work	207
	Bibliography	211

TABLE OF CONTENTS

List of Figures

1.1	Overview of thesis research areas.	3
1.2	Example of spatial crowdsourcing tasks in the city maintenance scenario.	4
1.3	A typical workflow of interaction between agents in a crowdsourcing environment.	5
1.4	Evolution of spatial crowdsourcing in a city maintenance scenario over time.	6
1.5	Comparison of the three assumptions for the uncertainty of information about assignment outcomes.	9
2.1	Major themes of crowdsourcing research.	27
2.2	Overview of the reCAPTCHA system for digitizing books using crowdsourcing.	28
2.3	Overview of the EpiCollect system for collecting sensing data from geographical locations using crowdsourcing.	30
2.4	Overview of physical actuation using crowdsourcing.	31
3.1	A conceptual overview of primary aspects of adaptive assignment problem in spatial crowdsourcing environments.	45
4.1	Comparison of the three major assignment protocols.	67
4.2	Three rounds of dynamic task assignment with probabilistic assignment success.	71
4.3	Work flow of the adaptive assignment problem.	75
4.4	Three rounds of adaptive task assignment with estimated worker reliabilities.	77

4.5	An overview of agents in the simulation process and their interactions.	81
5.1	Three rounds of dynamic task assignment with worker reliability and distance variables.	94
5.2	Distribution of check-ins in the New York dataset from Foursquare	109
5.3	Distribution of worker reliabilities in New York dataset	113
5.4	Comparison of algorithms on over multiple rounds of assignment on Foursquare data	114
5.5	Comparison of algorithms on over multiple rounds of assignment on Foursquare data	114
5.6	Effects of task expiry time (Uniform data)	116
5.7	Effects of the range of worker reliability (Uniform Data)	118
5.8	Effects of the number of tasks (Uniform Data)	119
5.9	Effects of the range of the number of workers (Uniform Data)	121
5.10	Effects of the δ parameter on DRR algorithm (Uniform Data).	122
5.11	Effects of task expiry time (Skewed data)	125
5.12	Effects of the range of worker reliability (Skewed Data)	126
5.13	Effects of the number of tasks (Skewed Data)	127
5.14	Effects of the range of the number of workers (Skewed Data)	128
6.1	Three rounds of adaptive task assignment with contextual vectors.	138
6.2	Work flow of the adaptive assignment problem with contextual bandits.	145
6.3	Distributions of users against check-ins in the Gowalla dataset and comparison of distance with check-ins and highlights.	150
6.4	Results of context-free algorithms where workers are simulated as Binomial stochastic processes.	153
6.5	Comparison of algorithms on 26,662 test tasks.	155
7.1	Three rounds of adaptive task assignment with task location.	160
7.2	The heat map of Tokyo dataset based on location diversity of locations.	165
7.3	Workflow of the adaptive assignment problem with the dynamic number of workers per task.	167

7.4	Overview of the interaction between simulated agents	171
7.5	Comparison of algorithm performance in terms of task completion rate.	175
7.6	Comparison of algorithms between fixed and dynamic assignment size.	176
7.7	Relative reduction in assignments for dynamic size policy against fixed size policy.	177
7.8	Comparison of assignment policies over time for the New York Dataset	177
7.9	Comparison of task completion rate under varying distributions of worker model coefficients for the New York dataset.	178
7.10	Comparison of task completion rate under varying distributions of worker model coefficients for the Tokyo dataset.	179
7.11	Comparison of algorithms in terms of task completion rate against mean inter-arrival time of tasks.	180
8.1	Overview of combined assessment approached for estimated worker specific expertise coefficients.	187
8.2	Distribution of the number of tasks against the number of topics in the Movies dataset.	192
8.3	The average level of interest, knowledge, expertise and confidence of workers before and after the experiment for the Actors dataset.	194
8.4	Graphs for the comparison of response rate and accuracy of all heuristics used for Movies and Actors datasets.	195
8.5	Relative workload, as compared to the TA approach, during the offline learning phase.	197

LIST OF FIGURES

List of Tables

1.1	Example of diverse tasks and workers in city maintenance scenario.	7
1.2	An overview of existing literature in terms of the research requirements for adaptive task assignment in spatial crowdsourcing. . .	14
1.3	An overview of the contributions of this thesis in terms of the research requirements for adaptive task assignment in spatial crowdsourcing.	17
2.1	The landscape of crowdsourcing with example platforms.	26
2.2	Overview of the crowdsourcing literature that uses pull task routing method	34
2.3	Overview of the crowdsourcing literature that uses push task routing method	36
2.4	An overview of the matching and assignment problems.	38
3.1	Overview of the existing literature on adaptive task assignment in non-spatial crowdsourcing.	46
3.2	Overview of the existing literature on dynamic task assignment in spatial crowdsourcing.	53
4.1	Datasets used for simulation-based evaluation in spatial crowdsourcing literature.	83
4.2	An overview of the approaches used to address the research requirements of adaptive task assignment in spatial crowdsourcing.	85
4.3	Datasets used for initialization of simulations in following chapters.	90

LIST OF TABLES

4.4	Simulation variables used during experimental evaluation.	91
5.1	Overview of dataset used for data-driven initialization	110
5.2	Experiment settings used for experimental evaluation	112
5.3	Comparison of algorithms in terms of execution time (seconds) on Uniform dataset.	122
5.4	Comparison of algorithms in terms of execution time (seconds) on Uniform dataset.	123
6.1	Overview of dataset used for data-driven initialization	149
6.2	Experiment settings used for experimental evaluation of Spa- tialUCB algorithm.	152
6.3	Comparison of algorithms for 5000 tuning tasks.	154
7.1	Overview of baseline and proposed algorithms	169
7.2	Overview of dataset used for data-driven initialization	172
7.3	Experiment settings used for experimental evaluation	174
8.1	Example of vectors for contextual variables and worker expertise in the city maintenance scenario of spatial crowdsourcing.	184
8.2	Characteristics of the Movies and Actors datasets describing entities, movies, and actors in DBpedia.	192
8.3	Details of workers and tasks for the Movies and the Actors datasets.	193
8.4	Relative performance of assessment approaches in comparison to the CA assessment approach.	198

LIST OF ACRONYMS

Chapter 1

Introduction

“Machines that fit the human environment, instead of forcing humans to enter theirs, will make using a computer as refreshing as taking a walk in the woods.”

The Computer for the 21st Century

MARK D. WEISER

Mark Wieser famously coined the term *ubiquitous computing* [1]. Ubiquitous computing aims to bring computing to the physical world [2]. It embeds computing into the everyday activities of people, serving one or more purposes. It helps people to achieve their goals without the need for their active attention on a device. Miniaturization of computing devices and wearable technologies has enabled ubiquitous computing [3]. Smartphones are the best example of such devices that go beyond their main functionality of communication. Smartphones help people in their daily tasks like navigation, reading, tracking, etc. The computation, sensing, and actuation capabilities of such devices define their usefulness. Now researchers

have started to combine the capabilities of devices and humans to solve difficult and large problems [4–6]. For instance, human capabilities are utilized to perform tasks that are difficult to solve with devices alone.

Employing groups of people to perform difficult and complex tasks was done even before the advent of computing [7]. This was generally achieved by dividing a complex task into small problems that could be solved by an individual person. For example, the path of a comet could be estimated with the help of many human calculators [7]. Recent technological advances have made such calculations much easier through machines. Still, there are problems difficult for computers alone, underlining the need for a symbiosis between humans and devices [8]. The Internet has made it easier to engage a large number of people in complex problem solving [9, 10]. *Crowdsourcing* expands the role of humans from mere users to contributors. Examples of crowdsourcing include Wikipedia [11] and reCAPTCHA [12]. People actively contribute to Wikipedia¹ by writing content about various topics. In the reCAPTCHA² system, people passively digitize documents while answering a set of security questions.

This thesis focuses on a specialization of crowdsourcing known as *spatial crowdsourcing* [13]. Spatial crowdsourcing considers tasks that have strong location specificity. It requires people to visit physical locations to perform the crowdsourced tasks. Examples of spatial crowdsourcing include crisis mapping [14], city maintenance [6], shop audits [15, 16], and personal logistics [17]. A typical spatial crowdsourcing environment consists of three interacting agents: *workers*, *requesters*, and *platform*. The interaction between these agents repeats over time. Each agent is involved in *sequential decision-making* to optimize

¹<http://www.wikipedia.org/>

²<http://www.google.com/recaptcha>

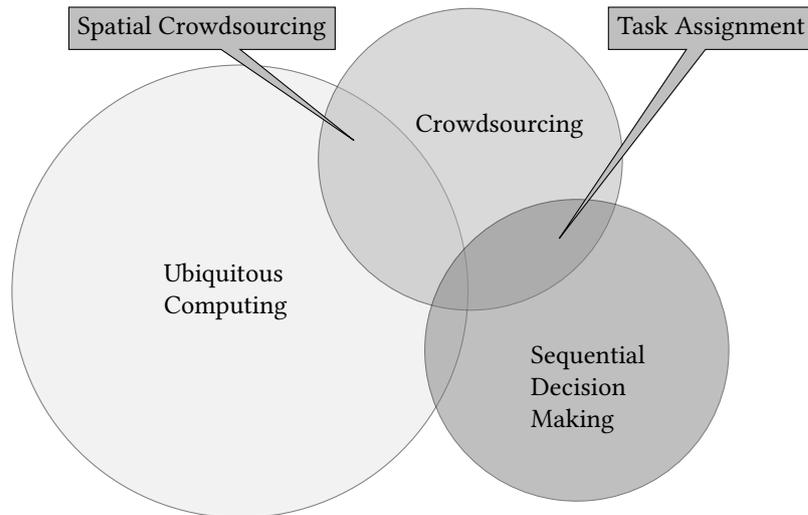


Figure 1.1: Overview of thesis research areas.

its utility. Sequential decision-making includes fundamental decision problems faced by an intelligent agent. The research problem addressed in this thesis involves sequential decision-making in spatial crowdsourcing. Figure 1.1 highlights the overlap of ubiquitous computing, crowdsourcing, and sequential decision-making.

The specific focus of this thesis is *adaptive task assignment* in spatial crowdsourcing. It is the process of assigning the best workers to the tasks while learning from the outcomes of assignment decisions over time. The following sections motivate and outline the main body of work behind this thesis.

1.1 Motivating Example

Consider the scenario of a spatial crowdsourcing system for the maintenance of a city [6, 18, 19]. The system supports the city administration with the help of volunteer citizens. It allows citizens to report problems in their neighborhoods. Potential problems include, but are not limited to, broken street lights, noise pollution, etc. The system distributes reported problems among volunteers who

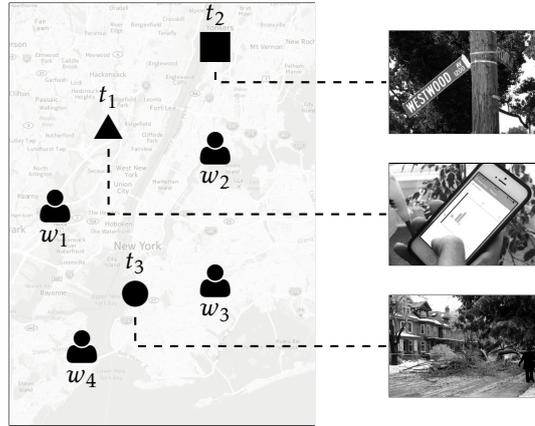


Figure 1.2: Example of spatial crowdsourcing tasks in the city maintenance scenario.

verify and fix them. Figure 1.2 illustrates this scenario on the map of New York city with the help of three example tasks and four volunteer workers³. The tasks are distributed over different locations in the city. Each task requires specific expertise to perform it. Task t_1 requires a worker to record the noise pollution level using their mobile phones. Task t_2 requires a worker to fix a street sign. Task t_3 requires a worker to take detailed photos of a fallen tree in a street to document the damage. Note that workers are also distributed over different locations. A fundamental research question highlighted by this scenario is: *how to assign the best workers to all tasks?* The specific challenges of answering this question are discussed in the following sections.

1.1.1 Repeated Interaction & Dynamism

Spatial crowdsourcing systems consist of the following three types of interacting agents:

- *Requesters* generate tasks for crowd workers. An example of a requester is

³For the sake of clarity, the terms "volunteers" and "workers" will be used interchangeably in the rest of thesis

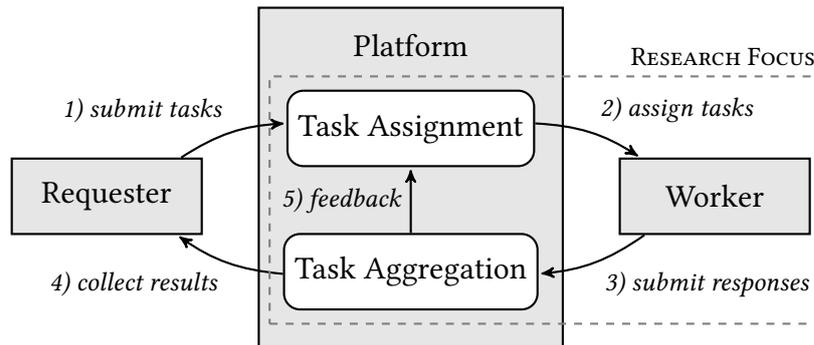


Figure 1.3: A typical workflow of interaction between agents in a crowdsourcing environment.

the citizen reporting problems in the neighborhood.

- *Workers* are the people who are willing to perform tasks. In the city maintenance scenario, workers are the volunteers who verify and fix problems.
- The *platform* is a software agent that mediates between requesters and workers. It defines the interaction mechanism between both agents. It also defines the mode of exchange for tasks, results, feedback, and incentives.

Figure 1.3 highlights the communication paths between these agents. The requester submits tasks to the platform. The platform provides the necessary functionality for assignment of tasks to workers. Appropriate interfaces and algorithms support the assignment process in the platform. Workers perform the assigned tasks and submit the results to the platform. The platform assembles the results according to the requirements of requesters. At the end, requesters are notified about the completion of tasks. In parallel, the platform updates its interfaces and algorithms based on the outcomes of assignments.

The spatial crowdsourcing scenario, as shown in Figure 1.2, illustrates the state of agents at one instance of time. In reality, a multi-agent environment like spatial

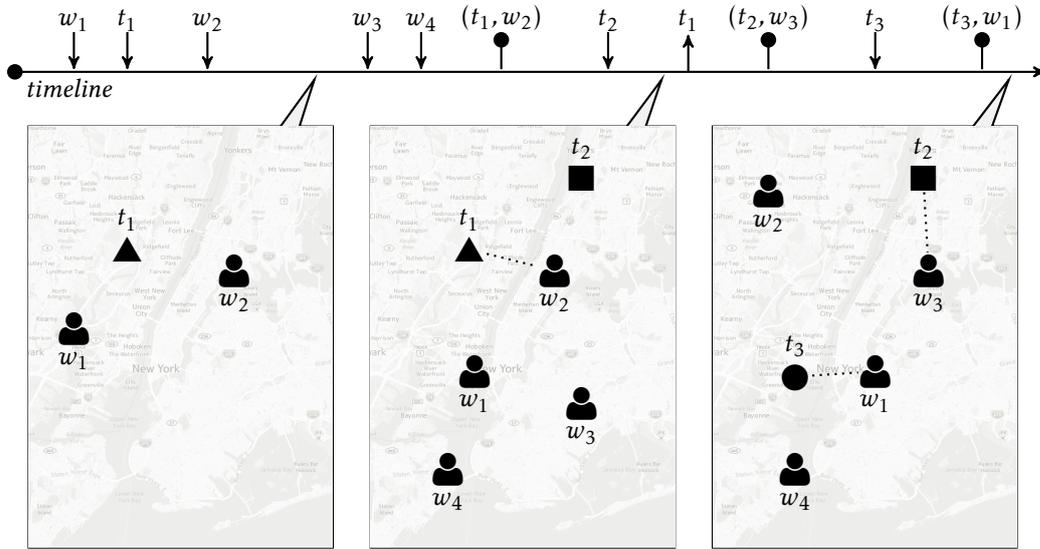


Figure 1.4: Evolution of spatial crowdsourcing in a city maintenance scenario over time.

crowdsourcing is dynamic [20–22]. The dynamism is the result of repeated and irregular interaction between participating agents. As illustrated in Figure 1.4, requesters submit many tasks to the platform over time. Workers may arrive or leave the platform over time. Workers are assigned to tasks which they may or may not perform. Tasks leave the platform either due to completion or expiry.

Due to repeated interaction each agent makes sequential decisions to optimize its own utility [21]. An agent’s utility depends on task assignment, task completion, and combined results. On one hand, a worker can decide whether to perform a task given its value and required effort. On the other hand, a requester can adjust the task deadline to maximize chances of completion. The platform can adjust the task assignment process to discourage malicious behavior. Hence, it is essential to design and implement appropriate algorithms for task assignment [21–23].

Table 1.1: Example of diverse tasks and workers in city maintenance scenario.

Task	Location	Validity	Task Type
t_1	Hasbrouck Heights	2 days	Environment
t_2	Mt Vernon	1 day	Construct
t_3	Ellis Island	1 day	Photo

Worker	Location	Reliability	Expertise
w_1	Kearny	0.65	Environment
w_2	Elliot Ave.	0.90	Construct
w_3	Kings Plaza	0.44	Photo
w_4	Staten Island	0.71	Photo

1.1.2 Diversity of Tasks & Workers

The diversity of tasks and workers is another design challenge of *task assignment* in spatial crowdsourcing. Table 1.1 exemplifies this diversity using the city maintenance scenario of spatial crowdsourcing. The diversity of tasks is not only limited to spatial locations. Tasks also have diverse types and varying deadlines. The heterogeneity of workers stems from the diversity of their reliability, expertise, and mobility. *Reliability* is the success rate of a worker in performing assigned tasks within time and with high-quality. *Expertise* is having high reliability on certain types of tasks. Some worker might be better on some tasks and not so good on other tasks. *Mobility* is the movement pattern of a worker over time. A recent study observed that two factors influence a worker’s decision to perform a spatial crowdsourcing task [16, 24]. These factors are the distance and socio-economic status.

A naive approach to task assignment is to make tasks available through a search and browse interface on the platform. Thereafter, a worker can then visit the platform to choose among the incomplete tasks. This self-determined approach to task assignment is common among existing crowdsourcing platforms. This

approach is also known as *worker selected tasks* (WST) based spatial crowdsourcing [13, 25, 26]. The WST approach requires the active attention of workers for selecting suitable tasks. Hence, it suffers from “search friction” and “task starvation” issues [22, 27]. Search friction is the result of a mismatch between tasks and workers [28, 29]. The inherent design limitations of the interaction mechanism contribute towards search friction. Often, workers have to search many times and filters to find suitable tasks, which is a time-consuming activity. Task starvation occurs when some tasks are never selected by workers [30]. Chon et al. showed that task starvation manifests in lower coverage in spatial crowdsourcing [31]; whereby, tasks at less visited locations are seldom completed.

This thesis addresses the dynamism and diversity issues of task assignment by means of an algorithmic approach. Such an approach is also known as *server assigned tasks* (SAT) based spatial crowdsourcing [13, 32, 33]. The platform implements a dynamic assignment process for dynamic arrivals of tasks and workers. In such a process, assignments are made at the arrival of a task; arrival of a worker; or after regular intervals [34]. The quality of an assignment depends on the uncertainty of information available at the time of decision. The formalization of the dynamic assignment process depends on the assumptions about uncertainty. Contextual information about tasks and workers further helps improve the assignment algorithms.

1.1.3 The Uncertainty Challenge

Dynamic task assignment is generally formulated under different assumptions of knowledge about assignment uncertainty. The uncertainty relates to the knowledge about the outcome of an assignment. The following list summarizes each of these assumptions, as illustrated in Figure 1.5:

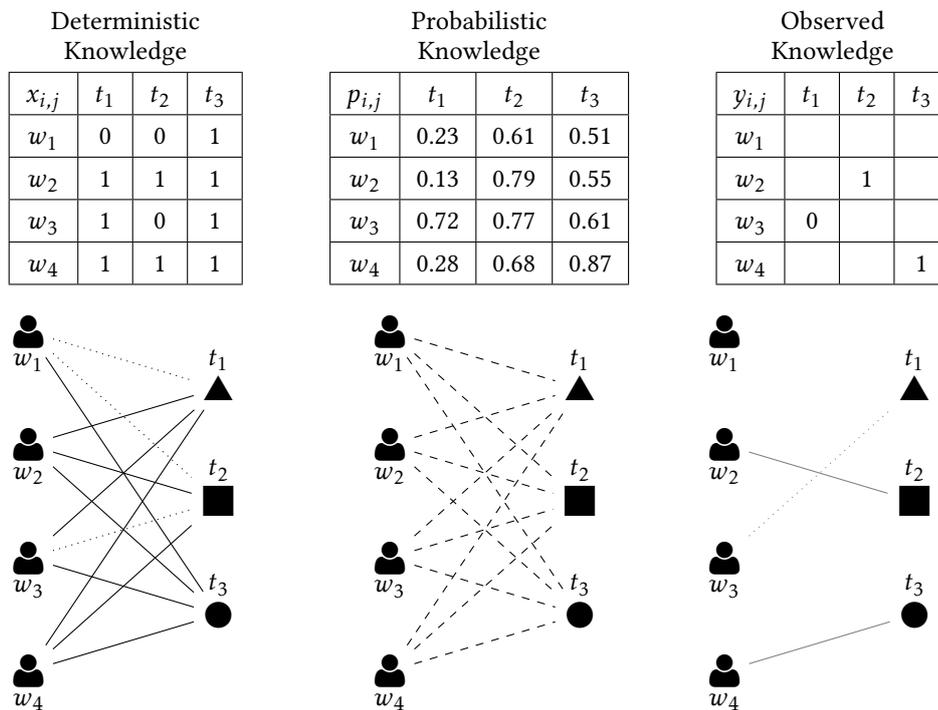


Figure 1.5: Comparison of the three assumptions for the uncertainty of information about assignment outcomes.

- *Deterministic knowledge* assumption considers the availability of deterministic information about the outcomes of assignments [4, 35]. Given that constraints are satisfied, each allowable assignment (i.e. $x_{i,j}$) is assumed to result in a successful outcome. The assignment algorithms aim to maximize the number of allowable assignments. The majority of existing literature on spatial crowdsourcing considers the deterministic knowledge assumption [13, 25, 36].
- *Probabilistic knowledge* assumption considers the likelihood of success (i.e. $p_{i,j}$) for each assignment. The aim of assignment algorithms is to maximize the success probabilities of chosen assignments. Recent research proposals have considered probabilistic knowledge for privacy preservation [37] and trajectory awareness [26, 38].

- *Observed knowledge* assumption considers neither deterministic nor probabilistic knowledge. Instead, only the outcomes (i.e. $y_{i,j}$) for chosen assignments are observed. The goal of assignment algorithms is to approximate the success probabilities for assignments. This is achieved by learning from observed outcomes of the previous assignments.

This thesis considers the observed knowledge assumption for dynamic task assignment. Given the challenges discussed in this section, the fundamental research question is restated as:

Assuming only observed knowledge on the outcomes of assignments, how can the most reliable workers be assigned to dynamically arriving heterogeneous tasks; while exploiting the spatial and non-spatial context of both tasks and workers?

1.2 Adaptive Task Assignment in Spatial Crowdsourcing

This thesis focuses on adaptive task assignment in spatial crowdsourcing. Adaptive task assignment aims to address dynamism, heterogeneity, and uncertainty challenges. It involves approximating assignment probabilities based on the learning from previously observed assignment outcomes. The learning enables adaptive adjustment of the assignment policy for optimizing spatial crowdsourcing over time. The design of appropriate algorithms depends on four primary dimensions of adaptive task assignment:

- The first dimension deals with the definition of an objective function for optimizing spatial crowdsourcing. An objective function quantifies the

utility of assignment decisions for spatial crowdsourcing. For instance, maximization of the task completion rate can be the primary optimization objective. Additionally, minimization of average travel costs can be the secondary optimization objective.

- The second dimension relates to the design of learning methods for approximation of assignment probabilities. The probabilities quantify the uncertainty of the reliability and expertise of workers. A learning method defines the knowledge structure and its update process, based on observed outcomes. It also formalizes the relationship between probabilities and context of tasks and workers.
- The third dimension concerns the context of tasks and workers. In spatial crowdsourcing, the context of tasks and workers include spatial and temporal information. Exploitation of this information distinguishes the design of assignment algorithms in spatial crowdsourcing. Contextual information is used for improving the optimization and learning aspects of adaptive task assignment.
- The fourth dimension covers the constraints imposed on assignment combinations of tasks and workers. Examples of constraints include the number of tasks per worker, the spatial region for a worker, and the expiry time of a task.

1.2.1 Research Requirements

Given the dimensions, a set of research requirements were identified for the design of appropriate algorithms in spatial crowdsourcing [21, 22, 39]. To highlight the

gap in existing research literature, these requirements are categorized into *design* and *performance* requirements. The design requirements that the algorithms should address are as follows:

1. ***Explore-exploit trade-off***: A design requirement of the adaptive task assignment is to solve the dilemma of learning versus optimization. It is also known as the explore-exploit trade-off in the research literature [40]. An algorithm could choose assignments that seem to optimize the optimization objectives. Such assignments might be suboptimal in the long run. As an alternative, the algorithm might choose assignments for the purpose of learning. In this case, suboptimal assignments enable learning from observed outcomes. This exploration might have an adverse effect on the optimization in the short term. But the new knowledge can help improve assignment choices in the long term. A good assignment algorithm aims to strike the right balance between learning and optimization [4, 21, 22].
2. ***Spatial context***: The spatial nature of tasks differentiates spatial crowdsourcing from traditional web-based crowdsourcing [41]. Hence, the spatial context of tasks and workers plays a crucial role in the outcomes of the assignment decisions. Consideration of the spatial context becomes a design requirement, from learning and optimization perspectives [13, 34, 38].
3. ***Multi-criteria optimization***: In spatial crowdsourcing, the optimization objectives might include one or more criteria. For instance, the motivating scenario discussed earlier necessitates optimization of two criteria. First the maximization of task completion rate and second the minimization of travel costs [25, 33, 34]. This highlights the need for adaptive task assignment with multi-criteria optimization.

4. **Contextual learning:** The outcome of an assignment depends on spatial and non-spatial contextual factors [22, 24, 32]. Contextualization of the learning process becomes a design requirement. It helps in improving the adaptive task assignment based on the relationship between context and outcome.

The assignment algorithms should also perform well under varying conditions. The research requirements for algorithm performance are as follows:

5. **Empirical performance:** Empirical evidence of performance should support the design of assignment algorithms. It establishes the claimed utility of the proposed algorithms. A comparative evaluation of the proposed algorithms against baseline and competing algorithms, using well-defined metrics, is necessary [4, 21, 22].
6. **Dimensional scalability:** Evidence of the scalability of the proposed algorithms across primary variables is also required [22, 42, 43]. Primary variables include the number of tasks and the number of workers.

Theoretical analysis entails computational complexity and competitive analysis of adaptive assignment algorithms. Such analysis is a known difficult problem that is out of scope for this research work [40, 44–46].

1.2.2 Existing Approaches

Table 1.2 highlights the research gap in existing literature for the research requirements. Clearly, the majority of works do not address the four design requirements identified earlier. Only three proposals have managed to address more than one requirement so far. Within spatial crowdsourcing proposals, none

Table 1.2: An overview of existing literature in terms of the research requirements for adaptive task assignment in spatial crowdsourcing.

Existing Literature	Explore-exploit trade-off	Spatial context	Multi-criteria optimization	Contextual learning
Expert Crowdsourcing [47, 48]	Budgeted Bandits			
Regression Crowdsourcing [49]	Regression Bandits			
Survey Crowdsourcing [50]	Bandit Survey			
Crowdsourced Classification [51, 52]	Offline Learning			Task Expertise
Spatial Crowdsourcing [13, 25, 36]		Location	Cardinality, Distance	
Diversity based Spatial Crowdsourcing [38]		Location, Direction	Reliability, Diversity	
Trajectory-aware Mobile Crowdsourcing [26, 53]		Location, Trajectory		

has addressed the explore-exploit trade-off and contextual learning requirements. This highlights the gap in existing research on adaptive task assignment in spatial crowdsourcing.

Kazemi and Shahabi first considered deterministic knowledge assumption for dynamic task assignment in spatial crowdsourcing [13]. Their proposed formalization aims to maximize the number of task assignments. They also considered spatial and capacity constraints imposed by workers. Deng et al. reformulated the same optimization objectives with travel costs and time constraints [36]. To et al. considered expertise maximization as an optimization objective [25]. To et al. first considered the probabilistic knowledge assumption to maximize the task completion rate [37]. In addition, they proposed a privacy-preserving framework to address the assignment problem. Recent proposals, under probabilistic knowledge, include diversity-based spatial crowdsourcing [38]

and trajectory-aware spatial crowdsourcing [26]. Notably, none of the existing proposals address the research requirements of adaptive task assignment.

Existing research on non-spatial crowdsourcing has attempted to address the explore-exploit trade-off. The research proposals span several different perspectives. The majority of proposals only consider dynamism of workers [21, 51, 52, 54]. Each dynamically arriving worker is assigned tasks from a fixed pool of tasks. The majority of research proposals aim to maximize the correctness of worker submitted responses. A two-phased approach is proposed to address the explore-exploit trade-off. The first learning phase involves observing the accuracy of worker responses to tasks with known responses. The second phase employs this learning to improve the accuracy of actual tasks with unknown responses.

Recent research proposals provide an alternative formulation based on the multi-armed bandit problem [47, 48, 50]. This formulation addresses the explore-exploit trade-off through online learning heuristics. Otherwise, no other design requirement is satisfied by these proposals. Furthermore, extending these proposals to spatial crowdsourcing is non-trivial. The remainder of this chapter discusses the key contributions of this thesis, in terms of the research requirements.

1.3 Adaptive Task Assignment: A Combinatorial Contextual Bandit Approach

This thesis proposes the novel *adaptive assignment problem* (AAP) which follows a server assigned tasks approach. The AAP provides a generalized framework for formalizing the adaptive task assignment in spatial crowdsourcing. It is not limited to specific types of tasks and considers dynamism of both tasks and workers. The

AAP is formulated as a sequential decision-making problem. It formulates multi-criteria optimization with contextual learning in adaptive task assignment. The learning process utilizes spatial and non-spatial contextual information and the outcomes of assignment decisions.

Existing literature provides various abstractions for formulating sequential decision-making problems. Each formulation depends on the uncertainty of information available at the time of assignment [40, 55, 56]. The *bandit problem* is one such abstraction [57]. It provides a general framework for studying sequential decision-making under partially observable outcomes. It models the explore-exploit trade-off faced by a player who is repeatedly playing a multi-armed slot machine. Each arm generates a stochastic reward on the play. The goal of the player is to maximize the cumulative reward after several rounds of play. The uncertainty in the bandit problem arises because the player has no knowledge on the expectation of reward for each arm. Hence, the player must estimate the expectation of reward based on the previous rounds of play. This equates to the observed knowledge assumption in adaptive task assignment.

The adaptive assignment problem also requires the assignment algorithms to balance the explore-exploit trade-off. Compared to the bandit problem, the adaptive assignment problem requires combinational assignment decisions. So, this thesis formulates the adaptive assignment problem as a combinatorial bandits problem. The proposed formulation enables heuristics-based learning and combinatorial optimization for intelligent assignments decisions. Heuristics-based learning controls the explore-exploit trade-off to approximate the probability of assignment success. The combinatorial optimization controls the selection of best assignments based on the approximated probabilities. The core hypothesis of this thesis is stated as follows:

Table 1.3: An overview of the contributions of this thesis in terms of the research requirements for adaptive task assignment in spatial crowdsourcing.

Chapter	Explore-exploit trade-off	Spatial context	Multi-criteria optimization	Contextual learning
Chapter 4	AAP	AAP	AAP	AAP
Chapter 5	DRR-GRD, DRR-UCB	DRR	DRR	
Chapter 6	SpatialUCB	SpatialUCB		SpatialUCB
Chapter 7	DynTS	DynTS	DynTS	
Chapter 8	WS-GRD			WS-GRD

By formalizing the adaptive task assignment according to the combinatorial bandits framework and enabling contextual learning of reliability and expertise of workers according to spatial and non-spatial contextual information, effective assignment algorithms can be designed for multi-criteria optimization of spatial crowdsourcing.

This thesis instantiates the AAP in four different scenarios of spatial crowdsourcing. A set of algorithms are proposed for each scenario. These algorithms address the design requirements of adaptive task assignment in spatial crowdsourcing, as highlighted in Table 1.3. Each value in the table represents a formal model, or an algorithm, that addresses the research requirement. The specifics of the proposed algorithms are as follows:

- **Explore-exploit trade-off:** The proposed algorithms use heuristics-based learning to balance the explore-exploit trade-off [33, 34]. Chapter 5 introduces an *interval estimation* heuristic to approximate the probability of assignment success. An assignment algorithm based on interval estimation is proposed. It is evaluated against an assignment algorithm with *semi-uniform* heuristic and a deterministic assignment algorithm. The adaptive algorithms achieve a high task completion rate even with approximated

probabilities. An interval estimation based algorithm achieves the best results in general. Chapter 7 proposes a *probability matching* heuristic to approximate the probability of assignment success. This heuristic also achieves high task completion rate compared to naive and non-adaptive algorithms.

- **Spatial context:** The proposed algorithms exploit the spatial context to adapt and optimize assignment decisions. Chapter 5 considers minimization of the distance between tasks and workers to find local workers for tasks [33]. Chapter 6 employs contextual learning to approximate the probability of assignment success [32]. The relationship between distance and task completion is used for the contextual learning. Chapter 7 proposes the location diversity heuristic to estimate the number of workers required for a task [33]. The proposed heuristic reduces unnecessary assignments for tasks at popular locations [24, 58].
- **Multi-criteria optimization:** In spatial crowdsourcing, bi-objective optimization is desirable to find reliable and local workers. Chapter 5 proposes a novel approach to jointly maximize the task completion rate and minimize average travel costs [34]. The proposed approach employs *combinatorial fractional programming* for multi-criteria optimization. This approach achieves 40% less average travel costs when compared to competing approaches. It also performs well on execution time. Chapter 7 explores the optimization of the number of workers assigned to a task while maximizing the task completion rate [33]. This approach reduces assignments by more than 15% without sacrificing the task completion rate.
- **Contextual learning:** Chapter 4 uses linear regression to learn the

relationship between assignment success and contextual attributes of tasks and workers [32]. The proposed approach significantly improves over non-contextual approaches. Chapter 8 considers the scenario when a worker requires specific expertise to perform tasks. Chapter 8 also introduces a warm-start approach to estimate worker expertise [59, 60]. The proposed approach combines workers' *self-stated* expertise with the *estimated* expertise to guide assignment decisions.

Although recent publications provide some initial results, competitive analysis of fully dynamic matching algorithms is still a hard problem [61, 62]. Theoretical analysis of adaptive assignment algorithms is further complicated due to the learning aspect. Previous results have also shown that the empirical performance of algorithms for the bandit problem depends on the problem domain [63]. Nonetheless, recent literature provides the performance bounds of learning heuristics employed in this thesis [46, 64].

Simulation-based evaluation has become the standard practice for both spatial and non-spatial crowdsourcing [13, 26, 37, 38, 52]. This is due to the high cost of deploying dynamic algorithms with real people at large-scale [65, 66]. This thesis takes a principled approach for evaluation of dynamic assignment algorithms in spatial crowdsourcing, and it uses an *agent-based simulation* methodology for this purpose. The simulation environment and agent behavior are parameterized using a *data-driven initialization* approach. During experimentation, both synthetically generated spatial-temporal data and real-world mobility data is used for data-driven initialization.

1.4 Research Methodology

This section describes the research methodology that was followed to investigate the research problem. The following steps are part of the methodology:

1. Comprehensive literature review of research on crowdsourcing and its related topics. Identification of the problem space.
2. Survey of the state-of-the-art on adaptive task assignment in spatial crowdsourcing. Identification of the core research requirements.
3. Classification of existing work according to research requirements, as well as gap analysis.
4. Formalization of the adaptive assignment problem in spatial crowdsourcing according to the combinatorial bandits approach.
5. Definition of an evaluation dataset characterized by a set of agents and their mobility patterns.
6. Investigation of learning and optimization heuristics for adaptive task assignment in spatial crowdsourcing.
7. Implementation of the proposed algorithms: DRR, DRR-GRD, DRR-UCB, SpatialUCB, DynTS, and DynTS.
8. Investigation of evaluation approaches and metrics, identification of baseline and competing algorithms, experiment design, dataset preparation, offline runs of experiments, and data collection.
9. Results analysis and conclusion.

1.5 Thesis Outline

The thesis is structured as follows:

Chapter 2 – Background: This chapter provides an overview of the emergence of crowdsourcing as an active field of research. It highlights the various perspectives of crowdsourcing in the physical world. It also provides a background discussion of the assignment problem from a theoretical perspective.

Chapter 3 – Related Literature: This chapter provides an overview of research literature on task assignment in crowdsourcing. It provides a conceptual analysis of the research problem. It also discusses the specific techniques and algorithms employed in existing literature.

Chapter 4 – Adaptive Task Assignment in Spatial Crowdsourcing: This chapter introduces the adaptive assignment problem and its application to spatial crowdsourcing. It also summarizes the agent-based simulation methodology used for empirical evaluation in the thesis.

Chapter 5 – Adaptive Assignment with Distance-Reliability Optimization: This chapter instantiates the adaptive assignment problem for *bi-objective optimization* in spatial crowdsourcing. This chapter proposes a combinatorial fractional programming approach for bi-objective optimization. It also proposes two adaptive assignment algorithms based on the semi-uniform and interval estimation heuristics.

Chapter 6 – Adaptive Assignment with Spatial Contextual Learning: This chapter instantiates the adaptive assignment problem for contextualized adaptive

assignment in spatial crowdsourcing. It proposes individualized linear models, for workers, to approximate the probability of assignment success.

Chapter 7 – Adaptive Semi-Assignment with Location Diversity: This chapter instantiates the adaptive assignment problem for semi-assignment in spatial crowdsourcing. It proposes a *location diversity* heuristic to decide the number of workers required for each task. It proposes an algorithm that employs probability matching to approximate probability of assignment success.

Chapter 8 – Adaptive Assignment using Spatial Expertise: This chapter instantiates the adaptive assignment problem for expertise based spatial crowdsourcing. It introduces a combined assessment approach to estimate worker expertise during warm-start.

Chapter 9 – Conclusion: This chapter concludes the thesis with discussion of the primary contributions and limitations. Additionally, it provides a road-map to guide future research.

1.6 Associated Publications

Different aspects of this thesis have been published in the following peer-reviewed publications.

Chapter 5 – Adaptive Assignment with Distance-Reliability Optimization

- Umair Ul Hassan and Edward Curry. Efficient Task Assignment for Spatial Crowdsourcing: A Combinatorial Fractional Optimization

Approach with Semi-bandit Learning. *Expert Systems with Applications*, 58:36–56, 2016

Chapter 6 – Adaptive Assignment with Spatial Contextual Learning

- Umair Ul Hassan and Edward Curry. A Multi-armed Bandit Approach to Online Spatial Task Assignment. In *Proceedings of the 11th IEEE International Conference on Ubiquitous Intelligence and Computing*, pages 212–219. IEEE, 2014 (Best Paper Award)

Chapter 7 – Adaptive Semi-Assignment with Location Diversity

- Umair Ul Hassan and Edward Curry. Flag-Verify-Fix: Adaptive Spatial Crowdsourcing leveraging Location-based Social Networks. In *Proceedings of the 23rd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 79:1–79:4, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3967-4

Chapter 8 – Adaptive Assignment using Spatial Expertise

- Umair Ul Hassan and Edward Curry. A Capability Requirements Approach for Predicting Worker Performance in Crowdsourcing. In *Proceedings of the 11th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 429–437. IEEE, 2013
- Umair Ul Hassan, Sean O’Riain, and Edward Curry. Effects of Expertise Assessment on the Quality of Task Routing in Human Computation. In *Proceedings of the 2nd International Workshop on Social Media for Crowdsourcing and Human Computation*. BCS, 2013

Other relevant publications

- Umair Ul Hassan, Murilo Bassora, Ali H Vahid, Sean O’Riain, and Edward Curry. A Collaborative Approach for Metadata Management for Internet of Things: Linking Micro Tasks with Physical Objects. In *Proceedings of the 9th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 593–598. IEEE, 2013
- Umair Ul Hassan, Sean O’Riain, and Edward Curry. SLUA: Towards Semantic Linking of Users with Actions in Crowdsourcing. In *Proceedings of the International Workshop on Crowdsourcing the Semantic Web*, pages 32–44, 2013
- Umair Ul Hassan, Sean O’Riain, and Edward Curry. Towards Expertise Modelling for Routing Data Cleaning Tasks within a Community of Knowledge Workers. In *Proceedings of the 17th International Conference on Information Quality*, 2012
- Umair Ul Hassan, Sean O’Riain, and Edward Curry. Leveraging Matching Dependencies for Guided User Feedback in Linked Data Applications. In *Proceedings of the 9th International Workshop on Information Integration on the Web*, pages 1–6. ACM Press, 2012

Chapter 2

Background

“The more that you read, the more things you will know. The more that you learn, the more places you’ll go.”

I Can Read With My Eyes Shut!

DR. SEUSS

Using large groups of people to solve difficult problems has been the underlying theme of distributed work for centuries [7]. Jeff Howe popularized the term “crowdsourcing” in his WIRED magazine article [9]. He described crowdsourcing as the process of outsourcing a job, usually performed by a designated agent, to a potentially large group of people through an open call. The exact definition of crowdsourcing is still the subject of much debate among researchers. Crowdsourcing is generally applied in support of computer controlled processes. It has been used in application areas such as clothing design, document editing, data curation, and image classification [22, 71–73]. One of the most well-known

Tasks	Volunteer Workers	Paid Workers
Digital	Wikipedia (wikipedia.org) Eyewire (eyewire.org) StackExchange (stackexchange.com) Zooniverse (zooniverse.org) Crowdcrafting (crowdcrafting.org)	Amazon Mechanical Turk (mturk.com) Samasource (samasource.org) CrowdFlower (crowdfower.com) UpWork (upwork.com) uTest (utest.com)
Physical	Ushahidi (ushahidi.com) CitySourced (citysourced.com) OpenStreetMap (openstreetmap.org) FixMyStreet (fixmystreet.com)	TaskRabbit (taskrabbit.com) FieldAgent (fieldagent.net) Gigwalk (gigwalk.com) Airtasker (airtasker.com)

Table 2.1: The landscape of crowdsourcing with example platforms.

examples of crowdsourcing is Wikipedia¹. It is a large encyclopedia created by thousands of contributors around the world. Amazon Mechanical Turk² has proved to be a success story of paid crowdsourcing. It has facilitated researcher across various fields of science and humanities. Table 2.1 shows the categorization of crowdsourcing systems based on nature of rewards and tasks.

The primary focus of this thesis are crowdsourcing systems in the bottom half of Table 2.1. Still, it is useful to study crowdsourcing systems in general. The variety of applications of crowdsourcing across different domains strongly impacts the design decisions and assumptions made for effective implementation of crowdsourcing systems [21, 22]. This chapter aims a) to present a short overview of the variety of research perspectives on crowdsourcing that have emerged over the years, b) to provide a better understanding of the heterogeneity of crowdsourcing system in the physical world and c) to detail its relation to the dynamic task assignment processes.

Section 2.1 introduces the different purposes of crowdsourcing as seen in existing research literature. This leads towards the discussion of crowdsourcing systems in physical world and subtle differences of spatial crowdsourcing in

¹<http://www.wikipedia.org>

²<http://www.mturk.com>

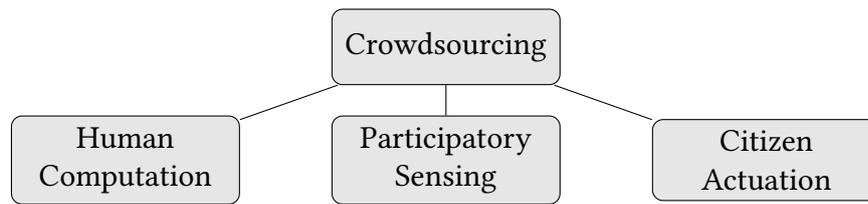


Figure 2.1: Major themes of crowdsourcing research.

Section 2.2. Section 2.3 elaborates the push and pull methods of task routing in crowdsourcing. Section 2.4 provides the theoretical grounding of the work presented in this thesis, by summarizing the evolution of research on assignment problem and bipartite matching.

2.1 Purpose of Crowdsourcing

The general focus here is on the form of crowdsourcing that supports the operation of a computing system [4]. Within this scope in mind, Figure 2.1 highlights some of the recent themes in computing research that are inspired by crowdsourcing.

2.1.1 Human Computation

Algorithms have existed even before the advent of computing machines. Human computers were successfully used to solve complex mathematical problems, for instance estimating the path of a comet [7]. Advances in computing devices have led to a general replacement of human computers with electronic computers, primarily due to their speed, accuracy, and cost. Today researchers are attempting to develop systems that are aimed at replacing error-prone humans in various work settings, as envisioned by Alan Turing [74]. Notwithstanding these efforts, there has been a general realization that not all problems can be solved only through Artificial Intelligence. These so-called computationally difficult problems

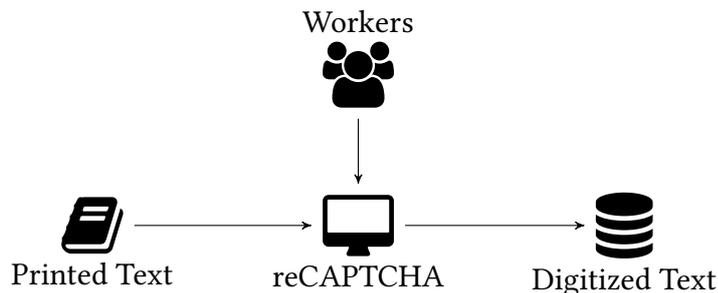


Figure 2.2: Overview of the reCAPTCHA system for digitizing books using crowdsourcing.

have been classified as *AI-Complete* problems [75, 76], similar to the *NP-Complete* problems in the field of Computational Complexity.

The phenomenal developments in network technologies have led researchers towards exploiting both machines and humans for solving *AI-Complete* problems through connectedness of humans with the Web. Recognizing that the machines are better at repeatable tasks at large scale and humans excel at solving subjective problems quickly. Licklider was an early proponent of the idea of mixed human and machine computation as presented in his seminal article titled “Man-Computer Symbiosis” [8]. His proposal was limited to augmenting human intelligence with the computation power of machines. Later on, researchers explored the potential of *human computation* in which the roles are reversed i.e. humans were employed in support of the computational process. Recently, there has been increased interest in using large numbers of Internet users for human computation. The reCAPTCHA system, as shown in Figure 2.2, is one of the most popular examples of human computation being employed for object character recognition on a large scale [4, 77, 78]. Another example of human computation is an algorithm that uses humans to verify machine generated translation between two languages [79].

Crowd-powered computation enables algorithmic access to human affordances in large groups. Specifically, it allows algorithms to work when a feasible solution

by computational means is impossible [4]. In short, the primary objective of human computation is to ask people to perform tasks which may be solved with algorithms but are hard to compute. The focus of human computation is on outsourcing the hardness of computing tasks. Crowdsourcing, in this case, enables access to distributed human intelligence for enabling parallelized and diverse computation. Amazon Mechanical Turk is one of the most popular crowdsourcing platforms that has been regularly used for human computation.

2.1.2 Participatory Sensing

The ubiquitous interaction between humans and computing systems has been fueled by the availability of pervasive devices such as mobiles, wearables, and kiosks. Recognizing its potential, the research proposals have started to exploit the sensing capabilities of devices and human mobility. The idea of *participatory sensing* was proposed to gather information about physical environments through large groups of people [80]. Generally, through an open call, the people are encouraged to provide specific data points mostly in the form in images or measurements. The idea is further extended to mobile phone based sensing with *mobile crowdsourcing*. Crowds of people allow collection large-scale sensory information about our physical world through their mobile phones, where the data collection can be explicit or implicit [41].

Crowd-powered sensing allows collection of data where deployment and maintenance of sensors would have been difficult otherwise. The difficulty might arise due to the scale of the sensing environment, dynamic nature of the environment, or even the requirements of the sensing task. To further illustrate the point, consider the situation where high-resolution pictures of buildings are required to study their specific features. This sensing task is difficult and

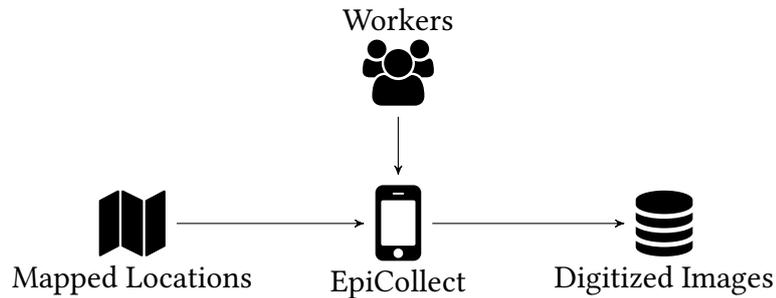


Figure 2.3: Overview of the EpiCollect system for collecting sensing data from geographical locations using crowdsourcing.

prohibitively expensive with computerized sensors; therefore, it makes more sense to involve a set of workers to perform the task. EpiCollect [81] is one such system that allows data collection from geographical locations using mobile-enabled crowdsourcing (see Figure 2.3). In summary, the use of humans for sensing a situation or environment for a purpose-built application is considered a valid use of crowdsourcing. Crowdsourcing, in this case, enables access to distributed human affordances for enabling parallelized and diverse sensing.

2.1.3 Citizen Actuation

The majority of crowdsourcing research has focused on computational and sensing aspects of utilizing humans in support of computing. It is also recognized that more and more applications are also starting to control the state of the environment in response to sensing and computation. *Citizen actuation* uses humans for performing actions in the physical environment when mechanical actuators are costly to deploy or the environment is constantly changing [6, 82, 83]. The applications of citizen actuation are limited so far but are expected to grow. For instance, a recent mobile application allowed city administrators to control the behavior of crowds in large events using direct messaging [84]. Crowdsourcing, in this case,

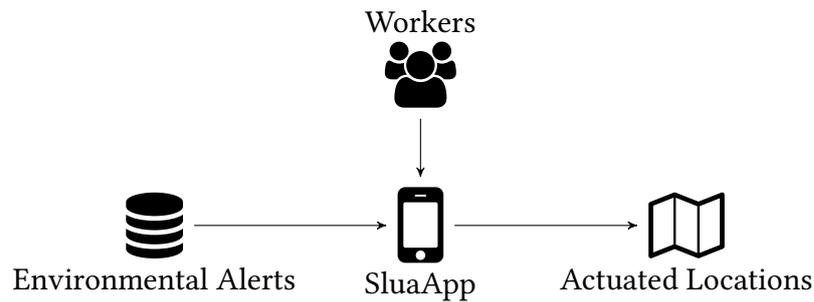


Figure 2.4: Overview of physical actuation using crowdsourcing.

enables access to distributed human motor capabilities for enabling targeted and accurate actuation in a dynamic physical environment (see Figure 2.4). Another application considers the use of human actuators to control smart environment such as intelligent buildings [85].

2.2 Crowdsourcing in the Physical World

In recent years, researchers have started to explore the application and problems at the junction of *ubiquitous computing* and *crowdsourcing*. A significant feature of such form of crowdsourcing is the spatial nature of tasks. The literature in this area focuses on using pervasive and mobile devices to engage workers while being aware of the spatial and non-spatial context of tasks and workers. The underlying intuition is to make tasks easily available to workers without the need for them to constantly sit in front of a desktop or laptop. As a consequence, the spatial mobility of workers is exploited to solve problems that need interaction with the physical world. Different perspectives of physically situated crowdsourcing have emerged over last few years, and each of these perspectives focuses on computation, sensing, or actuation. A non-exhaustive list of major perspectives in literature is provided here to highlight the heterogeneity of tasks and applications.

2.2.1 Spatial Crowdsourcing

Spatial crowdsourcing involves using crowd workers to perform tasks that are associated with physical locations [13, 37]. The majority of research work in this perspective have been lead by the Database and Geographic Information Systems communities [13, 36, 37, 86]. The general focus of this perspective is to collect spatial data; therefore, the motivation scenarios are based on volunteered geographic information. Another example of spatial crowdsourcing is the Ushahidi³—a crowdsourcing platform that was used for mapping disaster hit regions during the 2010 earthquake in Haiti with the help of volunteering workers. A commercial example of spatial crowdsourcing is the TaskRabbit⁴—an online marketplace for crowdsourcing spatial tasks in exchange for small payments. TaskRabbit is a continuously running system where tasks and workers dynamically arrive on the platform over time; on the other hand, the Ushahidi is designed for one-off crowdsourcing campaigns.

2.2.2 Mobile Crowdsourcing

Mobile crowdsourcing is a subclass of spatial crowdsourcing that uses the smartphones carried by people as the primary interaction mechanism with crowds. The main contributions in this perspective have been published in the field of Human-Computer Interaction; therefore, interaction mechanisms and user interface design are the primary focus of research. Typical application scenarios include shop audits, package delivery, etc [20]. Commercial examples of mobile applications based on the concept of mobile crowdsourcing include TaskRabbit

³<http://www.ushahidi.com>

⁴<http://www.taskrabbit.com>

and FieldAgent.

2.2.3 Mobile Crowd Sensing

While spatial crowdsourcing and mobile crowdsourcing focus on directly involving human workers in the computational process, *Mobile crowd sensing* takes a different perspective where the crowd contributions are results of implicit interaction with smartphone carried by people [87–90]. Since modern smartphones are equipped with motion sensors, cameras, and barometers, mobile crowd sensing is aimed at exploiting these sensing capabilities opportunistically to gather data about an environment. The majority of research on mobile crowd sensing is published by researchers in the fields of Mobile Networks and Pervasive Computing.

2.2.4 Other Perspectives

Other related perspectives of spatial crowdsourcing have also been proposed for specific application domains. Ross proposed the concept of *pervasive human computation* while describing the use of human intelligence in the daily routine of people [91]. Zambonelli proposed the application for ubiquitous crowdsourcing in an urban environment, also known as *pervasive urban crowdsourcing* [6]. This discussion on different perspectives of crowdsourcing in the physical world also underlines the variety of formalizations and modeling approaches in the relatively nascent field of crowdsourcing. The focus of this thesis remains spatial crowdsourcing which serves as a generalization of other perspectives.

Table 2.2: Overview of the crowdsourcing literature that uses pull task routing method

Article	Year	Context
Liu et al. [92]	2005	Question Answering
Cosley et al. [93]	2007	Wiki Editing
Guo et al. [94]	2008	Question Answering
Hu et al. [95]	2008	Question Answering
Qu et al. [96]	2009	Question Answering
Zhou et al. [97]	2009	Internet Forums
Yuan et al. [98]	2009	Wiki Editing
Jamjoom et al. [99]	2009	Knowledge Management
Li and King [100]	2010	Question Answering
Kao et al. [101]	2010	Question Answering
Liu et al. [102]	2010	Question Answering
Castro-Herrera [103]	2010	Internet Forums
Horowitz and Kamvar [104]	2010	Social Search
Welinder et al. [105]	2010	Crowdsourced Labeling
Karger et al. [35]	2011	Crowdsourced Labeling
Karger et al. [54]	2011	Crowdsourced Labeling
Dror et al. [106]	2011	Question Answering
Zhu et al. [107]	2011	Question Answering
Ambati et al. [108]	2011	General Crowdsourcing
Riahi et al. [109]	2012	Question Answering
Zhou et al. [110]	2012	Question Answering
Yan and Zhou [111]	2012	Question Answering
Liu and Hao [112]	2012	Question Answering
Zhang et al. [113]	2012	Prediction
Satzger et al. [114]	2013	General Crowdsourcing
Deng et al. [36]	2013	Spatial Crowdsourcing
Goel et al. [115]	2014	General Crowdsourcing
Lin et al. [116]	2014	General Crowdsourcing
Jung [117]	2014	General Crowdsourcing
Yuen et al. [118]	2015	General Crowdsourcing

2.3 Task Routing: Pull versus Push

It has been established that the reliability and expertise of workers do matter for the success of crowdsourcing systems [4, 22, 71]. This is especially true for spatial

tasks and knowledge-intensive tasks. The platform can intervene to ensure the utility of tasks by controlling the assignment process. The process can be designed to ensure match tasks and worker in more intelligent ways. This intervention should lead to the high utility as compared to assigning task to workers randomly. This problem has been referred to as the *task routing* problem [4]. There are two high-level categories of task routing methods: the *push method* and the *pull method*.

The pull method assumes a passive role for the platform which supports worker in their search for tasks. It is assumed that workers are capable of finding right tasks when appropriate searching and browsing tools are made available. In the spatial crowdsourcing literature, this method has been referred to as worker selected tasks [13]. Table 2.2 list the major research articles that employ a pull-based method for task routing. Notably, the majority of articles are addressing crowdsourcing scenarios in question answering systems and generalized crowdsourcing. This is primarily due to the adoption of techniques used in recommendation systems for task recommendation.

The push method is completely controlled algorithmically without leveraging human intelligence. Hence the platform has an active decision-making role and worker are passive receivers of tasks. In the spatial crowdsourcing literature this method has been referred to as *server assigned tasks* (SAT) [13, 25]. Depending on the uncertainty of information about tasks and workers, the push method can be formulated as an optimization problem, a learning problem or a combination of both. Table 2.3 lists the existing research works that use push based method for task routing in crowdsourcing. This method has been mostly used in spatial crowdsourcing and mobile crowd sensing. This primarily due to the passive nature of human interaction in mobile crowd sensing, since the sensing tasks can be

Table 2.3: Overview of the crowdsourcing literature that uses push task routing method

Article	Year	Context
Kazemi [119]	2012	Spatial Crowdsourcing
Ho and Vaughan [51]	2012	Crowdsourced Classification
Tran-Thanh et al. [120]	2013	Crowdsourced Classification
Boutsis and Kalogeraki [121]	2013	Crowdsourced Classification
Chen et al. [122]	2013	Crowdsourced Classification
Difallah et al. [123]	2013	Knowledge Management
Kazemi et al. [124]	2013	Spatial Crowdsourcing
Dang et al. [86]	2013	Spatial Crowdsourcing
Abraham et al. [50]	2013	Crowdsourced Classification
Ho et al. [52]	2013	Crowdsourced Classification
Hassan and Curry [60]	2013	Knowledge Management
Cheng et al. [38]	2015	Spatial Crowdsourcing
To et al. [37]	2014	Spatial Crowdsourcing
Boutsis and Kalogeraki [125]	2014	Crowdsourced Classification
Tarasov et al. [49]	2014	Crowdsourced Regression
Tran-Thanh et al. [126]	2014	Crowdsourced Classification
Tran-Thanh et al. [47]	2014	Expert Crowdsourcing
He et al. [127]	2014	Mobile Crowd Sensing
Zhang et al. [128]	2014	Mobile Crowd Sensing
Hassan and Curry [32]	2014	Spatial Crowdsourcing
Simpson and Roberts [129]	2015	Crowdsourced Classification
To et al. [25]	2015	Spatial Crowdsourcing
Hassan and Curry [33]	2015	Spatial Crowdsourcing
Fan et al. [130]	2015	Knowledge Management
Roy et al. [131]	2015	Knowledge Management
Wang et al. [132]	2015	Mobile Crowd Sensing
Xiong et al. [133]	2015	Mobile Crowd Sensing
Yao et al. [134]	2015	Mobile Crowd Sensing
Xiao et al. [135]	2015	Mobile Crowd Sensing
Cheung et al. [136]	2015	Mobile Crowd Sensing
Miao et al. [137]	2015	Mobile Crowd Sensing
Zhou et al. [138]	2015	Mobile Crowd Sensing
Chen et al. [26]	2015	Spatial Crowdsourcing
Hassan and Curry [34]	2016	Spatial Crowdsourcing

performed by mobile devices without the need of active attention from the user. Note that the research on push method is fairly recent (last 3 years). This thesis focuses the design of intelligent assignment algorithms for the push method in spatial crowdsourcing.

2.4 Evolution of the Assignment Problem

This section provides an introductory overview of the well-known assignment problem. It provides a state-of-the-art review of existing theoretical research on the assignment problem and its dynamic variants. The objectives of this overview are to highlight the state of existing theoretical research and to provide a theoretical grounding for the algorithms presented in this thesis.

Matching and assignment problems are commonly encountered in various real life situations; these problems have been studied extensively since early 20th century [139–141]. Matching problems are often concerned with bipartite graphs. Let $G = (U, V, E)$ be a balanced bipartite graph with two sets of vertices U and V such that $|U| = |V| = n$. The set of edges $E = \{e_{i,j} \mid 1 \leq i, j \leq n\}$ connects each vertex in U with every vertex in V ; therefore, G is a complete bipartite graph. Finding a maximum cardinality bipartite matching is the most fundamental theoretical problem [139]. Votaw and Orden first introduced the classic *assignment problem* (AP) [142] in which each edge $e_{i,j}$ is associated with a cost $c_{i,j}$ and the objective is to find a minimum cost perfect matching in G . Kuhn proposed the famous Hungarian algorithm that solves the AP problem in polynomial time [143]. Since then a series of extensions of the AP have been proposed to address problems under different variables, constraints, and optimization objectives, for instance, semi-assignment and quadratic assignment problems (see [141] for a

Table 2.4: An overview of the matching and assignment problems.

<i>Uncertainty</i>	Partially Dynamic	Fully Dynamic
Deterministic Knowledge	Online bipartite matching [144, 145] Adwords problem [147] Display ads problem [145] Online vertex-weighted bipartite matching [148] Online weighted bipartite matching [149–151]	Two-sided online bipartite matching [146]
Probabilistic Knowledge	Online stochastic matching [61, 62] Online stochastic generalized assignment problem [153]	Dynamic assignment problem [152]

detailed review).

Unlike the classic AP, real-world applications have two important aspects: *uncertainty* and *dynamism* [154]. Uncertainty relates to the information available about the assignment outcomes at the time of assignment decisions. For example, the weights of edges in G might indicate the probability of the successful completion of an assigned task. The dynamism relates to the fact that in some applications the information about vertices or edges changes over time. For instance, in some crowdsourcing applications the tasks might arrive dynamically over time; hence, the set of task vertices is dynamic over time.

The classic assignment problem falls into the deterministic knowledge and *static task assignment* category, where all vertices, edges, and edge weights are available at the time of assignment. The *robust assignment problem* (RAP) concerns solving the assignment problem under uncertainty and sits in the probabilistic knowledge and static task assignment category. In this case, the available information about the outcome of an assignment is probabilistic, and the vertices, edges and edge probabilities do not evolve over time.

Real world systems in general and crowdsourcing systems specifically are

dynamic in nature. Therefore, it becomes necessary to consider dynamic task assignment for the formulation of the assignment process. Table 2.4 categorizes the existing formulations of the assignment processes according to their consideration of dynamism and uncertainty. The assignment process can be either partially or fully dynamic. The uncertainty is defined in terms of the assumption about the outcomes of assignments i.e. deterministic knowledge, probabilistic knowledge, or observed knowledge. The rest of this chapter summarizes the existing formulations of the assignment process for deterministic and probabilistic knowledge assumptions.

For the crowdsourcing perspective, the adaptive assignment processes consider the situation when even the probabilities of assignment success are not available at the time of assignment decisions [21]. Subsequently, the assignment algorithms must learn from the success or failure of previous assignment decisions. The primary challenge of the adaptive assignment is to balance the optimization and learning i.e. how much learning is required to maximize the objective of assignment and when the learning should be scheduled during the execution. This observed knowledge assumption for dynamic assignment process is further discussed in Chapter 3.

2.4.1 Deterministic & Partially Dynamic

The *online bipartite matching* (OBM) considers the situation when the vertices of set U are fixed and the vertices of set V arrive dynamically and the knowledge about the corresponding edges becomes available at the time of vertex arrival [144, 149, 150]. In this case, the problem becomes difficult due to the uncertainty of the sequence of vertex arrivals. The OBM has been studied under various assumptions about the distribution of vertex arrivals. In adversarial arrivals it is assumed that

no information is provided about the time and sequences of vertex arrivals; with stochastic arrivals the vertices are sampled from a known probability distribution. The performance of such online algorithms is studied using competitive analysis against optimal offline algorithms [155].

2.4.2 Deterministic & Fully Dynamic

The *two-sided online bipartite matching* (TOBM) considers the case when vertices of both sets U and V arrive dynamically over time [146]. In this case, the edges for each vertex are revealed upon its arrival. So far, the TOBM has received little attention in terms of theoretical analysis. Recent results have focused on the study of TOBM problem in the context of market clearing applications where both buyers and sellers arrive online [146].

2.4.3 Probabilistic & Partially Dynamic

The *online stochastic matching* (OSM) [61, 62] problem extends the OBM to the situation when the success of a matching is probabilistic. In this case, the vertices of set V arrive dynamically. Upon each arrival, the edges of their neighbors in the set U are revealed as well as their associated probabilities. The *online stochastic generalized assignment problem* (OSGAP) problem is a generalization of the OSM problem where the matching constraints are removed to allow budget constraints [153].

2.4.4 Probabilistic & Fully Dynamic

Dynamic assignment problem (DAP) also considers dynamic arrivals of vertices but with probabilistic knowledge [152]. Therefore, the key difference is that the OBM

considers deterministic knowledge about assignment outcomes and DAP considers probabilistic knowledge. Spivey and Powell formulated the original DAP as a Markov decision process and proposed adaptive algorithms that used the observed history of assignment outcomes [152].

2.5 Chapter Summary

This chapter first focused on highlighting the different purposes of crowdsourcing. Then an analysis of crowdsourcing in the physical world was provided. The analysis motivates the fundamental demands for effective task assignment processes. Specifically due to the heterogeneity and dynamism in terms of spatial and temporal dimensions of crowdsourcing in the physical world. Such demands strongly impact the effectiveness of existing approaches for task assignment. At the center of this challenge is the need to develop algorithms for real-world deployments of spatial crowdsourcing.

The chapter also provided a brief introduction to the assignment problem and its static variants, following a detailed discussion on dynamic variants. This provides an overview of the current state of research literature dealing with theoretical foundations of dynamic task assignment under different levels of uncertainty. There is a lack of categorization to express different requirements of adaptive task assignment.

In fact, the existing research on spatial crowdsourcing is limited to the assumptions of probabilistic knowledge and dynamic arrivals of workers. This chapter introduced another level of uncertainty in spatial crowdsourcing i.e. observed knowledge assumption. This assumption defines the need for learning based algorithms for adaptive task assignment. Next chapter provides an initial

classification of literature on adaptive task assignment in crowdsourcing. This classification helps in understanding the challenges of adaptive task assignment and scope the evaluation methodologies used by existing approaches.

Chapter 3

Related Literature

“As far as the laws of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality.”

Geometry and Experience

ALBERT EINSTEIN

The general focus of this thesis is crowdsourcing in the physical world, where a platform distributes spatially situated tasks to workers using their mobile devices. The spatial nature of tasks and the physical mobility of workers differentiates spatial crowdsourcing systems from the traditional crowdsourcing systems. One of the main challenges of spatial crowdsourcing is the heterogeneity of tasks and workers. On one hand, tasks are associated with different locations which may require certain expertise. On the other hand, workers are generally mobile as well as sometimes unreliable. Matching dynamically arriving tasks with a dynamic pool of workers is a fundamental challenge of spatial crowdsourcing. Due to uncertain

nature of the real-world, practical spatial crowdsourcing systems must make assignment decisions with partial information. The majority of task assignment techniques in crowdsourcing focus on the dynamic task assignment [39] that assumes probabilistic knowledge about assignment decisions. This thesis argues the need to address the adaptive task assignment under observed knowledge assumption [21]. Under this assumption, the assignment algorithm only has access to the outcome of assignment decisions in the form of observed task completion.

Chapter 1 provided a summary discussion on the dimensions of adaptive task assignment in spatial crowdsourcing. This chapter provides an overview of dynamic task assignment and adaptive task assignment in both spatial and non-spatial crowdsourcing. Since the research in non-spatial crowdsourcing is relatively more advanced when compared to spatial crowdsourcing, the literature analysis concentrates on studying existing approaches for adaptive task assignment in non-spatial crowdsourcing. It is worth mentioning that the research area of crowdsourcing is still in its infancy and there is a general lack of unifying theories and algorithms that are applicable to different applications of crowdsourcing. As a consequence, the task assignment approaches are still designed for specific types of applications or crowdsourcing tasks. Section 3.1 discusses the primary dimensions of adaptive task assignment in crowdsourcing. Section 3.2 analyzes the major research works in non-spatial crowdsourcing. Section 3.3 analyzes the major research works in spatial crowdsourcing.

3.1 Dimensions of Adaptive Task Assignment

Section 1.2 introduced the four fundamental dimensions of adaptive task assignment, as illustrated in Figure 3.1. This section revisits these dimensions

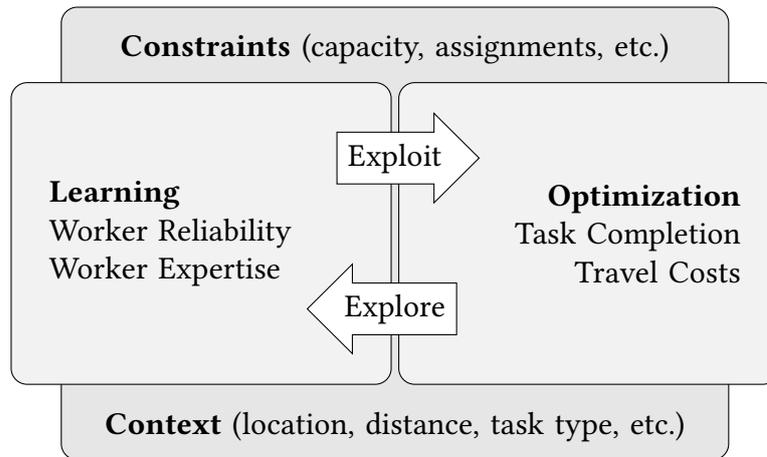


Figure 3.1: A conceptual overview of primary aspects of adaptive assignment problem in spatial crowdsourcing environments.

from spatial crowdsourcing perspective, as evidenced by explicit and implicit discussion in existing literature [21, 22, 39]. The following list briefly discusses each dimension:

- **Optimization Objectives:** The first dimension is concerned with the definition of an optimization objective that quantifies the utility of spatial crowdsourcing. In this regard, the maximization of task completion rate is the primary objective and minimization of average travel costs is the secondary objective.
- **Learned Variables:** The second dimension is related to the design of knowledge structures and learning methods for approximation of assignment success. The design must be driven by the fact that reliability and expertise of workers, in spatial crowdsourcing, are dependent upon the spatial and non-spatial context of corresponding task and worker.
- **Contextual Information:** The third dimension concerns the context of tasks and workers. Unlike web-based crowdsourcing, the context of tasks

Table 3.1: Overview of the existing literature on adaptive task assignment in non-spatial crowdsourcing.

Source	Optimization Objectives	Learned Variables	Context Information	Assignment Constraints
<i>Online learning for adaptive task assignment</i>				
Dekel et al. [156]	Accuracy	Reliability		Task Redundancy
Abraham et al. [50]	Accuracy	Reliability		
Tran-Thanh et al. [47]	Cardinality	Reliability		Incentives Budget
Tarasov et al. [49]	Accuracy	Reliability		
<i>Offline learning for adaptive task assignment</i>				
Ho et al. [52]	Accuracy	Reliability	Task Types	Worker Capacity

and workers also include spatial and temporal information. Exploiting this information for optimization and learning distinguishes the design of assignment algorithms in spatial crowdsourcing.

- **Assignment Constraints:** The fourth dimension covers the constraints specific to tasks or workers. For instance, the number of tasks per worker may be constrained to distribute workload between workers. Conversely, workers may specify the spatial regions they want to be limited to for performing tasks.

3.2 Non-spatial Crowdsourcing

Table 3.1 compares existing literature regarding the primary dimensions of the adaptive task assignment in non-spatial crowdsourcing. The majority of the research work for non-spatial crowdsourcing focuses on using workers for generating true labels for data items in machine learning. The reliability of workers is defined as the accuracy of labels they submit for classification tasks [51], regression tasks [49], or survey tasks [50]. Recent works have proposed

adaptive task assignment techniques based on the estimated reliabilities of workers [49, 50, 52]. These works are differentiated from each other based on the problem formulation, the types of constraints, and learning approach. For the purpose of analysis, these works are categorized according to the learning approaches:

- *Online learning for adaptive task assignment*
- *Offline learning for adaptive task assignment*

In the following, existing research works in each of these categories are further discussed and summarized.

3.2.1 Online Learning for Adaptive Task Assignment

Online learning approaches interweave the learning with assignment decisions [55]. As a result, there is not a clear distinction between exploration and exploitation.

Dekel et al. [156]

Dekel et al. considered the online binary classification with crowd workers [156]. The proposed formulation considers the dynamic arrival of tasks and access to a pool of workers with heterogeneous reliability. They formulated the adaptive task assignment as an extension to the *online selection sampling* problem [157]. The goal is to design algorithms that predict accurate labels for classification instances and learn from the available workers. Each query to a teacher imposes a cost such that overall costs need to be minimized. Furthermore, it is assumed that workers have differing expertise on classification instances. They might provide inaccurate labels to instances outside of their expertise area.

Dekel et al. proposed two selection algorithms for the estimating the expertise of workers and generating accurate predictions. For each task, the first algorithm only decides between choosing all available workers for learning or none at all. The second algorithm selects a subset of workers if needed to improve the predictions. Authors also provided a theoretical analysis of the algorithms in terms of bounds on performance regret and costs of queries to workers. Besides theoretical analysis, a simulation-based methodology was used to evaluate the performance of proposed algorithms. Three baseline algorithms based on different levels of prior information were used for comparative evaluation. The simulations were seeded using real-world dataset for relevance judgment from a commercial search engine. The expertise of a worker was simulated using offline linear classifiers on subsets of original data.

The empirical performance of proposed algorithms was evaluation in terms of the average error rate. The first set of experiments studied the performance of baseline algorithms on four different simulation scenarios. The second set of experiments compared the proposed algorithms against baselines by varying the average number of queries per workers. The results establish the higher performance of the proposed algorithm when the number of workers is relatively small.

Abraham et al. [50]

Abraham et al. proposed the *bandit survey problem* (BSP) that aims to find the correct answers to multiple choice questions [50]. Their proposed formulation addresses both task assignment as well as optimal stopping problem for crowd-sourcing of survey tasks. The proposed problem defines a distribution over the correctness of answers provided by the crowd among available choices. The BSP

is formulated as a variation of the bandit problem. During each time instance, a crowd is chosen for each survey task. It is differentiated from the bandit problem because the outcome of each assignment decisions is the crowd opinion instead of a reward. The optimization objective of the BSP is to trade-off the total cost and error rate.

Abraham et al. provided a theoretical analysis of the BSP using two benchmarks: deterministic and randomized [50]. They propose two crowd selection algorithms with provable guarantees against the deterministic benchmark. Furthermore, it is shown that the randomized benchmark outperforms the deterministic benchmark in special cases. Besides theoretical analysis, a simulation-based methodology was used to evaluate the performance of proposed algorithms. The simulations were initialized using a synthetic dataset.

The empirical performance of proposed algorithms was measured in terms of the average cost and average error rate. The first set of experiments studied the effects of quality threshold on the performance of single crowd stopping rule. The second set of experiments compares the proposed algorithms against a naive approach. The performance comparisons were performed by varying the workloads and quality threshold.

Tran-Thanh et al. [47]

Tran-Thanh et al. proposed the *bounded bandit problem* that considers identification of best workers with a limited set of tasks [47]. The proposed problem was mapped to the expert crowdsourcing scenarios where each task involves complex development activities. The proposed problem extends the bandit problem with budget constraints. More specifically, each round of play is associated with a cost that is limited by the overall budget constraint.

Tran-Thanh et al. proposed an online algorithm for bounded bandit problem based on a semi-uniform heuristic. A theoretical analysis showed sub-linear performance regret of the proposed algorithm. The proposed algorithm was also evaluated against a set of benchmark algorithms. A simulation-based methodology was followed to experiment with different settings of the bounded bandit problem. The simulations were seeded using a real-world dataset gathered from the commercial oDesk platform. The reliability of workers was established based on actual ratings they received on the platform.

The performance of algorithms was compared in terms of the total quality and costs over all tasks. The first set of experiments studied the effects of varying budget constraint on the performance of algorithms. The second set of experiments looked at different sizes of worker pool and their effect on performance. The final set of experiments investigated the effects of the trade-off between quality and cost on the performance.

Tarasov et al. [49]

Tarasov et al. proposed an adaptive task assignment for dynamic estimation of worker reliabilities [49]. Their proposed approach considers dynamic arrivals of workers for a static set of regression tasks. The objective of assignment algorithms is to estimate the worker reliabilities without using gold standard tasks. Their proposed approach maps the reliability estimation problem to the bandit problem. The selection of exploration versus exploitation tasks is controlled through learning heuristics.

Authors proposed two assignment algorithms based on online learning heuristics [49]. The algorithms were evaluated against two baseline algorithms: a naive algorithm using first-come-first-served policy and an optimal algorithm

with access to all assignment outcomes. A simulation based methodology was employed to experiment with the dynamism of workers. The simulations were initialized from regression tasks generated from an audio-visual speech database with responses provided by the crowd. The gold standard responses for tasks were generated manually.

The performance of proposed algorithms was evaluated in terms of the error rate, the costs, and the execution time. The first set of experiments studied the performance of algorithms with constant availability of workers. The second set of experiments compared algorithms when workers have intermittent availability. The third set of experiments considered the effects of varying the number of workers assigned per task.

3.2.2 Offline Learning for Adaptive Task Assignment

Offline learning approaches follow a two-phased approach [55]. The exploration phase estimates the worker reliabilities through test tasks. Usually, the same set of gold standard tasks is used for all new workers. The exploitation phase uses those estimates for making assignment decisions on real tasks.

Ho et al. [52]

Ho et al. considered crowdsourcing for binary classification where tasks are heterogeneous [52]. The proposed formulation considers a pool of tasks and the dynamic arrival of workers. The goal of assignment algorithms is to match a set of tasks with each worker arrival. They formulated the adaptive task assignment problem where the optimization objective is to minimize costs given constraints on the minimum accuracy of classification and the capacity of workers.

Authors employed *online primal-dual* methods to propose an adaptive task

assignment algorithm. Competitive analysis of the proposed algorithm showed that it is competitive with the optimal offline algorithm. A simulation-based methodology was used to establish the empirical performance of the proposed algorithm. The simulations were seeded using a synthetic dataset. The synthetic dataset was generated by varying the reliabilities of a worker on different tasks. The proposed algorithm was compared against a baseline message-passing inference algorithm that follows a non-adaptive task assignment strategy [54].

The performance of algorithms was evaluated in terms of the error rate. The first set of experiments studied the effects of varying the total number of assignments with uniform and diverse workers. The second set of experiments looked at heterogeneity of tasks and its effect on the performance.

3.2.3 Comparative Analysis

Apart from being focused on the spatial crowdsourcing, the work in this thesis is differentiated from these works from three key perspectives.

- The proposed formulation in this thesis considers dynamic arrivals of both tasks and workers; whereas, most of the existing approaches for non-spatial tasks are limited to dynamic arrival for either tasks or workers.
- The goal of assignment algorithms is to optimize the reliability and travel costs. By comparison, above discussed proposals only optimize one objective while imposing constraints on the other objective.
- This thesis also considers the situation where the number of tasks is not fixed in advance; therefore, enabling adaptive task assignment in long-running crowdsourcing system.

Table 3.2: Overview of the existing literature on dynamic task assignment in spatial crowdsourcing.

Source	Optimization Objectives	Contextual Information	Assignment Constraints
<i>Deterministic knowledge for dynamic task assignment</i>			
Kazemi and Shahabi [13]	Assigned Tasks, Travel Costs	Locations	Worker Capacity, Worker Spatial Region
To et al. [25]	Expertise Match, Travel Costs	Locations	Worker Capacity, Worker Spatial Region
Deng et al. [36]	Assigned Tasks	Locations	Worker Spatial Region
<i>Probabilistic knowledge for dynamic task assignment</i>			
To et al. [37]	Reliability	Locations	
Cheng et al. [38]	Reliability, Diversity	Locations, Worker Velocity	
Chen et al. [26]	Reliability	Locations	Travel Time

3.3 Spatial Crowdsourcing

Spatial crowdsourcing entails relatively different assignment approaches due to the spatial nature of tasks and longer durations of time required to perform tasks. Notably, none of the existing approaches address the requirements of adaptive task assignment. Instead, the closely related existing approaches focus on dynamic task assignment as summarized in Table 3.2. These approaches are categorized according to the assumptions of uncertainty:

- *Deterministic knowledge for dynamic task assignment*
- *Probabilistic knowledge for dynamic task assignment*

In the following, prominent research works in each of these categories are further summarized.

3.3.1 Deterministic Knowledge for Dynamic Task

Assignment

The majority of existing research works have considered deterministic knowledge assumption for dynamic task assignment. The commonality among proposed approaches is the use of optimization constraints for the inclusion of worker preferences and task requirements. Most of the proposed approaches are evaluated using an offline simulation methodology. In the following, a set of research works under this category are described that are closely related to this thesis.

Kazemi and Shahabi [13]

Kazemi and Shahabi proposed the *maximum task assignment* (MTA) problem for SAT-based spatial crowdsourcing [13]. The MTA aims to maximize the number of assigned tasks over multiple rounds of assignment. The MTA problem formalizes two constraints for workers: spatial region and task capacity. A dynamic version of the MTA was considered with no knowledge of future task and worker arrivals in a round. The dynamic MTA was reduced to the *maximum network flow* problem. A greedy approach was proposed to solve the dynamic MRA problem using Ford-Fulkerson algorithm [158]. The greedy approach was improved with two spatial heuristics for prioritizing tasks: least location entropy and nearest neighbor.

The scalability and the effects of constraints on the performance of proposed approaches were evaluated. The primary metrics used for performance evaluation were the number of assigned tasks and average travel costs. A set of offline experiments were performed using a simulation based methodology. The simulations were initialized using synthetic and real-world data. The real-world data was extracted from a *location-based social network* i.e. Gowalla. The first

experiment evaluated the scalability by varying the number of tasks over the same period. The second experiment evaluated the effects of worker capacity constraint. The third experiment evaluated performance when the spatial region constraint for workers was increased.

The MTA problem is constructed with some unrealistic assumptions. It does not consider the probabilistic nature of task completion after assignment. The need to specify constraints, such as tasks capacity and spatial region, puts an extra cognitive load on workers. The MTA problem is also limited due to no differentiation between tasks.

To et al. [25]

To et al. proposed the *maximum score assignment* (MSA) problem for SAT-based spatial crowdsourcing [25]. The MSA extends the MTA with worker expertise of different task types. The optimization objective of MSA is to maximize the expertise score for matches made between tasks and workers. The dynamic MSA problem was reduced to the *maximum weight bipartite matching* problem, with the unit capacity assumption for workers. A greedy approach was proposed to solve the dynamic MSA problem using the Hungarian algorithm [143]. The greedy approach was extended with two spatial heuristics to improve global performance: least location entropy and close distance priority.

The performance of proposed approaches was evaluated in terms of three metrics: the average expertise score per round, the average number of assignments per round, and average travel cost. Offline simulation based experiments were performed. The datasets used for initializing the simulation include synthetically generated data and location-based social network data. The first sets of experiments evaluated the scalability of our proposed approaches.

These experiments varied the average number of eligible workers per task and the average number of available tasks per worker. The second set of experiments evaluated the impacts of the workers' constraints on the performance of approaches. The final experiment compared the performance of proposed approach against the well-known Ranking algorithm [144].

The MSA problem is also limited in its applicability due to deterministic knowledge assumption. The close distance priority heuristic showed the overall best results for all metrics. However, this approach may suffer from computational complexity due to the need to solve an integer program in its second phase. The experimental evaluation did not provide a performance comparison of approaches in terms of execution time.

Deng et al. [36]

Deng et al. proposed the *maximum task scheduling* (MTS) for WST-based spatial crowdsourcing [36]. The goal of MTS problem to maximize the number of tasks assigned in a sequence to a worker. Besides the time deadlines for spatial tasks, the MTS assumes spatial region constraints from workers. The MTS problem is shown to be an *NP-Hard* problem. Three approximation algorithms were proposed based on spatiotemporal heuristics: least expiration time heuristics, nearest neighbor heuristic, and most promising heuristic.

The proposed algorithms are evaluated in terms of their scalability and the effects of task expiry times. Three metrics were employed to compare the algorithms: running time, approximation ratios, and a number of completed tasks. This work also used a simulation-based experimental methodology. Both synthetic and real-world datasets were used for initializing the simulations. The MTS problem is limited to the maximization of assigned tasks to one worker. It does

not consider other workers available at the time of assignment. Furthermore, it does not optimize the travel costs for worker which might lead to low completion rate in a real-life situation.

3.3.2 Probabilistic Knowledge for Dynamic Task

Assignment

Recent research works have considered probabilistic knowledge assumption for dynamic task assignment. Most of the proposed approaches model the probability of task completion as a stochastic process. In the following, a set of research works under this category are described that are closely related to this thesis.

To et al. [37]

To et al. defined a privacy enabling framework for the spatial crowdsourcing [37]. They proposed a task assignment approach that aims to maximize the task completion rate. Their proposed framework disseminates tasks to a set of workers, given probability of task being accepted by a worker. The probability of task acceptance is defined in terms of a decreasing function of distance. A greedy algorithm was proposed to geocast a spatial region, containing the set of workers, with a high probability of task completion. A compactness-based heuristic is proposed to improve the greedy algorithm.

The performance of proposed approach was evaluated in terms of three metrics: average acceptance rate, average travel costs, and an average number of notified workers. An offline simulation based methodology was used to evaluate the algorithm using datasets from two location-based social networks: Gowalla and Yelp. The first experiment studied the effects of privacy budget on greedy

algorithm and its variants. The second experiment evaluated the effects of privacy budget against the compactness-based heuristics. The third experiment compared the proposed algorithm against a non-private algorithm.

The primary contribution of this research work is the application of the concept of differential privacy in spatial crowdsourcing. For this purpose, the definition of probability of task acceptance is rather simplified. It is assumed that the probability is purely a function of the distance, independent of the task and worker. Another limitation of this research work is that it does not capture the individualized reliability of a worker. Instead, it is assumed that worker always completes an accepted task.

Cheng et al. [38]

Cheng et al. proposed a framework for *reliable diversity-based spatial crowdsourcing* (RDB-SC) [38]. The proposed problem aims to assign a set of workers to each task such that the reliability and the data diversity is maximized. The task reliability is defined in terms of the probability that at least one of the several assigned workers will complete the task. The task diversity of task in terms of spatiotemporal variation of data submitted by workers. The reliable diversity-based spatial crowdsourcing models the worker according to three variables: the location, the velocity, and the reliability. The worker reliability is defined as the expectation of the worker performing an assigned task.

Cheng et al. showed that the RDB-SC problem is an NP-Hard problem by reduction to the *number partition problem* (NPP). Three approximation algorithms were proposed to address the RDB-SC problem, namely the greedy algorithm, the sampling algorithm, and the divide-and-conquer algorithm. In general, a simulation-based methodology was used to evaluate the performance of the

proposed algorithms. The simulations were initialized using either a synthetic dataset or a real-world dataset. The synthetic dataset was generated using uniform and skewed distributions for over a 2D data space. The real-world dataset contained points of interest in large cities (task locations) and GPS trajectories for taxis (worker trajectories).

The performance of proposed algorithms was compared against two metrics: the minimum reliability and the summation of expected diversity. Experiments on real-world dataset studied the effects of varying the worker reliabilities and task expiry times. The synthetic data was used to evaluate the scalability of proposed algorithms. The experiments on synthetic data included variations in the number of workers, the number of tasks, and the angles of worker velocity. Furthermore, the computational performance of proposed algorithms was compared in terms of the execution time and index construction time.

The algorithms were also evaluated using data from a prototype spatial crowdsourcing platform and real workers. The platform requested workers to submit photos of specific locations. In this case, the reliability of a worker is approximated with a peer rating given by other users. The algorithms were adapted to follow an incremental strategy for updates to task-worker assignments. The algorithms were deployed for 15 minutes or varying intervals update between updates. The performance of algorithms was reported against different durations of update intervals. The design of this experiments is not detailed; therefore, it is not clear how each algorithm was deployed. Furthermore, the size of worker population is rather small (10 paid workers). In general, this research work does not consider dynamism of tasks and the proposed algorithms do not adapt to the observed reliability of workers.

Chen et al. [26, 53]

Chen et al. formulated the *multi-agent task recommendation* problem under stochastic uncertainty of worker trajectories in spatial crowdsourcing [26, 53]. The proposed problem aims to recommend tasks to a set of workers such that the probability of task completion is maximized. The uncertainty of task completion is defined in terms of the probability of a worker following a routine trajectory. Chen et al. also showed that the proposed problem is an NP-Hard by reduction to the *orienteering problem* [26, 53].

A *lagrangian relaxation* based heuristic is proposed to solve the computationally difficult problem. In general, a simulation-based methodology was used to evaluate the performance of proposed approach. The simulations were initialized using either a synthetic dataset or a real-world dataset. The synthetic dataset was generated using random trajectories. The real-world dataset contained trajectories from a public transit network in a large city.

The proposed algorithms were compared in terms of execution time and quality of heuristic against the optimum exact solution. Experiments on synthetic datasets studied the effects of the number of agents, the number of tasks, and the number of locations on the performance of algorithms. The real-world data was used to compare deterministic approaches versus stochastic approaches. The experiments on real-world data compared the proposed approach against a deterministic search based heuristic and a proximity based heuristic.

3.3.3 Comparative Analysis

In comparison, the work presented in this thesis is differentiated from existing spatial crowdsourcing research works in three key aspects.

- Existing works consider known probabilities for assignment; whereas, the algorithms in this thesis approximate the probabilities using online learning heuristics.
- Existing works are limited to optimization of reliability; whereas, this thesis considers the adaptive assignment problem with multi-criteria optimization.
- None of the existing works addresses the adaptive task assignment in spatial crowdsourcing. Hence, the techniques discussed in this thesis are complimentary to existing works.

3.4 Chapter Summary

This chapter presented and analyzed the state-of-the-art for dynamic and adaptive task assignment in crowdsourcing. Different categories of approaches including offline learning and online learning were identified for adaptive task assignment in non-spatial crowdsourcing. The main mechanisms used for dynamic task assignment in spatial crowdsourcing were identified. The analysis supported the identification of the main gaps in literature which include: (i) need for adaptive task assignment approaches in spatial crowdsourcing; (ii) investigation of such approaches for various aspects of spatial crowdsourcing; and (iii) design and evaluation of appropriate algorithms. The contribution of this thesis concentrates on formulating a generalized framework for adaptive task assignment in spatial crowdsourcing and demonstrating its applicability through instantiation of the framework in various spatial contexts.

Chapter 4

Adaptive Task Assignment in Spatial Crowdsourcing

“The purpose of abstracting is not to be vague, but to create a new semantic level in which one can be absolutely precise.”

The Humble Programmer

EDSGER W. DIJKSTRA

There are two primary goals of this thesis. First, formulation of a generalized framework for adaptive task assignment for spatial crowdsourcing based on online learning and optimization. Second, demonstration of the utility of the framework through concrete instantiations under different scenarios of spatial crowdsourcing. This chapter focuses on the first goal and the following chapters on the second goal.

Section 4.1 formally defines the main concepts of adaptive task assignment

in spatial crowdsourcing Section 4.2 formally introduces the adaptive assignment problem. Section 4.3 presents an agent-based simulation methodology for the evaluation of proposed algorithms. Section 4.4 summarizes four instances of the adaptive assignment problem under different contexts and assumptions of spatial crowdsourcing.

4.1 Preliminaries

This section formally defines the underlying concepts needed to formulate the adaptive task assignment in spatial crowdsourcing. The introduced concepts, their definitions, and their associated notations provide the necessary constructs to formulate the assignment problem. The two most fundamental concepts in spatial crowdsourcing are tasks and workers:

Definition 4.1 (Spatial crowdsourcing task). *A spatial crowdsourcing task $t_i \in T$ is defined by a vector of task attributes. At a minimum each task vector has three attributes $(r_i, r_i + \xi, l_i)$.*

The task t_i starts at time r_i and requires a predefined set of actions to be performed at location l_i before it expires at the end of round $r_i + \xi$. These attributes signify that tasks appear dynamically in the spatial crowdsourcing platform and disappear from it dynamically. For example, a task might arrive that requires a worker to take photos of an event in the city. As soon as the task becomes visible in the platform it must be completed before the expiry time; otherwise, the task is considered incomplete which results in lower than desired coverage of the associated event. The attribute l_i specifies the location associated with the event targeted by the task. Note that a spatial task requires a worker to travel physically to the task location l_i to perform it.

Additional task attributes include but are not limited to the types of task and the number of workers required for a task. Examples of task types include but are not limited to shop reviews [20], pollution readings [6], or citizen actuation [82]. The attribute k_i specifies the number of unique workers that must perform a task. Generally, a task requires only one worker. Some scenarios might require more than one worker to perform the task. For instance, a pollution monitoring task might require at least three different readings to ensure the accuracy of the gathered data [37]. Another example is a task that might require multiple workers to take photos of a building from different angles and at different times of the day to maximize diversity [38].

Definition 4.2 (Spatial crowdsourcing worker). *A worker w_j is a person who participates in spatial crowdsourcing by registering with the platform. The worker remains mobile during the crowdsourcing process by moving around at different locations dynamically over time.*

A worker generally consents to perform the assigned tasks either implicitly by registering with the platform or explicitly by requesting tasks. Each worker is characterized by a set of attributes: location, capacity, and spatial region. The location attribute of a worker specifies the last recorded co-ordinates of the worker. Understandably, the location tracking of the worker is an important part of the spatial crowdsourcing that can be achieved through the worker's mobile device. This thesis assumes access to an appropriate location tracking mechanism with the consent of participating workers. The capacity attribute of a worker specifies the maximum number of tasks the worker is willing to accept at a time [13, 25]. In reality, the requirement to specify such constraints may become burdensome for workers due to the additional decision making involved. A worker might also

have the spatial preferences in terms of the maximum distance that the worker is willing to travel. Such preferences might be specified with the spatial region attribute; otherwise, the worker might be assigned tasks distant from her. Kazemi and Shahabi considered the case when workers explicitly state both capacity and spatial constraints along with the task requests [13, 25].

Definition 4.3 (Spatial crowdsourcing platform). *The crowdsourcing platform serves as a middleware that matches task with workers. It provides the necessary interaction mechanisms through appropriate user devices. At any instant in time, the platform maintains two lists: the list of tasks T and the list of workers W .*

Note that, the primary function of the platform is to provide the necessary technology for enabling the exchange of tasks between requesters and workers. To this end, the platform provides necessary interfaces; maintains databases of tasks and workers; and tracks the status of task progress. New arriving tasks and workers are assigned unique identifiers for this purpose. A platform follows a fixed assignment policy for matching incomplete tasks with available workers; it must ensure that the assignment policy adheres to the constraints specific to tasks and workers. The specifics of the assignment policy are determined by the assignment protocol.

4.1.1 Assignment Protocol

The timing of assignment decisions plays an important role in the formulation of the dynamic assignment problem and the design of appropriate algorithms. In particular, the assignment protocol implemented by a spatial crowdsourcing platform depends on the requirements of task completion and the workers' expectations of task provisioning. Figure 4.1 illustrate the three major assignment

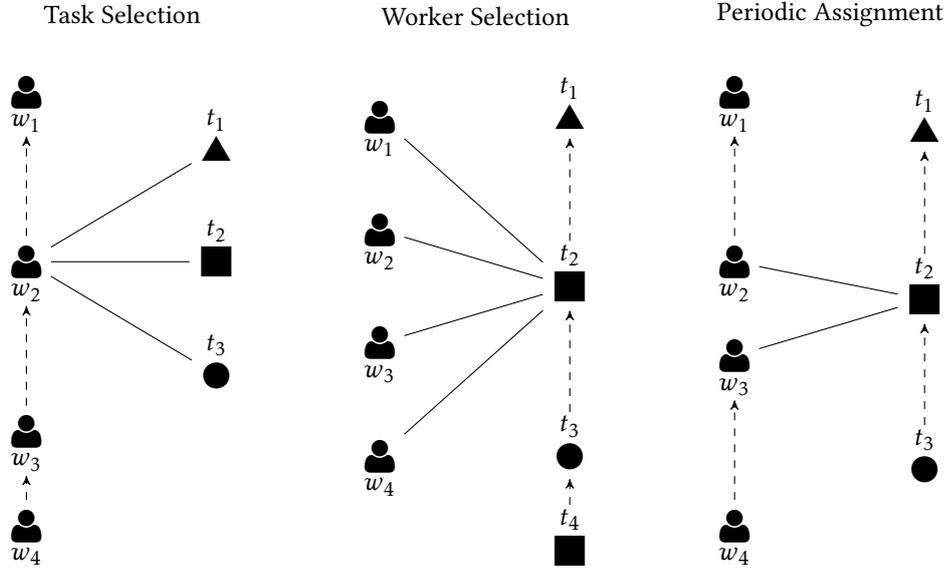


Figure 4.1: Comparison of the three major assignment protocols.

protocols, as described below:

- In *task-selection protocol* tasks are assigned to dynamically arriving workers expect a set of tasks upon request. The majority of existing research works on both spatial crowdsourcing and non-spatial crowdsourcing assume task-selection protocol. A pool of tasks is assumed to be available and algorithms are designed to select a set of tasks for each dynamically arriving worker [49, 51, 52, 54, 130].
- The *worker-selection protocol* assumes that the tasks dynamically arrive on the platform. The platform then chooses a set of workers for each task from the pool of available workers [32, 33, 37, 47]. Dynamic task arrivals are more common in real-time crowdsourcing scenarios [125].
- The *periodic-assignment protocol* assumes that both the tasks and worker arrive dynamically over time. The assignment decisions are performed on discrete time instances after regular intervals. The periodic-assignment

protocol is most commonly considered in spatial crowdsourcing [13, 25, 34]. Note that the algorithms developed for the periodic-assignment protocol can be easily adapted to task-selection protocol and worker-selection protocol.

Given an assignment protocol, the interaction between requesters, platform, and workers is a dynamic process with repeated decision making. Understandably, the assignment decisions are not once-off as tasks and workers arrive dynamically on the platform. The assignment process proceeds in rounds where each round involves matching a set of tasks with a set of workers. Next section formulates the dynamic task assignment process in spatial crowdsourcing based on the periodic-assignment protocol.

4.1.2 Dynamic Task Assignment

Let $r \in \{1, 2, \dots, R\}$ be the rounds of assignment for a spatial crowdsourcing platform. The duration of a round is generally based on fixed time in the periodic-assignment protocol. Whereas, the time between two consecutive arrivals is considered as the duration of a round in other two protocols. At the start of a round r , let $W_r \subset W$ be the set of unassigned and available workers and $T_r \subset T$ be the set of unassigned and incomplete tasks. A task is added to the set T_r as it becomes available for workers and removed after completion or expiry. A worker joins the set W_r on becoming available and leaves the set when busy. Depending on the interaction mechanism, workers might indicate availability by explicitly sending task requests to the platform or they are assumed to be available as soon as they complete a task.

Let $G(W_r, T_r, E_r)$ denote a dynamically changing bipartite graph. The two sets of vertices W_r and T_r are disjoint sets of size m and n , respectively. The set $E_r =$

$\{ \langle t_i, w_j \rangle \mid t_i \in T_r, w_j \in W_r \}$ contains edges that connect vertices W_r with vertices in T_r . Vertices and edges are dynamically added and removed from G as tasks and workers become available or unavailable. Let the binary variable $x_{i,j}$ be 1 when task $t_i \in T_r$ is assigned to the worker $w_j \in W_r$ and 0 otherwise. The assignment algorithm chooses an assignment in G , at the start of each round r .

Definition 4.4 (Assignment Matrix). *The binary matrix $[x_{i,j}]^{n \times m}$ defines an assignment of tasks and workers in round r .*

The assignment matrix $[x_{i,j}]^{n \times m}$ must satisfy the constraints imposed on tasks and workers. One type of constraint might require multiple workers to be assigned to a task, we assume that such a constraint can be satisfied by creating multiple instances of the same task. These simplifying assumptions ensure that the problem of finding an assignment set is equivalent to finding a bipartite matching. The static version of the problem can be mapped to the well-known assignment problem [143]. A new worker is matched to a task in each round until the task is removed from the set T_r , due to successful completion or expiry. Note that a list of previous assignments is maintained for a task to prevent the same worker being assigned to the same task repeatedly. The design goal for an assignment algorithm is to find the best assignment strategy for all rounds. The best strategies are characterized by the optimization of the utility objectives of the spatial crowdsourcing.

If $y_{i,j}$ is the binary variable that is 1 when w_j performs the task t_i with before the end of the current round and 0 otherwise, then the expectation of $y_{i,j}$ is defined as

Definition 4.5 (Worker Reliability). *Each worker w_j assigned to task t_i has an*

associated probability of success $p_{i,j}$ such that

$$y_{i,j} = \begin{cases} 1 & \text{with probability } p_{i,j} \\ 0 & \text{otherwise} \end{cases}$$

Without loss of generality, it is assumed that $p_{i,j} \in [0, 1]$. The outcome of each assignment is considered an independent coin tossed with success probability $p_{i,j}$ that is observed at the end of a round. In this case, the task is completed before the end of the round, it is removed from the set of incomplete tasks T_r otherwise the task remains available for assignment in next round. Assuming that each worker is independent of other workers, the sequence of assignment outcomes for a worker is $\{y_{i,j} \mid t_i \in T, w_j = w\}$. The independence of a worker is a reasonable assumption given that workers do not form a coalition. The simplest model considers the situation when the reliability of a worker remains constant for all tasks i.e. $p_{i,j} = p_{i',j}$ [38]. In this thesis, it is assumed that worker reliability might change depending on the assigned tasks and other unobserved random factors.

Figure 4.2 illustrates the dynamic task assignment in detail for three example rounds, with probabilistic assignment success. In the first round of assignment, there are one task $T_{r_1} = \{t_1\}$ and two workers $W_{r_1} = \{w_1, w_2\}$ with their respective assignment success probabilities. The task t_1 is assigned to the worker w_2 who performs the task with success probability $p_{1,2} = 0.65$ before the next round. Before the start of second round r_2 , the previously assigned task t_1 is completed; a new task t_2 appears into the system; and two new workers $\{w_3, w_4\}$ become available. Again incomplete tasks $T_{r_2} = \{t_2\}$ are matched with available workers $W_{r_2} = \{w_1, w_2, w_3, w_4\}$. The same process continues to the next round while

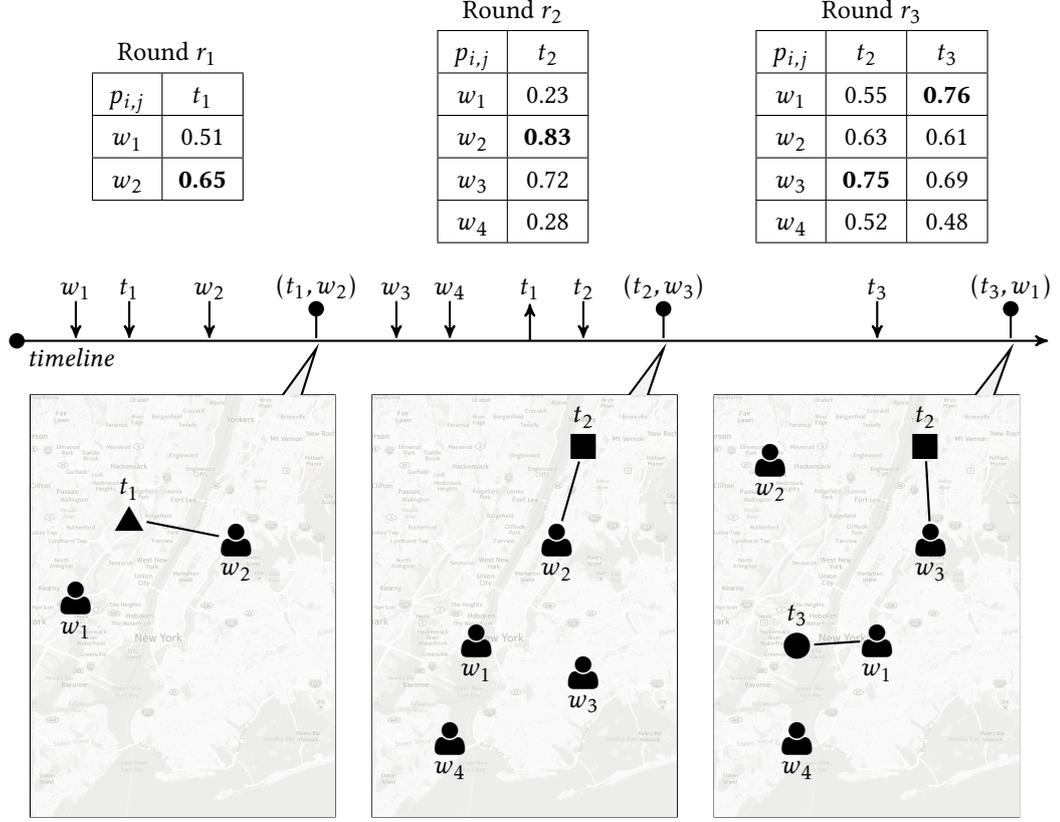


Figure 4.2: Three rounds of dynamic task assignment with probabilistic assignment success.

ensuring that previously assigned workers are not assigned to the same task again. In round r , the following function defines the cumulative reliability of assignments chosen by an algorithm.

$$P(r) = \prod_{t_i \in T_r} \prod_{w_j \in W_r} p_{i,j} \cdot x_{i,j} \quad 4.1$$

The overall objective is to design an algorithm that aims to maximize the cumulative reliability $P(r)$ over all rounds of assignment.

Definition 4.6 (Maximum Reliability Assignment). *The problem of maximum reliability assignment (MRA) is to assign workers to tasks such that the cumulative*

reliability is maximized over all rounds i.e.

$$\max \prod_{r=1}^R P(r)$$

From a theoretical perspective, previous literature discusses some restricted variants of the MRA problem. The *online stochastic matching* (OSM) considers the situation when the success of assignments are probabilistic; however, it assumes dynamic arrivals for either tasks or workers [61, 62]. Such problems have one-sided dynamism as opposed to the full dynamism of the MRA problem [152]. The most recent theoretical analysis for fully dynamic matching has been studied under assumption deterministic knowledge of outcomes [159]. A theoretical analysis of fully dynamic matchings with probabilistic knowledge assumption is out of the scope of this thesis and a recognized hard problem in theoretical computer science [62]. Instead, this thesis focuses on adaptive task assignment that considers fully dynamic matching in bipartite graphs with estimated probabilities. The estimation of probabilities requires formalization of the learning process in support of the assignment decisions. In this regard, the next section introduces the bandit problem that formulates the trade-off between learning and optimization.

4.1.3 The Bandit Problem

Learning the expected value of decision choices is one of the fundamental challenges of the sequential decision-making. Learning in adaptive task assignment necessitates the balancing of the exploration-exploitation trade-off. The bandit problem is a well-studied problem that formulates this trade-off in sequential decision-making [57]. The bandit problem assumes a gambler who wishes to play an arm from a multi-armed slot machine in a casino. In the standard version of the

bandit problem, the gambler decides to pull one arm in each play which results in a reward. The objective of the gambler is to decide which arm to play in a sequence of plays such that the cumulative rewards are maximized. This presents an exploration-exploitation trade-off since the gambler would like to maximize the reward over time by pulling best arm while learning the expectation of rewards for each arm.

Combinatorial Bandits

The *combinatorial bandits* extend the bandit problem with the choice of a combination of arms in each round of play [160, 161]. Combinatorial bandits have been studied under three level of observation [46]. *Full feedback* assumes that the outcomes and costs are observed for all workers irrespective of the assignment. *Semi-bandit feedback* means that the outcomes and costs are only observed for the assigned workers. *Bandit feedback* means that only the aggregated reward value is revealed after the assignment. This thesis formulates adaptive task assignment based on combinatorial bandits with semi-bandit feedback.

Contextual Bandits

The *contextual bandits* extends the bandit problem by considering the rewards as a function of the contextual variables [65, 66]. Instead of learning the expectation of rewards, the function specific parameters are learning to maximize cumulative rewards. Contextual bandits have been *global* [162, 163] or *individualized* parameter learning [66, 164]. In global learning, the values of function parameters are assumed to be same for all arms; whereas, individualized learning assumes different values for each arm. This thesis considers the spatial attributes of tasks and workers as the variables for contextualized learning.

4.2 Adaptive Task Assignment: A Combinatorial Contextual Bandits Approach

The basis of adaptivity in an assignment process may vary depending on the formulation of the problem and its adaptivity criteria. A short overview of the literature on adaptive assignment is provided to further illustrate the variations of adaptivity. Zhang et al. proposed a hierarchical approach for adaptive assignment in distributed settings for multi-agent systems [165]. Their proposed formulation considers multiple agents learning from each other while competing for available tasks. Ho et al. proposed an offline learning approach for adaptive task assignment, specifically for classification tasks in crowdsourcing [52]. In their formulation, the dynamically arriving workers are first served test tasks to estimate their reliability before proceeding to the assignment of classification tasks that are selected based on previously estimated reliabilities. Tarasov et al. proposed an online learning approach for adaptive task assignment for regression tasks [49]. The formulation is based on the bandit problem learning worker reliability based on correctness of worker's response against the majority response.

A common assumption among all these formulations is that the crowdsourcing platform has a fixed set of tasks and workers arrive dynamically. Furthermore, all these approaches are limited to multiple choice and non-spatial tasks. By comparison, the majority of research work on spatial crowdsourcing considers dynamic arrivals and departures for both tasks and workers. This thesis also considers the adaptive task assignment with dynamic arrivals for both tasks and workers. The basis for adaptivity in this thesis is the estimation of worker reliabilities through an appropriate learning process.

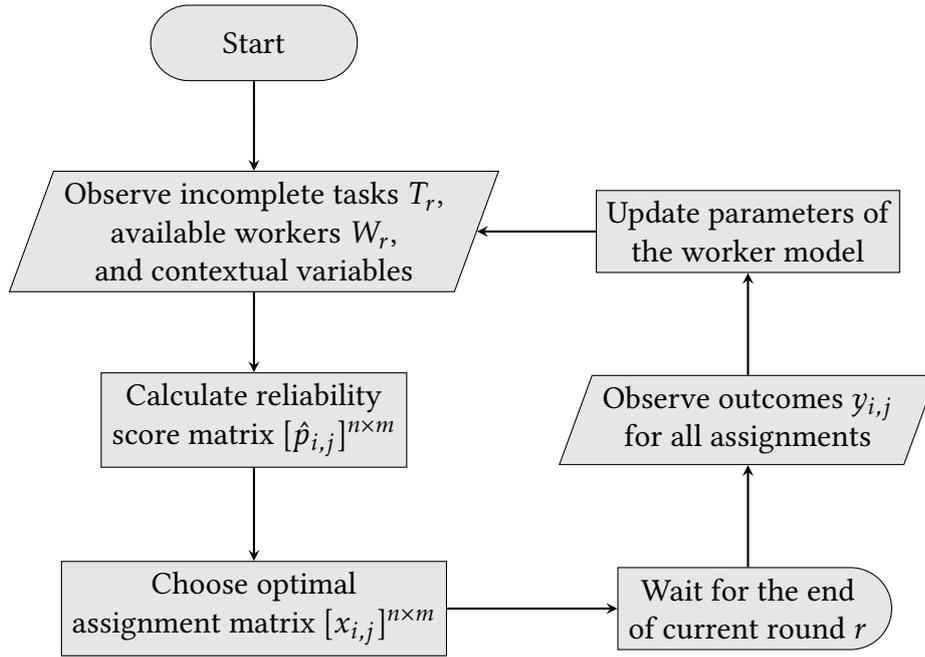


Figure 4.3: Work flow of the adaptive assignment problem.

4.2.1 The Adaptive Assignment Problem

The success of assignment depends on the task, worker, and their context; however, the worker has most of the decision-making influence. Each worker's decisions are characterized by the reliability and expertise of worker. To this end, this thesis assumes that the reliability of a worker is governed by a stochastic process that is independent of other workers. The design goal of the adaptive task assignment are two folds: generate accurate estimates of the worker reliabilities and select optimal assignment set based on those estimates. Estimation of worker reliabilities is a non-trivial problem due to the assumption of non-identical trials; instead, they can be approximated by learning from observed success of previous assignments. This thesis introduces the adaptive assignment problem in spatial crowdsourcing as a combined formulation of combinatorial bandits and contextual bandits. The combinatorial bandits formulation enable decoupling

between the learning and optimization processes. The learning process concerns approximation of worker reliability based on observed assignment outcomes and contextual variables. The optimization process concerns the selection of optimal assignment set. Figure 4.3 shows the work flow of the AAP that starts with a new round and consists of five steps:

1. Observe the set of incomplete tasks T_r , the set of available workers W_r , and their contextual variables. The contextual variables quantify both spatial and non-spatial information such as distance, location, task types, etc.
2. Calculate reliability score matrix $[\hat{p}_{i,j}]^{n,m}$. Each score variable $\hat{p}_{i,j}$ approximates the reliability of worker for each task t_i and worker w_j . The reliability scores are defined according to learning function that is defined in terms of input worker specific parameters and contextual variables.
3. Choose the assignment matrix $[x_{i,j}]^{n \times m}$ that optimizes the utility of spatial crowdsourcing. For the MRA problem, the chosen assignment set $\hat{p}_{i,j}$ maximizes the reliability score.
4. Observes the outcomes variables $y_{i,j}$ of chosen assignment set only, which equates to semi-bandit feedback.
5. Update parameters for learning function using the $y_{i,j}$. The parameters are individualized for each worker to learn worker specific reliability scores.

Figure 4.4 illustrates the adaptive task assignment process for three example rounds using a simple greedy heuristic. The reliability score of each worker is set equal to the observed success rate of assignment to the worker. The success rate parameter for each worker initialized with a fixed prior value (i.e. 0.5). After each

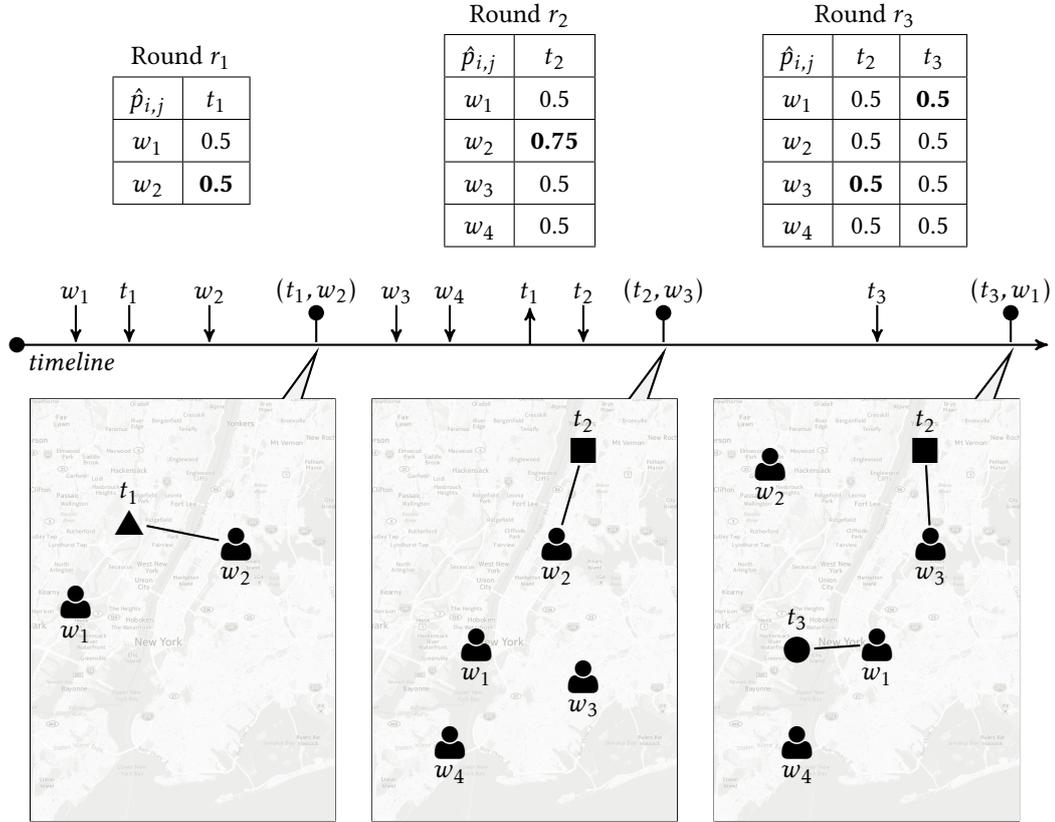


Figure 4.4: Three rounds of adaptive task assignment with estimated worker reliabilities.

round the success rate parameter is updated based on the outcomes of assignments. The assignment set is chosen that maximizes the sum of reliability scores, where ties are broken arbitrarily. As discussed in Chapter 1, the first research requirement of the adaptive assignment problem is to address the explore-exploit trade-off.

4.2.2 Heuristics based Learning

A variety of heuristics have been proposed to address the explore-exploit trade-off in the literature [46, 55]. Throughout Chapters 5 to 8 a set of assignment algorithms is presented, based on such heuristics, for different instantiations of the adaptive assignment problem.

Online Learning

Online learning address the explore-exploit trade-off by controlling the reliability scores dynamically as the assignment process proceeds through rounds [63, 64, 166]. This thesis employs following heuristics for online learning:

- **Semi-uniform** heuristics alternate between exploration and exploitation rounds [40]. Chapter 5 employs the simplest semi-uniform heuristic called the ϵ -greedy, which selects optimal assignment set during $1 - \epsilon$ proportion of rounds and random assignment set during ϵ proportion of rounds.
- **Probability matching** heuristics select assignment sets based on the probability distribution that measures the likelihood of each assignment being close to the optimal assignment [63]. Chapter 7 employs a Bayesian heuristic for probability matching called *Thompson sampling*.
- **Interval estimation:** heuristics calculate a confidence interval on the expectation of assignment success [63, 64]. Assignments with a higher upper bound of the confidence interval are preferred during each round. Chapter 5 employs the simple upper confidence bound heuristic to generate reliability scores. Chapter 6 uses the contextual upper confidence bound heuristic to calculate reliability scores as a function of spatial and non-spatial contextual variables.

Offline Learning

In comparison to the online learning, the offline learning involves estimation of worker parameters using a set of test tasks before assignment of actual tasks [55]. This offline learning, also known as *warm-start* approach, is used in place

of the fixed prior values as shown in the first round Figure 4.4. For instance, Ho et al. proposed an online learning approach for adaptive task assignment for classification tasks in web-based crowdsourcing markets [52]. In Chapter 8, this thesis considers the adaptive assignment problem with offline learning for expertise-based crowdsourcing. The expertise of a worker is defined in terms of the differences of reliability across different topics related to tasks.

4.3 Evaluation Methodology

Although recent publications have provided some initial results for partially dynamic assignment problems, the competitive analysis of dynamic task assignment algorithms is known to be a hard problem [61, 62]. In the case of the adaptive task assignment, theoretical analysis is further complicated due to the learning aspect of algorithms. The performance bounds of some learning heuristics employed in this thesis are known from previous literature [46, 64]. Still, it is well known that the empirical performance of such heuristics depends on the problem domain [63]. Simulation-based evaluation has been the standard practice for existing research on dynamic task assignment algorithms in both spatial crowdsourcing and non-spatial crowdsourcing [13, 26, 37, 38, 52]; primarily due to the practical difficulties of large-scale deployments of multiple dynamic algorithms with real people [65, 66]. This thesis takes a principled approach towards the evaluation of proposed algorithms based on an agent-based simulation.

4.3.1 Agent-based Simulation

An agent-based simulation methodology helps define the behavior of a multi-agent environment over time, using individually defined agents [167–169]. The agent-

based simulation is a well-established tool for the evaluation of human-based complex systems. It has been used for the study of population dynamics [170], humanitarian assistance [171], environmental modeling [172], and crowdsourcing systems [173]. In agent-based simulation, each agent is defined in terms of its state transitions, decision rules, and interaction with other agents. An agent can be considered a self-directed object with the capability to autonomously choose actions based on the situation. The simulation of a spatial crowdsourcing environment consists of three types of agents: requesters, workers, and platform. The following list summarizes the activities and interactions of each agent during a round:

- A *requester* agent dynamically submits new tasks to the platform during a round. In our simulations, a requester is defined in terms of a task list. The list consists of a set of tasks distributed over a spatial region and a temporal timeline. Each task consists of three values: the start round, the spatial location, and the expiry round. The requester agent is defined on in terms of distribution parameters for the start of tasks and locations.
- A *worker* agent dynamically receives tasks from the platform at the start of a round and submits responses during a round. While performing tasks the worker also moves around in a spatial region that defines spatial mobility of worker. In our simulation, a worker agent is defined in terms of a list of locations and the reliability model. The location list is distributed over a spatial region and temporal timeline. Each location in the list also has an associated time when the worker visits the location. A worker's reliability model is defined with worker specific parameters.
- A *platform* agent serves the mediator's role between requesters and workers,

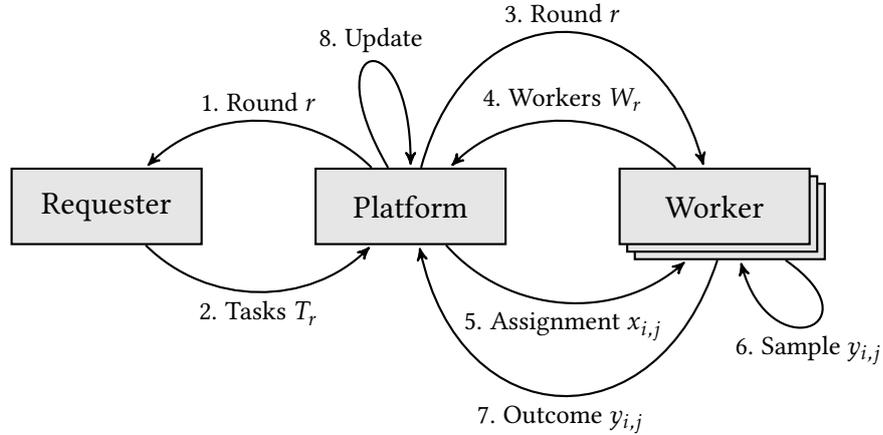


Figure 4.5: An overview of agents in the simulation process and their interactions.

while optimizing the utility of spatial crowdsourcing. An assignment algorithm is implemented in the platform agent that maintains the list of incomplete tasks and available workers. During each round, the set of tasks is matched with workers using a fixed assignment policy. In case of a learning based assignment algorithm the platform’s knowledge about workers is updated at the end of each round. The platform agent is defined in terms of the parameters that are specific to the assignment algorithm.

During the experimental evaluation, the simulation environment is configured with different parameter values of the number of agents and the duration of the simulation. Given the experimental setting in terms of the parameters of the simulation environment and its agents, the simulation process proceeds in discrete rounds. In each round, following steps are performed:

1. Platform agent queries the list of new tasks from a requester agent for the current round.
2. Requester agent returns the list of incomplete tasks to platform agent.

3. Platform agent queries the availability from worker agents.
4. Worker agents return the list of available workers to platform agent.
5. Platform agent chooses the assignment set and sends it to worker agents.
6. Worker agents sample the outcomes of assignment.
7. Worker agents return assignment outcomes to platform agent.
8. Platform agent updates its worker specific parameters and logs the assignments and outcomes.

At the end of all rounds, the performance of the assignment algorithms is compared by varying simulation and agent parameters. The simulation framework is implemented in Python programming language using Anaconda distribution ¹. Each agent is implemented as a separate class that is instantiated by the main control loop of the simulation process.

4.3.2 Data-driven Initialization

A *data-driven initialization* approach is used to parameterize the simulation environment and agent behavior. The algorithms are evaluated using both synthetically generated spatial-temporal data and real-world mobility data from a location-based social network. Location-based social networks contain voluntarily submitted data from thousands of users which indicate their mobility patterns over large time periods. Note that, the locations of worker agents are updated according to the location-based social network data. Existing research works on spatial crowdsourcing have extensively used such data sets for evaluation of algorithms, as shown in Table 4.1.

¹<http://www.continuum.io/anaconda>

Table 4.1: Datasets used for simulation-based evaluation in spatial crowdsourcing literature.

Source	Synthetic Data	Location-based Social Network		
		Gowalla	Foursquare	Yelp
Kazemi and Shahabi [13]	x	x		
Kazemi et al. [124]	x	x		
Deng et al. [36]	x	x		
To et al. [37]	x	x		
Cheng et al. [38]	x			
To et al. [37]		x		x
Chen et al. [53]	x			
Hassan and Curry [32]		x		
To et al. [25]	x	x		x
Chen et al. [26]	x			
Yu et al. [174]			x	
Yang et al. [175]			x	
Hassan and Curry [33]			x	
Yu et al. [176]			x	
Li et al. [177]	x		x	

For the purpose of simulation, the task locations are sampled from the set of unique locations defined in the dataset. The success of an assignment is modeled according to three different types of sampling processes for worker agents:

- **Check-in based process:** Given the mobility data for a worker. The agent is assumed to perform the tasks that are associated with locations visited by the worker.

$$y_{i,j} = \begin{cases} 1 & \text{if } l_i \in L_j \\ 0 & \text{otherwise} \end{cases}$$

where L_j is the set of locations a worker agent w_j visits during the simulation process. This sampling process is used for evaluation in Chapter 6.

- **Binomial process:** The assignment success is modeled as a Binomial

process.

$$y_{i,j} = \begin{cases} 1 & \text{with probability } p_{i,j} \\ 0 & \text{otherwise} \end{cases}$$

The outcome of each assignment is a coin toss with the probability $p_{i,j}$. The Binomial process is already used in existing literature of spatial crowdsourcing [37, 38]. The distribution of a worker agent's $p_{i,j}$ parameter can be informed through observed reliabilities of workers in empirical studies [20] This sampling process is used for evaluation in Chapter 5.

- **Logistic process:** The success probability of an assignment is defined in terms of a logistic function of the task and worker attributes.

$$p_{i,j} = \frac{e^{\mathbf{U}_j \mathbf{Z}_{i,j}}}{1 + e^{\mathbf{U}_j \mathbf{Z}_{i,j}}}$$

where $\mathbf{Z}_{i,j}$ is the vector of contextual variables for task t_i and worker w_j . The vector \mathbf{U}_j specifies the worker specific weights for the contextual attributes. Similar to the Binomial process, the outcome of each assignment is a coin toss with the probability $p_{i,j}$. The worker agents are initialized with the weight vectors \mathbf{U}_j that are based on empirical studies [24]. This sampling process is used for evaluation in Chapter 7.

4.4 Instantiating Adaptive Task Assignment

The contributions of this thesis can be considered from two perspectives: problem analysis and algorithm design. From a problem analysis perspective, it provides a detailed conceptual understanding of the adaptive assignment problem.

Table 4.2: An overview of the approaches used to address the research requirements of adaptive task assignment in spatial crowdsourcing.

Chapter	Explore-exploit trade-off	Spatial context	Multi-criteria optimization	Contextual learning
Chapter 5	Semi-uniform, Interval estimation	Distance	Combinatorial fractional programming	
Chapter 6	Interval estimation	Distance		Distance, Task Types
Chapter 7	Probability matching	Location	Location diversity	
Chapter 8	Greedy exploration			Task topics

The analysis allows comparison of dynamic task assignment techniques in crowdsourcing and related fields. It also serves as a benchmark for the future work to identify the existing gaps in the literature and describing the relative differences between proposed approaches. Furthermore, it facilitated the formulation of the specific hypothesis that addresses the research requirements listed in Chapter 1. The adaptive assignment problem introduced in this chapter allows domain-specific instantiation while providing necessary mathematical formalism and analytical constructs. The usefulness of the proposed formalization is established by instantiating the adaptive assignment problem in different scenarios of spatial crowdsourcing. The following list discusses the research contributions for each scenario in further detail:

1. **DRR, DRR-GRD, & DRR-UCB Algorithms:** Chapter 5 instantiates the adaptive assignment problem for bi-objective optimization in spatial crowdsourcing. It considers the scenario when the task completion is probabilistic and the travel distances are deterministic. The design goal of algorithms is to maximize the reliability of assignments and minimize the required travel. The specific contributions of this chapter in terms of

research requirements are:

- Explore-exploit trade-off: The exploration-exploitation trade-off is addressed by proposing two adaptive assignment algorithms. The DRR-UCB algorithm employs an interval estimation based heuristic for approximating reliability of assignment and the DRR-GRD algorithm uses a semi-uniform heuristic.
- Spatial context: The proposed algorithms consider the distance between tasks and workers as one of the optimization objectives; therefore, enabling spatial context in assignment decisions.
- Multi-criteria optimization: The DRR algorithm is proposed, based on the combinatorial fractional programming approach, for selecting task-worker pairs optimizing the bi-objective function. The DRR assumes dynamic task assignment with known probabilities; therefore, it supports DRR-UCB and DRR-GRD algorithms with approximated probabilities.
- Empirical performance: The empirical evidence of performance of both algorithms is provided for synthetic and real-world datasets. The results show that the proposed algorithms perform better than the non-adaptive and linear utility algorithms. The DRR algorithm achieves the least 80% lower costs when compared to competing algorithms while maintaining high reliability. The DRR-UCB algorithm achieves the best performance with unknown probabilities.
- Dimensional scalability: The algorithms are evaluated across various scalability dimensions including number of tasks, number of workers, and number of rounds. The results suggest that the relative perfor-

mance of DRR algorithm and its variants remains consistent across these dimensions.

2. **SpatialUCB Algorithm:** Chapter 6 instantiates the adaptive assignment problem for contextualized adaptive assignment in spatial crowdsourcing. It considers the scenario when the task completion is probabilistic and contextual information about tasks and workers is correlated with the probability of success for assignments. The design goal of algorithms is to learn the relationship between contextual variables and success of assignments while making assignment decisions. The contributions of this chapter in terms of research requirements are:

- **Explore-exploit trade-off:** The adaptive assignment problem with context learning is formalized according to the contextual bandits framework. The SpatialUCB algorithm is proposed that uses linear regression to learn the relationship between contextual variables and success of assignments.
- **Spatial context:** The SpatialUCB algorithm requires the locations of both workers and tasks to estimate the distance between them; therefore, enabling the spatial context in assignment decisions.
- **Contextual learning:** The SpatialUCB algorithm builds an individualized linear model for each worker to learn worker specific weights for contextual variables. The algorithm uses the type of task and the distance as the contextual variables.
- **Empirical performance:** The SpatialUCB algorithm is evaluated against non-contextual algorithms on synthetic and real-world datasets. The results show that the proposed algorithm performs better than the

best performing non-contextual algorithm. The proposed algorithm achieves 20% higher performance on real-world datasets.

- **Dimensional scalability:** The scalability of SpatialUCB algorithm is analyzed in terms of the number of tasks on a large dataset. The performance of the SpatialUCB algorithm improves significantly in comparison to best baseline, as the number of tasks increases.

3. **DynTS Algorithm:** Chapter 7 instantiates the adaptive assignment problem for generalized assignment in spatial crowdsourcing. It considers the scenario when the task completion is probabilistic and multiple workers can be assigned to a task to maximize coverage. The design goal of algorithms is to decide the number of workers to be assigned to a task and maximize the reliability of assignments. The contributions of this chapter in terms of research requirements are:

- **Explore-exploit trade-off:** The DynTS algorithm is proposed that employs a probability matching heuristic to address the exploration-exploitation trade-off.
- **Spatial context:** The DynTS algorithm considers uses the location of tasks to dynamically calculate the number of workers required for each task.
- **Multi-criteria optimization:** The DynTS algorithm uses the location diversity of a task to calculate the number of workers. An entropy measure quantifies the diversity of location based on the distribution of people visiting the task location.
- **Empirical performance:** The empirical evidence of algorithm performance is provided on real-world datasets from location-based social

networks. The results show that the DynTS algorithm performs better than the fixed size and non-adaptive algorithms. The DynTS algorithm achieves more than 90% coverage while requiring 10% fewer assignments on the average.

- Dimensional scalability: The DynTS algorithm is evaluated on two datasets each with a different number of tasks and number of workers. The relative performance of algorithm remains the same across different values.

4. **WS-GRD Algorithm:** Chapter 8 instantiates the adaptive assignment problem for expertise based in spatial crowdsourcing. It considers the scenario when task are differentiated due to domain knowledge required for performing them and workers are differentiated due to their expertise [22, 178]. In this regard, the expertise is defined in terms domain specific topics associated with each task. The design goal of algorithms to learn to workers' reliability on each topic and make assignment decisions accordingly. The contributions of this chapter in terms of research requirements are:

- Explore-exploit trade-off: The exploration-exploitation is addressed by adopting a offline learning approach, where each new worker is required to self-assess their expertise on topics and perform test tasks before assignment of actual tasks. The assignment decisions are made based on the estimated expertise of workers on domain topics.
- Contextual learning: An expertise estimation heuristic is proposed that learns worker reliability as a skills vector based on the topics related to a task. It combines a worker's self-reported expertise on the topics with the observed performance of test tasks.

Table 4.3: Datasets used for initialization of simulations in following chapters.

Source	Algorithm	Synthetic Data	Inhouse Platform Data	Location-based Social Network	
				Gowalla	Foursquare
Chapter 5	DRR DRR-GRD DRR-UCB	x			x
Chapter 6	SpatialUCB	x		x	
Chapter 7	DynTS	x			x
Chapter 8	WS-GRD		x		

- Empirical performance: The empirical performance of both algorithms is evaluation using data collected from real workers in a crowdsourcing platform. The results show that the proposed approach requires 18%-25% less exploration as compared to baseline approach while achieving a similar utility during exploitation.

Table 4.3 lists the datasets used for data-driven initialization during experimental evaluation. Chapters 5 to 7 utilize both synthetic and real-world datasets for this purpose. Chapter 8 used an in-house platform for collection of experimental data with real workers. Table 4.4 summarizes the agent specific parameters for agent-based simulation.

4.5 Chapter Summary

This chapter introduced the basic terminology and the conceptual framework used throughout the rest of this thesis. It compared the assignment protocols that dictate the design and evaluation of algorithms. The periodic-assignment protocol encompasses the assignment problem due to the dynamism of both tasks and workers. This chapter also highlighted the need for adaptive task assignment in spatial crowdsourcing, due to the observed knowledge assumption.

Table 4.4: Simulation variables used during experimental evaluation.

Agent	Parameter	Description	Chapter
Worker	$ W $	Number of workers	Chapters 5 to 8
	ρ_{min}	Minimum reliability of a worker	Chapter 5
	ρ_{max}	Maximum reliability of a worker	Chapter 5
	u_{ses}	Sampling parameter for the co-efficients of socio-economic variable	Chapter 7
	u_{dist}	Sampling parameter for the co-efficients of distance variable	Chapter 7
Requester	$ T $	Number of tasks	Chapters 5 to 8
	ξ	Duration of tasks in terms of rounds	Chapters 5 to 8
Platform	R	Number of rounds	Chapters 5 to 8
	δ	Convergence parameter for Newton's method	Chapter 5
	ε	Exploration parameter of semi-uniform learning	Chapter 5
	ϵ	Exploration parameter of semi-uniform learning	Chapter 6
	τ	Temperature parameter of Softmax heuristic	Chapter 6
	λ	Balance parameters for interval estimation heuristic	Chapter 6
	b	Maximum budget of worker to be assigned to a task	Chapter 7
	α_0	First parameter of Thompson sampling heuristic	Chapter 7
	β_0	Second parameter of Thompson sampling heuristic	Chapter 7

It introduced the adaptive assignment problem in spatial crowdsourcing. The adaptive assignment problem provides the necessary formalism for the design and comparison of appropriate assignment algorithms. This chapter also introduced an evaluation methodology based on agent-based simulation and data-driven initialization. The methodology utilizes both synthetic and real-world datasets for initializing the simulation process. The real-world dataset is based on the human mobility patterns collected from location-based social networks.

Chapter 5

Adaptive Assignment with Distance-Reliability Optimization

Chapter 4 introduced the theoretical foundations for adaptive assignment problem; however, the exact definition of success depends on the worker model and optimization requirements. The distance between task and worker location plays a significant role in spatial crowdsourcing. It is intuitive to assume that workers would find it easier to perform tasks near to them [24, 58]. From a requester's perspective, workers in the near vicinity of a task have a higher chance of completing the task in time [25]. Figure 5.1 illustrates a scenario where the probability of success for each assignment as well as its related travel distance is known at the time of assignment decisions. In such scenario, an appropriate algorithm not only has to find highly reliable assignments but also assignments that require minimum travel. This scenario highlights the spatial context and multi-criteria optimization requirements for adaptive task assignment in spatial

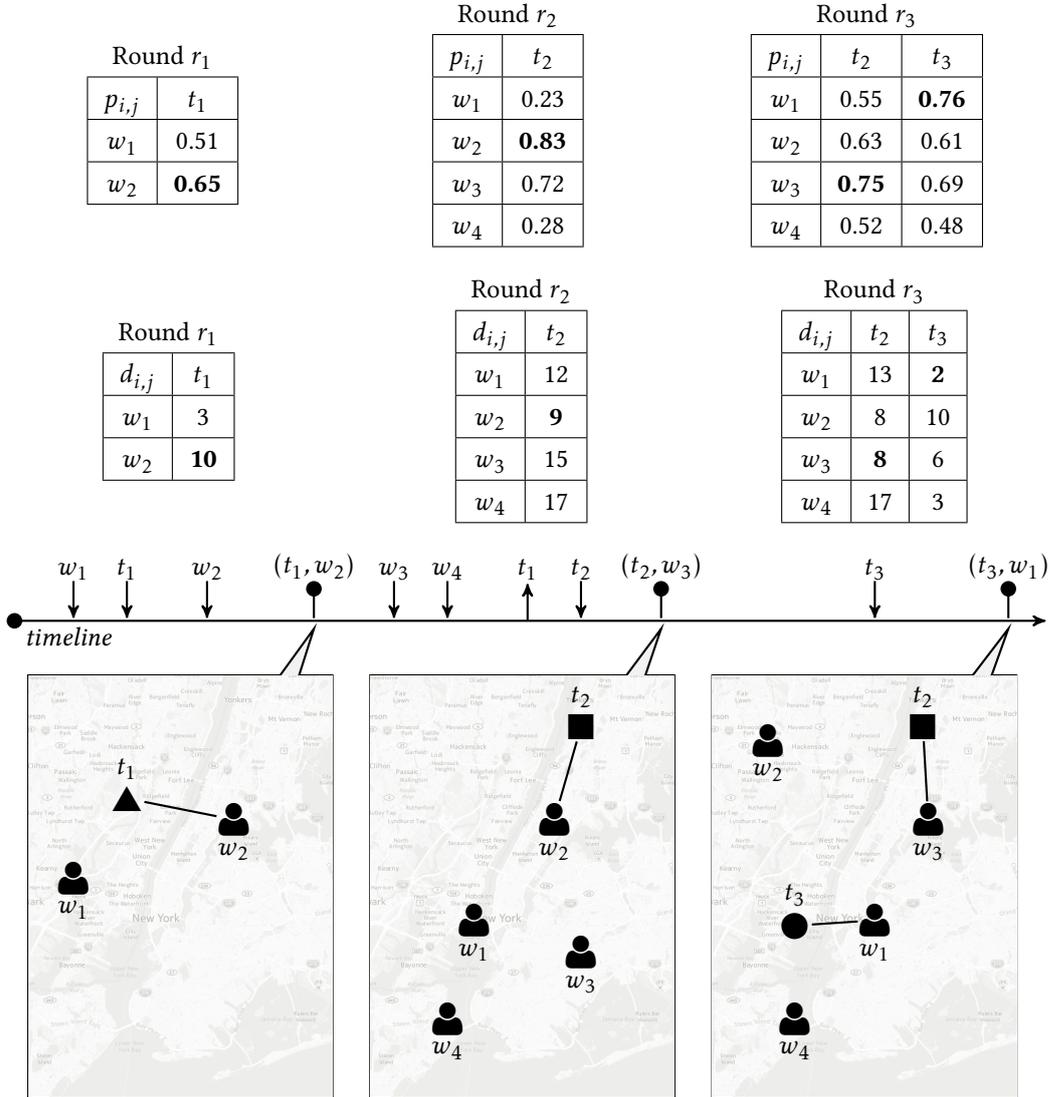


Figure 5.1: Three rounds of dynamic task assignment with worker reliability and distance variables.

crowdsourcing; further, the need to address the explore-exploit trade-off becomes evident when the probability of success for each assignment is unknown.

Section 5.1 introduces the *minimum cost maximum reliability assignment* (MC-MRA) problem that extends the *maximum reliability assignment* (MRA) problem introduced in Chapter 4 for bi-objective optimization. Section 5.2 first presents a local optimization approach for the MRA problem and then extends it to

address the MC-MRA problem. Section 5.3 proposes a combinatorial fractional programming approach for MC-MRA problem that addresses the limitations of the baseline approaches. Section 5.4 instantiates the MC-MRA problem under the observed knowledge assumption of adaptive assignment problem and proposes two adaptive task assignment algorithms based on the semi-uniform and interval estimation heuristics for online learning, respectively. Section 5.5 presents an experimental evaluation of the proposed algorithms on both synthetic and real-world datasets. Section 5.6 highlights limitations of proposed approaches and Section 5.7 summarizes the chapter with key findings.

5.1 Problem Description

Let $G(W_r, T_r, E_r)$ denote a dynamically changing bipartite graph. The two sets of vertices W_r and T_r are disjoint sets of size m and n , respectively. The set E_r contains edges that connect vertices in W_r with vertices in T_r . Apart from the worker reliability $p_{i,j}$, each edge in E_r has an associated weight $c_{i,j}$ that quantifies the cost of assigning task t_i to the worker w_j . In this case, the *maximum reliability assignment* (MRA) problem is extended as follows:

Definition 5.1 (Minimum Cost Maximum Reliability Assignment). *The problem of minimum cost maximum reliability assignment (MC-MRA) is to assign workers to tasks such that the reliability is maximized and travel costs are minimized over all rounds.*

$$\max \prod_{r=1}^R \prod_{t_i \in T_r} \prod_{w_j \in W_r} p_{i,j} \cdot x_{i,j} \quad \text{and} \quad \min \sum_{r=1}^R \sum_{t_i \in T_r} \sum_{w_j \in W_r} c_{i,j} \cdot x_{i,j}$$

In spatial crowdsourcing, the travel costs are be defined in terms of distance to

be traveled for performing a task or the time need to for travel and task completion. This thesis assumes that the travel costs are deterministic and revealed at the time of assignment decision. The basic premise is to incorporate the travel costs in the assignment decisions. Subsequently, the assignment decisions are not only targeted at the reliability but also the spatial characteristics of tasks and workers as well.

Definition 5.2 (Assignment Distance). *Each edge in E_r has an associated distance $d_{i,j}$ that quantifies the cost in terms of the travel required from worker w_j while assigned to the task t_i .*

The distance may be calculated according to different metrics: Euclidean distance, Haversine formula, or travel time. It is assumed that one consistent metric is used to quantify the distance between two locations; therefore, smaller distances are preferred by both worker and requesters.

5.2 Baseline Approaches

Given the definitions of MRA and MC-MRA problems for spatial crowdsourcing, two baseline assignment approaches are presented to address them in this section.

5.2.1 Maximum Weighted Bipartite Matching

A global solution to the MRA problem is not feasible unless the assignment algorithm is clairvoyant i.e. all information about rounds and probabilities is known beforehand. Instead, the algorithm can aim to maximize reliability locally in each round.

Definition 5.3 (Local Maximum Reliability Assignment). *The problem of local maximum reliability assignment is to assign workers to tasks in a round such that the reliability is maximized i.e. for round r*

$$\max \prod_{t_i \in T_r} \prod_{w_j \in W_r} p_{i,j} \cdot x_{i,j}$$

A simple local optimization strategy is introduced that reduces the local maximum reliability assignment problem to the *maximum weight bipartite matching* (MWBM) problem. For this purpose, the reliability of an assignment set (see Equation 4.1) in round r is re-written as the reliability score:

$$\bar{P}(r) = -\ln \left(\prod_{t_i \in T_r} \prod_{w_j \in W_r} p_{i,j} \right) = \sum_{t_i \in T_r} \sum_{w_j \in W_r} -\ln(p_{i,j}) \quad 5.1$$

Based on above equation, the goal of maximizing the reliability for all rounds (see Equation 4.1) is equivalent to maximizing $\bar{P}(r)$ over all rounds. Given the time interval of R rounds, the global objective is to maximize the sum of assignment scores for all rounds, i.e. $\sum_{r=1}^R \bar{P}(r)$. In a round, the goal of local optimization strategy is to choose an assignment set such that the reliability score is maximized; thereby, solving the local MRA problem.

Theorem 5.4. *The local maximum reliability assignment problem is reducible to the maximum weight bipartite matching problem.*

Proof. The theorem is proved for a round r in which the set of available workers is W_r and the set of incomplete tasks is T_r . Recall that $G(W_r, T_r, E_r)$ denotes a dynamically changing bipartite graph. The two sets of vertices W_r and T_r are disjoint sets of size m and n , respectively. The set of edges E_r connects vertices in T_r with vertices in W_r . A match is deemed valid only if the vertex for task t_i and

the vertex for worker w_j appears in at most one edge in E_r . Meaning that each task is assigned to at most one worker and each worker gets at most one task. A weight is associated with each edge according to following reliability score:

$$\hat{p}_{i,j} = \begin{cases} -\ln(p_{i,j}) & \text{if assignment of } t_i \text{ is allowable to } w_j \\ 0 & \text{otherwise} \end{cases} \quad 5.2$$

An assignment is not allowable when the worker did not complete the same task in a previous round. The zero score discourages assignment of the same worker to a task that was not completed in previous rounds. Subsequently, the local MRA problem reduces to finding the solution to MWBM in graph G . \square

Polynomial time algorithms for the MWBM problem have been proposed using techniques based on network flows [179] or linear programming [141]. The algorithm can be employed to find a solution for the local MRA problem in each round. This thesis employs the well-known Hungarian algorithm. For this purpose, a cost matrix is generated based on edge weights defined in Equation 5.2. The cost of an edge is set to

$$c_{i,j} = \left(\max_{t_i \in T_r, w_j \in W_r} \hat{p}_{i,j} \right) - \hat{p}_{i,j} \quad 5.3$$

The following integer linear program specifies equivalent integer program to

be solved using the Hungarian algorithm:

$$\begin{aligned}
 \min \quad & \sum_{t_i \in T_r} \sum_{w_j \in W_r} c_{i,j} \cdot x_{i,j} & 5.4 \\
 \text{s.t.} \quad & \sum_{w_j \in W_r} x_{i,j} = 1 & \forall t_i \in T_r \\
 & \sum_{t_i \in T_r} x_{i,j} = 1 & \forall w_j \in W_r \\
 & x_{i,j} \in \{0, 1\} & \forall t_i \in T_r, w_j \in W_r
 \end{aligned}$$

where the binary variable $x_{i,j}$ indicates assignment of a task to a worker. The Jonker-Volegant variant of the Hungarian algorithm was used during experiments since it has an improved time complexity of $O(n^3)$, where $n > m$ [180]. The maximum weight bipartite matching discussed here optimizes reliability locally for each round; therefore, the global optimization is not guaranteed. Furthermore, this approach only considers the reliability as the optimization criteria while ignoring spatial characteristics of tasks and workers. Such a naive approach may result in an unnecessary burden on workers in terms of travel cost to the task locations. To overcome this issue, an appropriate algorithm can be designed to solve the MC-MRA problem over all rounds of assignment.

5.2.2 Close Distance Priority

Similar to the MRA problem, a global solution to MC-MRA is not possible without being clairvoyant. Note that, the assignment decisions in MC-MRA problem are dependent on two types of variables for each assignment: the reliability and the travel cost. The distribution of reliabilities and travel costs observed during previous rounds of assignment can also be exploited. Nevertheless, the local MC-

MRA problem is reduced to the minimum-cost MWBM problem.

Theorem 5.5. *The local minimum cost maximum reliability assignment problem is reducible to the minimum-cost maximum weight bipartite matching problem.*

Proof. Similar to the Theorem 5.4, this proof is based on a round r where assignments are made over a set of tasks T_r and a set of workers W_r . Let $G'((W_r, T_r, E_r))$ be the undirected weighted bipartite graph constructed in a similar way as in the proof of Theorem 5.4. Each edge in the E_r has an associated score $\hat{p}_{i,j}$. A cost $c_{i,j}$ is also associated with each edge based on the distance between the locations of task and worker. Therefore, each edge in G' has two associated values: a reliability score and a cost. The solution to the local MC-MRA problem reduces to finding a bipartite matching with bi-objective optimization i.e. cost minimization and score maximization. \square

A simple approach is to solve the local MC-MRA problem sequentially [140]. The first step is to find assignment sets with the maximize possible value for the reliability objective. Then the assignment set with minimum travel costs is selected among the available maximal assignment sets. This sequential approach is also known as the *close distance priority* (CDP) approach [25]. In each round, the Hungarian algorithm is used to find the maximal score assignment matrix $[x_{i,j}^{max}]^{n \times m}$ using Equation 5.4. Let $fmax$ be the total reliability score of a maximal score assignment set, i.e.

$$fmax = \sum_{t_i \in T_r} \sum_{w_j \in W_r} x_{i,j}^{max} \cdot \hat{p}_{i,j}$$

In the next step, the cost minimization problem is reformulated as the following

integer program:

$$\begin{aligned}
 \min \quad & \sum_{t_i \in T_r} \sum_{w_j \in W_r} d_{i,j} \cdot x_{i,j} & 5.5 \\
 \text{s.t.} \quad & \sum_{t_i \in T_r} \sum_{w_j \in W_r} \hat{p}_{i,j} \cdot x_{i,j} \geq fmax \\
 & \sum_{w_j \in W_r} x_{i,j} = 1 \quad \forall t_i \in T_r \\
 & \sum_{t_i \in T_r} x_{i,j} = 1 \quad \forall w_j \in W_r \\
 & x_{i,j} \in \{0, 1\} \quad \forall t_i \in T_r, w_j \in W_r
 \end{aligned}$$

where $d_{i,j} = \text{distance}(t_i, w_j)$. A branch and cut technique can be employed for solving the integer program formulated in Equation 5.5 [181]. The result is the minimal cost assignment among all feasible assignments with a total reliability score of at least $fmax$. The CDP approach finds a solution for the local MC-MRA problem in each round.

5.3 Distance-Reliability Ratio

The CDP approach is applicable to situations when there are more than one feasible solutions for the first phase of the MC-MRA problem; therefore, priority is given to the solution with the smallest total costs. This approach does not account for situations when there are limited possible solutions in the first phase and the costs for those solutions are significantly high. To address this issue, the *distance reliability ratio* (DRR) approach is proposed based on the combinatorial fractional programming. The basic idea is to formulate the local MC-MRA problem as an integer linear-fractional program, where the optimization objective is the ratio of

Algorithm 1 The DRR algorithm

Require: $\delta, [\hat{p}_{i,j}]^{n \times m}, [d_{i,j}]^{n \times m}$

 1: $[x_{i,j}]^{n \times m} \leftarrow [0]^{n \times m}$

 2: **repeat**

3:

$$\varphi \leftarrow \frac{\sum_{i=1}^n \sum_{j=1}^m d_{i,j} \cdot x_{i,j}}{\sum_{i=1}^n \sum_{j=1}^m \hat{p}_{i,j} \cdot x_{i,j}}$$

 4: $[c_{i,j}]^{n \times m} \leftarrow [d_{i,j}]^{n \times m} - \varphi \cdot [\hat{p}_{i,j}]^{n \times m}$

 5: $[x_{i,j}]^{n \times m} \leftarrow \text{Hungarian}([c_{i,j}]^{n \times m})$

 6: $f_\varphi \leftarrow \sum_{i=1}^n \sum_{j=1}^m (d_{i,j} - \varphi \hat{p}_{i,j}) \cdot x_{i,j}$

 7: **until** $f_\varphi \geq \delta$

 8: **return** $[x_{i,j}]^{n \times m}$

two linear functions (i.e. cost and reliability score). The following integer program aims to minimize the costs over reliability:

$$\begin{aligned} \min & \frac{\sum_{t_i \in T_r} \sum_{w_j \in W_r} d_{i,j} \cdot x_{i,j}}{\sum_{t_i \in T_r} \sum_{w_j \in W_r} \hat{p}_{i,j} \cdot x_{i,j}} & 5.6 \\ \text{s.t.} & \sum_{w_j \in W_r} x_{i,j} = 1 \quad \forall t_i \in T_r \\ & \sum_{t_i \in T_r} x_{i,j} = 1 \quad \forall w_j \in W_r \\ & x_{i,j} \in \{0, 1\} \quad \forall t_i \in T_r, w_j \in W_r \end{aligned}$$

It can be safely assumed that the denominator $\sum_{i=1}^n \sum_{j=1}^m \hat{p}_{i,j} \cdot x_{i,j}$ is always non-zero because $p_{i,j} \in (0, 1)$. Finding a direct solution to the above fractional program is a difficult problem; therefore, it is common to solve a linearized equivalent of Equation 5.6. Dinkelbach proposed a parametric approach to iteratively solve the

linearized version of linear-fractional programs, known as the Newton's method [182]. The same method is applied here to transform the Equation 5.6 to an equivalent integer linear program using a Charnes and Cooper transformation [183]. The following integer program formulates the transformed program with the variable φ :

$$\begin{aligned}
 \min \quad & \sum_{t_i \in T_r} \sum_{w_j \in W_r} (d_{i,j} - \varphi \hat{p}_{i,j}) \cdot x_{i,j} & 5.7 \\
 \text{s.t.} \quad & \sum_{w_j \in W_r} x_{i,j} = 1 & \forall t_i \in T_r \\
 & \sum_{t_i \in T_r} x_{i,j} = 1 & \forall w_j \in W_r \\
 & x_{i,j} \in \{0, 1\} & \forall t_i \in T_r, w_j \in W_r
 \end{aligned}$$

Similar to the approaches discussed previously, the Hungarian algorithm is employed to solve the above assignment problem [184]. To discourage assignment of a previously unsuccessful worker to an incomplete task, the value of cost variable $(d_{i,j} - \varphi \hat{p}_{i,j})$ is replaced with a reasonably high value. The variable φ is updated in each iteration using the Newton's method [185]. Algorithm 1 summarizes the parametric algorithm for solving MC-MRA using the DRR approach. The DRR algorithm requires an optimality parameter δ which takes on reasonably small values. This parametric approach is known to quickly coverage to a solution since the number of iterations has strong polynomial bounds [184, 186].

5.4 Estimating Reliabilities

So far, it is assumed that the worker reliabilities are known at the time of assignment decisions. In reality the worker reliabilities are unknown; therefore, this assumption is now relaxed. Instead, the assignment process must proceed under uncertainty and learn from the observed outcomes of assignments in previous rounds. The assignment algorithms can estimate worker reliabilities by employing online learning techniques; therefore, the MC-MRA problem is cast as an adaptive assignment problem. The objective of learning is to minimize the difference between estimated worker reliability $\hat{p}_{i,j}$ and actual worker reliability $p_{i,j}$. In an online learning setting, the assignment algorithm must choose assignments for optimizing the MC-MRA and for generating good estimates. Two exploration strategies for estimating $\hat{p}_{i,j}$ over time are presented: one is based on a semi-uniform approach and the other follows an interval estimation approach.

5.4.1 Greedy Exploration

The greedy exploration approach is implemented in two phases: a random exploration phase is followed by pure exploitation phase [64]. The basic idea is to randomly choose sub-optimal actions at the start to quickly generate estimates for reliability. Afterwards, assignments are chosen greedily based on the existing estimates of reliability while also learning from the outcomes. The duration of the exploration phase is controlled through an appropriately tuned parameter.

Algorithm 2 details the complete assignment process for R number of rounds based on the DRR assignment and the greedy exploration. The algorithm maintains two variables for each worker w_j : the estimated reliability μ_j and the

Algorithm 2 The DRR-GRD algorithm

Require: $R, \varepsilon, \delta, T, W$

```

1:  $M \leftarrow |W|$ 
2:  $[\mu_j]^M \leftarrow [0]^M$                                      {Initialize estimates}
3:  $[\theta_j]^M \leftarrow [0]^M$                                  {Initialize counters}
4: for  $r \leftarrow 1$  to  $R$  do
5:    $T_r \leftarrow \text{Active}(T)$                                {Set of incomplete tasks}
6:    $W_r \leftarrow \text{Available}(W)$                          {Set of available workers}
7:    $n \leftarrow |T_r|$ 
8:    $m \leftarrow |W_r|$ 
9:    $[d_{i,j}]^{n \times m} \leftarrow \text{Distance}(T_r, W_r)$      {Distance matrix}
10:  if  $r \leq \varepsilon$  then
11:     $[\hat{p}_{i,j}]^{n \times m} \leftarrow [\mathcal{U}(0,1)]^{n \times m}$    {Random scores}
12:  else
13:     $[\hat{p}_{i,j}]^{n \times m} \leftarrow \text{Greedy}(T_r, W_r, [\mu_k]^M)$  {Greedy scores}
14:  end if
15:   $[x_{i,j}]^{n \times m} \leftarrow \text{DRR}(\delta, [\hat{p}_{i,j}]^{n \times m}, [d_{i,j}]^{n \times m})$ 
16:   $\text{Assign}(t_i, w_j) \quad \forall x_{i,j} > 0$                    {Assign tasks}
17:   $\text{Wait}(\tau)$                                            {Wait for the end of round}
18:  for all  $x_{i,j} > 0$  do
19:     $y_{i,j} \leftarrow \text{Complete}(t_i, w_j)$                  {Completion indicator}
20:     $\mu_j \leftarrow (y_{i,j} + \theta_j \cdot \mu_j) / (\theta_j + 1)$  {Update estimates}
21:     $\theta_j \leftarrow \theta_j + 1$                              {Update counters}
22:  end for
23: end for

```

number of assignments to the worker θ_j . Each round starts with a listing of incomplete and active tasks along with available workers (Line 5-8). A distance matrix is calculated between incomplete tasks and available workers by calling the *Distance* sub-routine (Line 9). The algorithm requires a parameter ε that dictates how the reliability scores are calculated during each round. For the

first ε rounds the score matrix is sampled using a standard Uniform distribution; therefore, resulting in random assignments for the purpose of pure exploration (Line 11). During the rest of the rounds, the score matrix is calculated based on the estimated reliabilities by calling the *Greedy* sub-routine (Line 13). The sub-routine approximates the worker reliability as $\hat{p}_{i,j} \approx \mu_j$ and uses Equation 5.2 to calculate the elements of the score matrix. The algorithm uses the DRR algorithm for solving the local MC-MRA problem using distance and score matrices (Line 15). At the end of the round, the algorithm observes the outcomes of the chosen assignments and updates the μ_j estimates and counters variables accordingly (Line 19-23).

Note that the computational complexity of Algorithm 2 is bound by the complexity of the Algorithm 1, since all other operations are linear with respect to the number of tasks and number of workers. The performance of the DRR-GRD algorithm in terms of approximating the actual worker reliabilities is dependent on the exploration parameter ε . Understandably, a very small value of the exploration parameter ε may result in inaccurate estimates which may lead to sub-optimal exploitation. Conversely, large values result in a high ratio of sub-optimal assignments due to over exploration.

5.4.2 Optimistic Exploration

The interval estimation approach does not make any explicit distinction between exploration and exploitation. Instead, assignments are chosen optimistically by giving preference to workers which have not been explored previously. The most widely known variant of this approach is based on the *upper confidence bound* (UCB) heuristic [64]. The basic idea is to calculate the confidence interval for the estimated worker reliability μ_j and define an upper bound for the expected

Algorithm 3 The DRR-UCB algorithm

Require: $R, \varepsilon, \delta, T, W$

```

1:  $M \leftarrow |W|$ 
2:  $[\mu_j]^M \leftarrow [0]^M$  {Initialize estimates}
3:  $[\theta_j]^M \leftarrow [0]^M$  {Initialize counters}
4: for  $r \leftarrow 1$  to  $R$  do
5:    $T_r \leftarrow \text{Active}(T)$  {Set of incomplete tasks}
6:    $W_r \leftarrow \text{Available}(W)$  {Set of available wokrsers}
7:    $n \leftarrow |T_r|$ 
8:    $m \leftarrow |W_r|$ 
9:    $[d_{i,j}]^{n \times m} \leftarrow \text{Distance}(T_r, W_r)$  {Distance matrix}
10:   $[s_{i,j}]^{n \times m} \leftarrow \text{UCB}(T_r, W_r, [\mu_k]^M, [\theta_k]^M)$  {UCB scores}
11:   $[x_{i,j}]^{n \times m} \leftarrow \text{DRR}(\delta, [s_{i,j}]^{n \times m}, [d_{i,j}]^{n \times m})$ 
12:   $\text{Assign}(t_i, w_j) \quad \forall x_{i,j} > 0$  {Assign tasks}
13:   $\text{Wait}(\tau)$  {Wait for the end of round}
14:  for all  $x_{i,j} > 0$  do
15:     $y_{i,j} \leftarrow \text{Complete}(t_i, w_j)$  {Completion indicator}
16:     $\mu_j \leftarrow (y_{i,j} + \theta_j \cdot \mu_j) / (\theta_j + 1)$  {Update estimates}
17:     $\theta_j \leftarrow \theta_j + 1$  {Update counters}
18:  end for
19: end for
    
```

values of estimates based on the confidence interval. During each round, actions are chosen by giving preference to higher upper confidence bounds instead of the actual estimates. The higher the uncertainty of the estimates the higher the chance of the worker being selected in a round. The uncertainty is reduced over time due to the optimistic exploration strategy unless the worker population is highly unstable.

Algorithm 3 summarizes the assignment process based on the DRR assignment and the optimistic exploration. Similar to Algorithm 2, it stores both estimates and counter variables for each worker. During each round, the score matrix is

generated using the UCB sub-routine (Line 10). For each assignment between task t_j and worker w_j , the reliability score is calculated by using Equation 5.2. The worker reliability is approximated as follows:

$$\hat{p}_{i,j} \approx \mu_j + \sqrt{\frac{3 \ln(r)}{2\theta_j}}$$

where the second term quantifies the upper bound on the confidence interval for the estimated reliability of the worker. The computational complexity of the DRR-UCB algorithm is also dominated by the DRR sub-routine. Each assignment round takes at most $O(n^3)$ time when $n > m$ and vice versa. Besides computational complexity, online learning algorithms are also analyzed under the notion of *regret*. Regret is defined as the difference between an algorithm's decisions and optimal decisions in a sequential decision-making problem. Regret analysis of the proposed algorithm is non-trivial; hence, it is left as future work. A generalized regret analysis of upper confidence bound based algorithms can be found in [160].

5.5 Experimental Evaluation

The performance of proposed algorithms (DRR, DRR-GRD, DRR-UCB) against baseline algorithms (MWBM, CDP) is evaluated in terms of task completion rate and average travel costs. For this purpose, a set of experiments were performed on both real-world and synthetic data, Next, the experimental methodology followed during the experiments under various settings is presented.



Figure 5.2: Distribution of check-ins in the New York dataset from Foursquare

5.5.1 Evaluation Methodology

As discussed in Chapter 4, evaluation of online algorithm with large-scale deployments of prototypes is prohibitively expensive and time-consuming [65]. Existing research works have circumvented this issue by simulating crowdsourcing environments using real-world data. Specifically, datasets from location-based social networks [13, 25, 36, 37, 175], mobile networks [41], and urban transport systems [38] have been used to evaluate crowd-based algorithms. A similar methodology is followed to evaluate the performance of the proposed algorithms.

A principled approach is taken for evaluation of the algorithms using an agent-based simulation methodology [173, 187]. The agent-based simulation methodology instantiates a platform agent, a request agents, and a set of worker agents. The simulation process proceeds in discrete rounds by exchanging data between these agents (see Chapter 4 for further details). All agents in the simulation were instantiated using a data-driven initialization; next section details the datasets used for this purpose.

Table 5.1: Overview of dataset used for data-driven initialization

	Synthetic Dataset	New York Dataset
Task location, l_i	$latitude \sim \mathcal{U}(0, 1), longitude \sim \mathcal{U}(0, 1)$	38,333 locations
Workers, W	1000	1,083 users
Worker locations, $l_j(r)$	$latitude \sim \mathcal{U}(0, 1), longitude \sim \mathcal{U}(0, 1)$	227,428 check-ins

Datasets

Both synthetic and real-world data were used to populate the variables of each simulated agent. The real-world dataset is based on data collected from a popular location-based social network: Foursquare¹. The dataset contains check-ins, by people, on various locations in New York city from April 2012 to February 2013 [175]. The dataset contains 1,083 unique users, 38,333 unique locations, and 227,428 check-ins. A *check-in* represents the visitor relationship between a user and a location at a particular time. Figure 5.2 shows the distribution of check-ins as a heat map, where red areas indicate a higher concentration of check-ins. The users in the Foursquare dataset are considered as the crowd workers and initialize each agent’s visited locations according to the check-ins of the corresponding user. The requester agent is initialized by sampling check-ins in the Foursquare dataset. Default settings were used for synthetic datasets for all other experiment settings, as described below.

Based on the existing literature, the task and worker locations were randomly initialized 2D space such that $latitude \sim \mathcal{U}(0, 1)$ and $longitude \sim \mathcal{U}(0, 1)$ [25, 38]. The worker agents were defined in terms of their average reliability ρ_j . The average reliabilities were sampled from a parameterized uniform distribution i.e. $\rho_j \sim \mathcal{U}(\rho_{min}, \rho_{max})$. For each worker, the reliabilities on tasks were also sampled from uniform distribution i.e. $p_{i,j} \sim \mathcal{U}(\rho_j, 0.1)$. Table 5.1 summarizes both datasets

¹<https://www.foursquare.com/>

used for data-driven initialization. The starting round for each task also uniformly distributed such that $r_i \sim \mathcal{U}(1, 90)$. The expiry time for tasks was initialized to a fixed value; such that, the task expires at the end of round $r_i + \xi$. We used Euclidean distance to quantify the travel costs between tasks and workers.

Metrics

Four metrics were used for the evaluation of assignment algorithms as described below:

- **Average Reliability** is the mean reliability over all tasks at the end of all rounds. Task reliability is the probability of success for the last assignment to the task.
- **Average Travel Cost** is the average of the distances to be traveled by workers assigned to tasks. Only the distances for the completed tasks were considered for the purpose of reporting.
- **Assignments per Task** is the number of assignments made for each task until it is performed and expires. Lower reliabilities should result in less chance of task completion; therefore, resulting in more assignments in subsequent rounds. Note that, the number of assignment for a task is bound by the expiry time ξ .
- **Task Completion Rate** is the percentage of tasks completed with high quality after the end of all rounds. From a requester's perspective, task completion is the primary success criteria.

Table 5.2: Experiment settings used for experimental evaluation

Agent	Parameter	Description	Values
Worker	$[\rho_{min}, \rho_{max}]$	Range of worker reliabilities	$[0.2, 0.5]$, $[0.2, 0.6]$, $[0.2, 0.7]$, $[0.2, 0.8]$
	$ W $	Number of workers	100 , 500, 1000
Requester	$ T $	Number of tasks	500, 1000 , 5000
	ξ	Duration of tasks in terms of rounds	1, 3 , 5, 7, 9
Platform	R	Number of rounds	90
	δ	Convergence parameter for Newton's method	0.01, 0.1 , 0.2, 0.3
	ε	Exploration parameter of semi-uniform learning	0.05, 0.1, 0.2

Experiment Settings

Each experiment was performed by changing a single parameter value while keeping others fixed. All of the experiments were run on an Intel Core i7-4600 CPU @2.90 GHz with 16 GB RAM. The algorithms were implemented using the open source libraries in Python. The Jonker and Volgenant variant of the Hungarian algorithm, as implemented in Pymatgen² library [188], was used for implementing the MWBM, CDP, and DRR approaches. The linear programming phase of CDP approach was implemented using Pyomo³ optimization library [189] and the GNU Linear Programming Kit⁴. Table 5.2 lists the range of values for the experimental settings and algorithm parameters, with default values in bold font. All reported metrics are based on the average of 10 runs of the same experiment settings.

²<http://pymatgen.org/>

³<http://www.pyomo.org/>

⁴<http://www.gnu.org/software/glpk/>

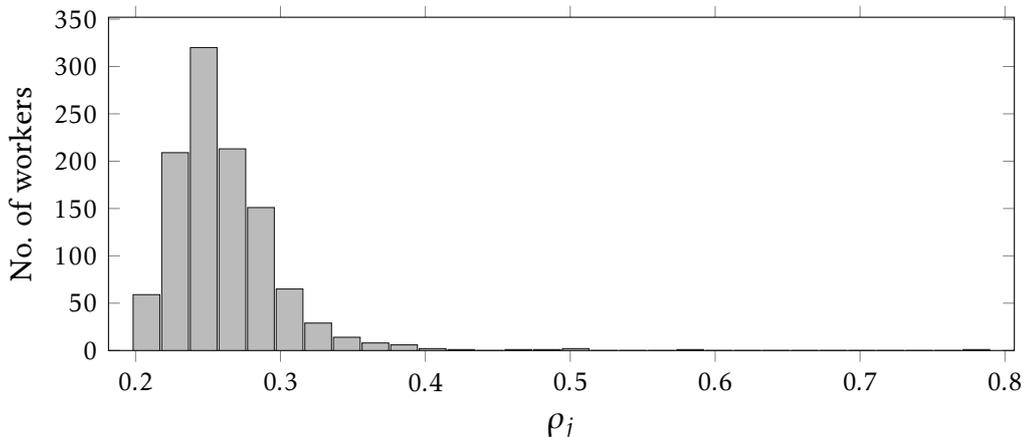


Figure 5.3: Distribution of worker reliabilities in New York dataset

5.5.2 Experiments on Real-world Data

All of the algorithms discussed in this chapter, as well as the baseline random assignment algorithm, were compared using the real-world dataset. The reliabilities of workers are initialized based on the ratio of unique locations in a worker’s check-ins history against the total number of locations. Figure 5.3 shows the distribution of worker reliabilities that shows that the majority of workers have low reliabilities, similar to the distributions observed in commercial crowdsourcing systems [20]. The time duration of a round is initialized to a day. For each worker, the location of the last check-in is considered as the current location of the worker in a round. Dynamically arriving tasks were simulated by uniformly sampling 50,000 check-ins; therefore, the distribution of tasks locations and start time was the same as the distributions of check-ins. In summary, there are 322 rounds, 1,083 workers, and 50,000 tasks in the experiments based on the real-world dataset.

Figure 5.4 shows the comparison of algorithms in terms of average reliability. Understandably, all the algorithms discussed in this chapter outperform the baseline algorithm. The MWBM, CDP, and DRR algorithms perform at similar

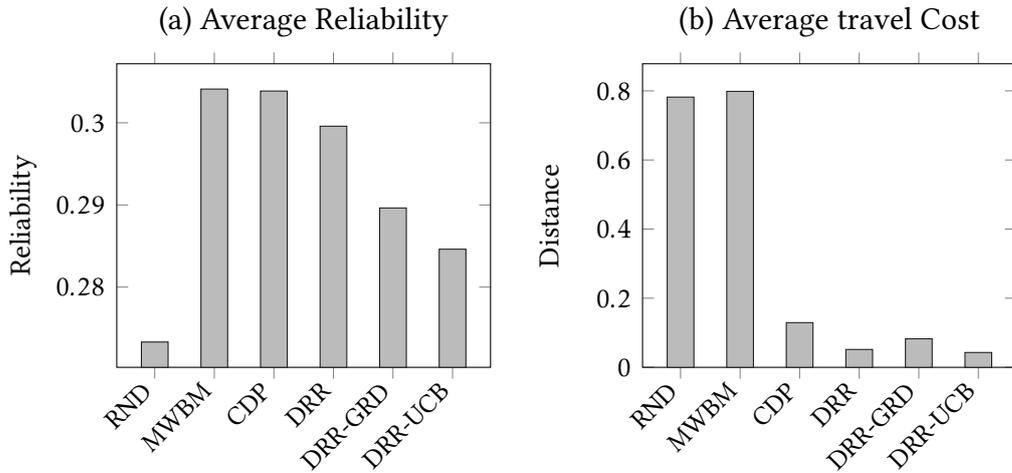


Figure 5.4: Comparison of algorithms on over multiple rounds of assignment on Foursquare data

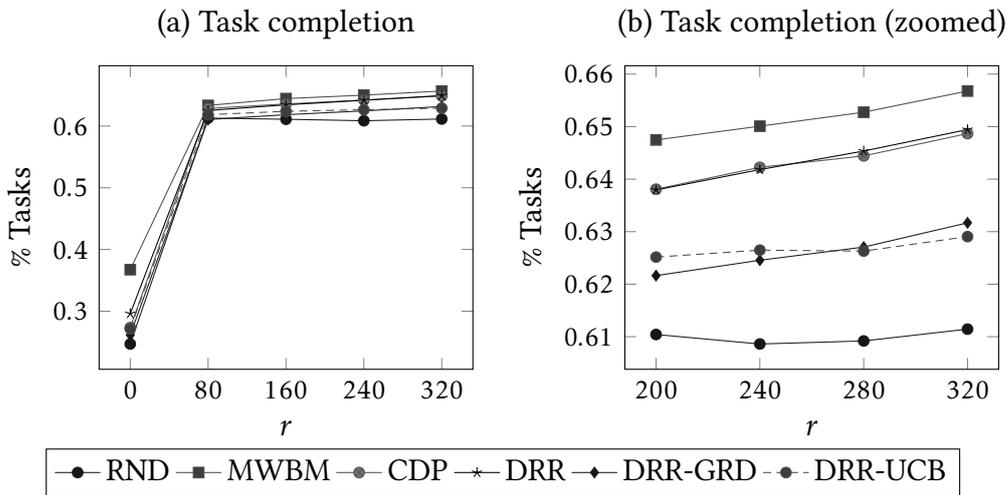


Figure 5.5: Comparison of algorithms on over multiple rounds of assignment on Foursquare data

levels, while the DRR-GRD and DRR-UCB algorithms achieve a 5-6% lower average reliability. Note that both CDP and DRR algorithms achieve high reliability with very small travel costs, as shown in Figure 5.4. In fact, the DRR algorithms and its learning based variants achieve even lower travel costs as compared to CDP, with a relative decrease of almost 50%.

Figure 5.5 shows the performance of algorithms in terms task completion rate

after each round. The task completion is the ratio of the cumulative number of completed tasks against the cumulative number of tasks appeared until round r . The MWBM achieves the best completions rate; however, the relative performance of other algorithms is slightly lower than MWBM. All algorithms achieve less than 70% completion rate with expiry set to 3 rounds for all tasks. Understandably, the low completions rate is due to the skewed distribution of worker reliabilities. The skewness of worker reliabilities is due to the fact that majority of users in New York dataset visited very few locations. This means that assignment algorithms might not have the desired effect on task completion rate if general worker population is not motivated to perform tasks.

5.5.3 Experiments on Uniform Data

This subsection compares the algorithms on synthetic dataset under various settings. The objective of these experiments is to establish the effects of individual parameters on algorithm performance.

Effects of Expiry Time

Figure 5.6 shows the effect of varying the task expiration time δ , on the performance of the algorithms with known reliabilities. The increase in the expiry does not affect the average reliability of tasks across all algorithms since it remains with a relatively stable range of all values (Figure 5.6a). A similar pattern is observed for the travel costs of algorithms when compared to the increase in task expiry times (Figure 5.6b). In terms of the relative performance of algorithms, the MWBM and CDP algorithms perform best in terms of reliability but they also perform worst in terms of the travel costs. By comparison, the DRR algorithm although performs 10% less in term of reliability and achieves 80% less travel

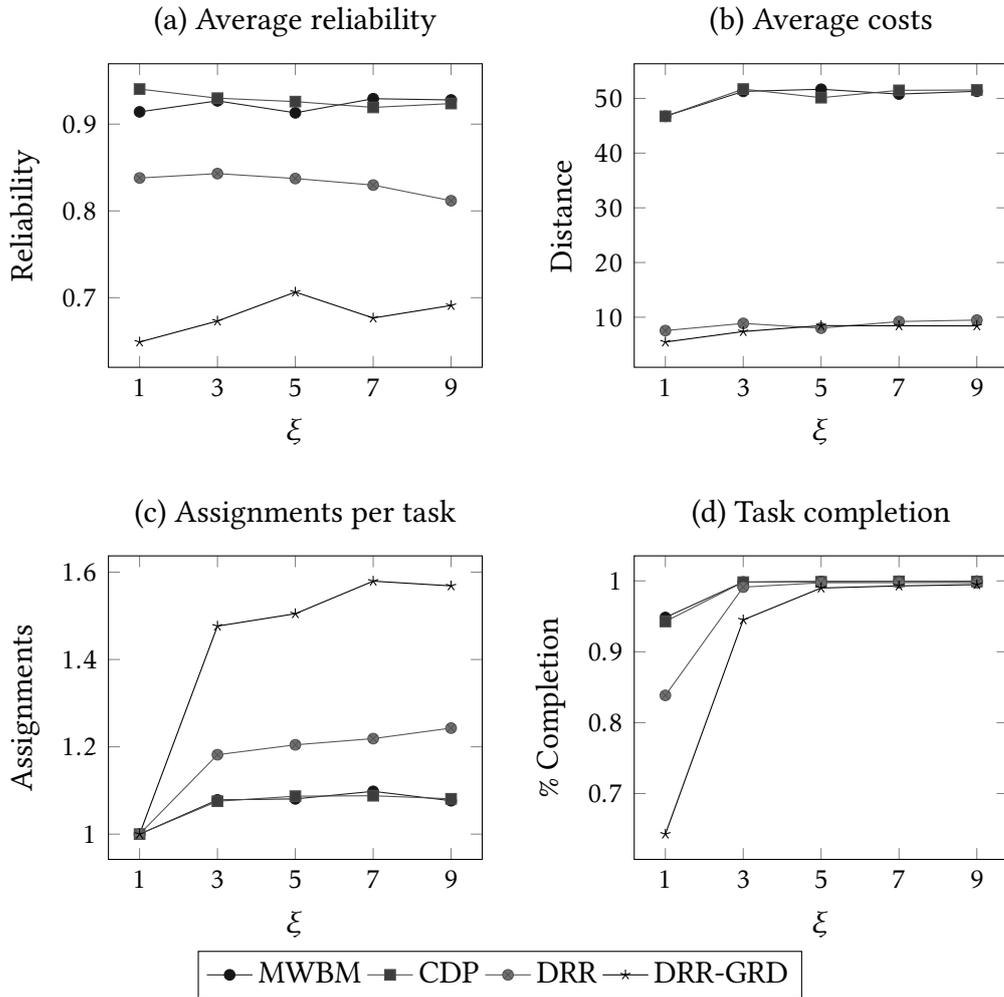


Figure 5.6: Effects of task expiry time (Uniform data)

cost. This demonstrates the effectiveness of the DRR algorithm against other algorithms with known reliabilities. The CDP algorithm fails in prioritizing small distances due to the uniqueness of solutions generated during the first phase of the algorithm; therefore, the second optimization phase faces limited choices with already high distances. The DRR algorithm optimizes the distance-reliability ratio; hence, achieving better results for both metrics.

Figure 5.6a and Figure 5.6b also show the performance of the DRR-GRD algorithm. Note that, the DRR-GRD algorithm does not have access to worker

reliabilities at the time of assignment; instead, it approximates the reliabilities based on worker reliabilities estimated over time. The exploration parameter was fixed at $\varepsilon = 0.2$ to control the percentage of rounds with the randomized assignment for the purpose of learning. The DRR-GRD algorithm achieves reliability within 10%-15% of the DRR algorithm while performing similarly on the travel costs. This supports the claim that DRR-GRD algorithm quickly estimates the worker reliabilities that are exploited for assignments in later rounds. Figure 5.6c and Figure 5.6d show the number of assignments per task and the percentage of completed tasks for all algorithms. Understandably, the DRR and DRR-GRD algorithm require more assignments per task to ensure task completion. In the worst case, the DRR-GRD algorithm requires no more than 1.6 assignments even when expiry times are more than 9 rounds for each task. Intuitively, the percentage completion of tasks reaches near maximum with expiry times of more than 3 rounds. The variation in expiry times does not affect the performance in terms of reliability and costs; however, higher expiry times lead to better completion rates due to repeated assignments.

Effects of Worker Reliability

Figure 5.7 shows the effects of the range of worker reliability on the comparative performance of the algorithms. The minimum value of the range was fixed at $\rho_{min} = 0.2$ while varying the maximum value $\rho_{max} \in \{0.5, 0.6, 0.7, 0.8\}$ of the reliability of workers. The reliability increases linearly with the increase in the range of worker reliabilities, for all workers (Figure 5.7a). Intuitively, the higher ρ_{max} results in more workers available with high success rates for tasks assigned to them. The distance reliability ratio based algorithms consistently achieve small travel costs with no effects due to changes in worker reliabilities (Figure 5.7b).

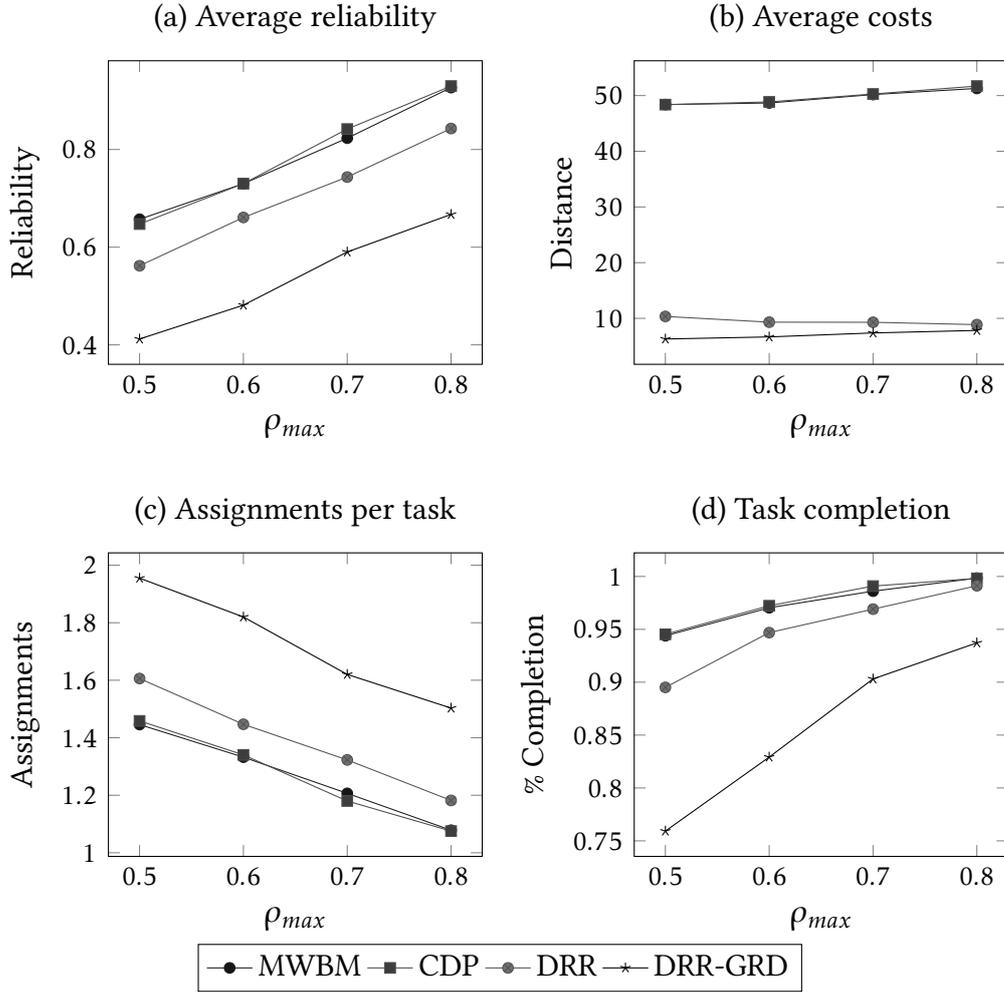


Figure 5.7: Effects of the range of worker reliability (Uniform Data)

As expected, the number of assignments per task decreases with an increase in ρ_{max} , indicating the availability of more reliable workers during each round. The relative performance, in terms of the number of assignments, decreases constantly for all algorithms (Figure 5.7c). Similar, to the task reliability the percentage of completed tasks increases with increase in the worker reliability range (Figure 5.7d). Note that, even when the reliability range of workers is very low $\rho_j \in [0.2, 0.5]$ the percentage of completed tasks is above 90% for algorithms with known reliabilities and above 75% for the algorithm with

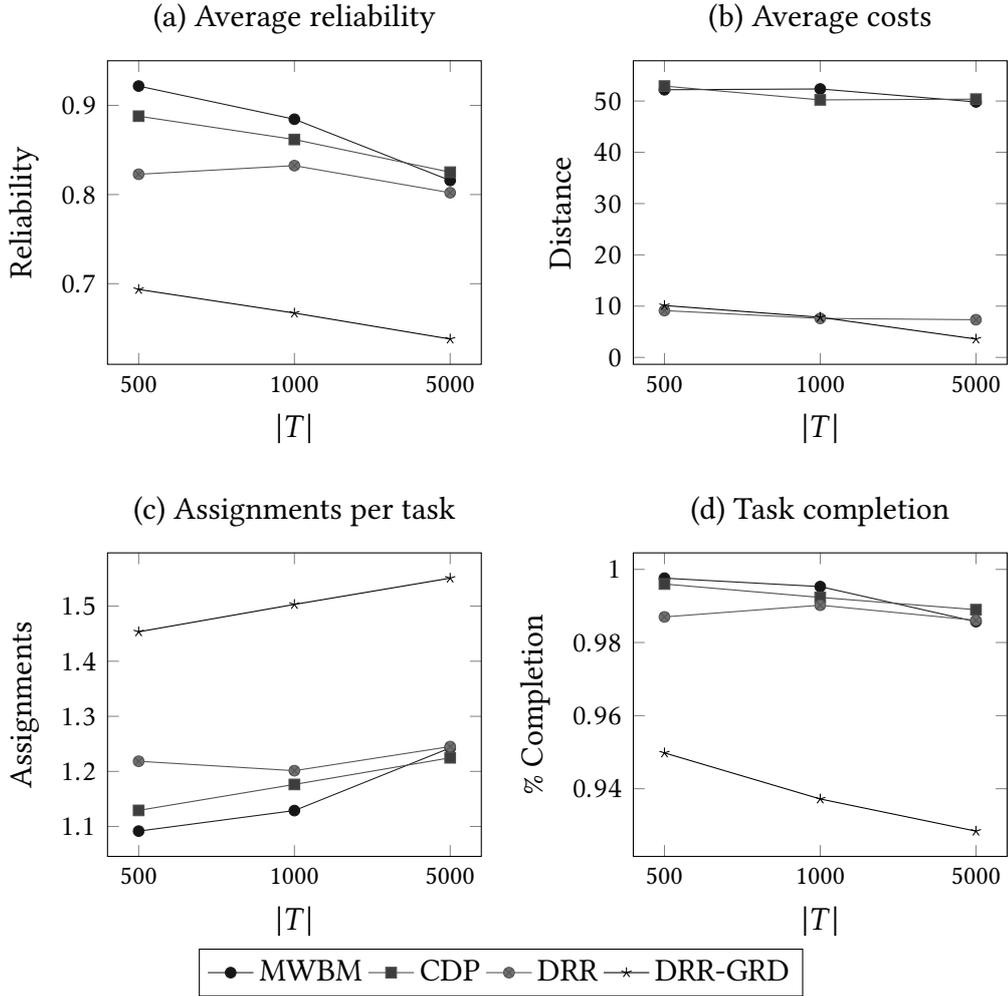


Figure 5.8: Effects of the number of tasks (Uniform Data)

estimated reliabilities. This indicates the effectiveness of the server-based assignment mode when workers are generally unreliable or reluctant.

Effects of Number of Tasks

Figure 5.8 shows the effects of varying the number tasks with parameter $|T|$; therefore, there are more tasks available for assignment during each round. The task reliability decreases slightly with increasing number of tasks (Figure 5.8a), while the average travel costs still remain the same even when there are ten times

more tasks during a round (Figure 5.8b). The decrease is primarily contributed due to low-reliability workers being selected for more tasks in each round. Alternatively, if the worker reliabilities were skewed towards the higher end of the range $[\rho_{min}, \rho_{max}]$ then the decrease might have been less. The performance of algorithms, in terms of the number of assignments per task, is the opposite of the reliability performance (Figure 5.8c). The percentage completion of tasks also falls when the number of tasks per round increases. The rate of decrease is strongest for the DRR-GRD algorithm, possibly due to the randomized assignment during the exploration phase.

Effects of Number of Workers

Figure 5.9 shows the effects of varying the number of workers with parameter $|W|$; such that, there are more alternatives available for assignment during each round. The average reliability increases due to more reliable workers being assigned, as the number of workers increase (Figure 5.9a). The relative rate of increase in task reliability is smaller for the DRR-GRD algorithm as compared to other algorithms. The MWBM and CDP algorithms achieve near maximum reliability for more than 500 workers. Note that, the CDP algorithm manages to reduce the average travel cost as the number of workers increases when compared to MWBM. This reduction is due to the close distance priority in the second stage of the algorithm while choosing an assignment from the feasible high-reliability assignments. The number of workers does not have a significant variation for the percentage completion of tasks (Figure 5.9d).

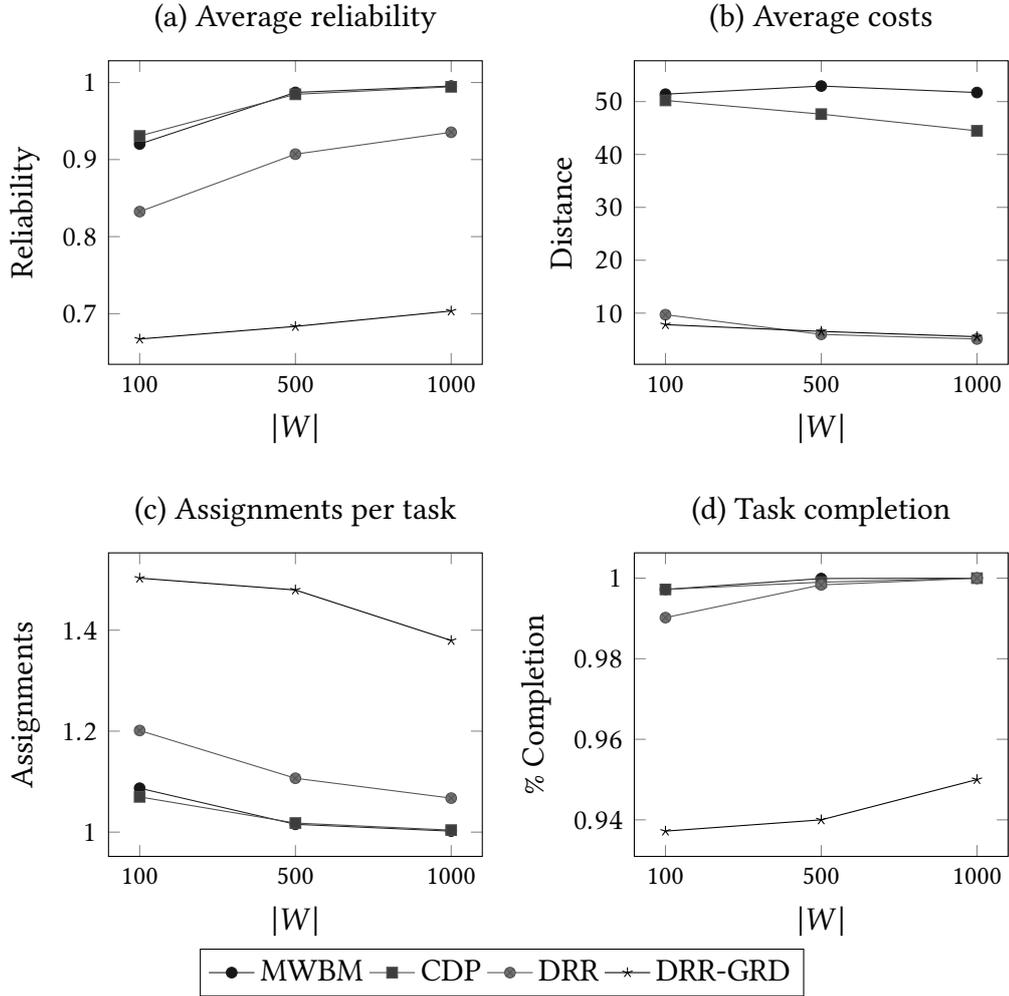


Figure 5.9: Effects of the range of the number of workers (Uniform Data)

Effects of Algorithm Parameters

Besides analyzing the comparative performances of algorithms, the relationship between the performance of algorithms and their parameters was also studied. The convergence parameter δ for the distance reliability ratio algorithms is based on the Newton's method. Figure 5.10 shows the comparison of the DRR, DRR-UCB and the DRR-GRD algorithm (using different values of exploration parameter $\varepsilon \in \{0.1, 0.2\}$). The average reliability of tasks does not change significantly when the convergence parameter δ takes on reasonably small values (Figure 5.10a). The

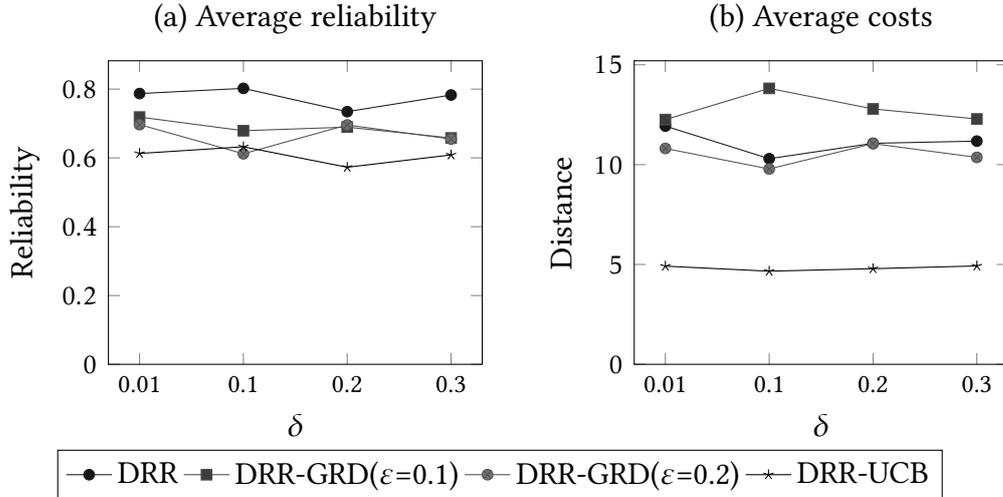


Figure 5.10: Effects of the δ parameter on DRR algorithm (Uniform Data).

Table 5.3: Comparison of algorithms in terms of execution time (seconds) on Uniform dataset.

Algorithm	Execution Time (seconds)
MWBM	0.66 \pm 0.46
CDP	2.53 \pm 0.59
DRR	0.72 \pm 0.45
DRR-GRD	1.83 \pm 1.01
DRR-UCB	0.95 \pm 0.62

DRR-UCB algorithm achieves performance similar to the DRR-GRD algorithm in terms of the average reliability, specifically for $\delta = 0.1$. The DRR-UCB achieves almost half of the travel costs achieved by other algorithms. These results establishes the relative superiority of DRR-UCB algorithm which does not require any parameter to control exploration. The UCB approach is also known to perform best asymptotically i.e. with an infinite number of rounds [190].

Table 5.4: Comparison of algorithms in terms of execution time (seconds) on Uniform dataset.

Algorithm	$ T = 500$		
	$ W = 50$	$ W = 100$	$ W = 200$
CDP	0.409 \pm 0.10	1.524 \pm 0.62	5.595 \pm 2.59
DRR	0.188 \pm0.12	0.564 \pm0.49	2.411 \pm2.78
DRR-UCB	0.187 \pm 0.10	0.707 \pm 0.79	4.087 \pm 4.59
	$ T = 1000$		
	$ W = 50$	$ W = 100$	$ W = 200$
CDP	0.726 \pm 0.15	3.801 \pm 1.01	14.338 \pm 5.94
DRR	0.526 \pm0.15	2.774 \pm1.21	11.925 \pm6.38
DRR-UCB	0.625 \pm 0.16	3.069 \pm 1.33	16.384 \pm 10.09
	$ T = 2000$		
	$ W = 50$	$ W = 100$	$ W = 200$
CDP	2.184 \pm0.89	9.318 \pm 1.31	49.560 \pm 14.87
DRR	2.481 \pm 1.41	8.356 \pm1.35	42.699 \pm10.50
DRR-UCB	5.842 \pm 3.95	10.108 \pm 1.73	60.629 \pm 17.35

Time Performance Comparisons

The running times of proposed approaches are also reported while fixing the number of rounds $R = 50$, the number of tasks $|T| = 500$, and the number of workers $|W| = 100$. Table 5.3 lists the mean execution time (in seconds) for a round, as well as the standard deviation. As expected, the MWBM algorithm performs best in terms of execution. The results also establish the relative performance gain of the DRR approach as compared to the CDP approach. Although both approaches rely on the Hungarian algorithm as the base assignment approach, the parametric approach of DRR outperforms the two-phased approach of CDP. The open source GLPK solver for the second phase of CDP with linear programming results in higher execution times. Using commercial solvers such

as CPLEX or GUROBI can improve the relative performance of CDP [191]. The DRR-GRD algorithm takes more time when compared to the DRR algorithm due to the extra enumeration through feasible solutions when early estimates of worker reliability are similar. By comparison, the DRR-UCB algorithm performs similarly to the DRR algorithm in terms of execution time. The relative degradation of DRR-GRD algorithm, against the DRR-UCB algorithm, can attribute to the dense costs matrices generated by the initial random exploration.

Table 5.4 studies the scalability of the CDP, DRR, and DRR-UCB algorithms against a different number of tasks and workers. The experiments were conducted with $R = 50$, $|T| \in \{500, 1000, 2000\}$, $|W| \in \{50, 100, 200\}$, and other parameters with default values. As highlighted in bold font, the DRR algorithm performs the best in general. Interestingly the DRR-UCB algorithm performs the worst when the number of tasks and workers are high. This could be due to the extra time taken by the DRR subroutine to converge.

5.5.4 Experiments on Skewed Data

This section discusses the experimental evaluation of proposed algorithms on skewed distributions. Give the range of worker reliabilities $[\rho_{min}, \rho_{max}]$, the average reliabilities for worker agents were sampled from Normal distribution i.e. $\rho_j \sim \mathcal{N}((\rho_{max} - \rho_{min})/4 + \rho_{min}, (\rho_{max} - \rho_{min})/4)$. Performance results of algorithms under various experimental settings as follows.

Effects of Expiry Time

Figure 5.11 shows the effect of varying the task expiration time δ , which is similar to the results for the Uniform distribution of worker reliabilities. In general, the average reliability is lowered due to the smaller values of worker reliabilities

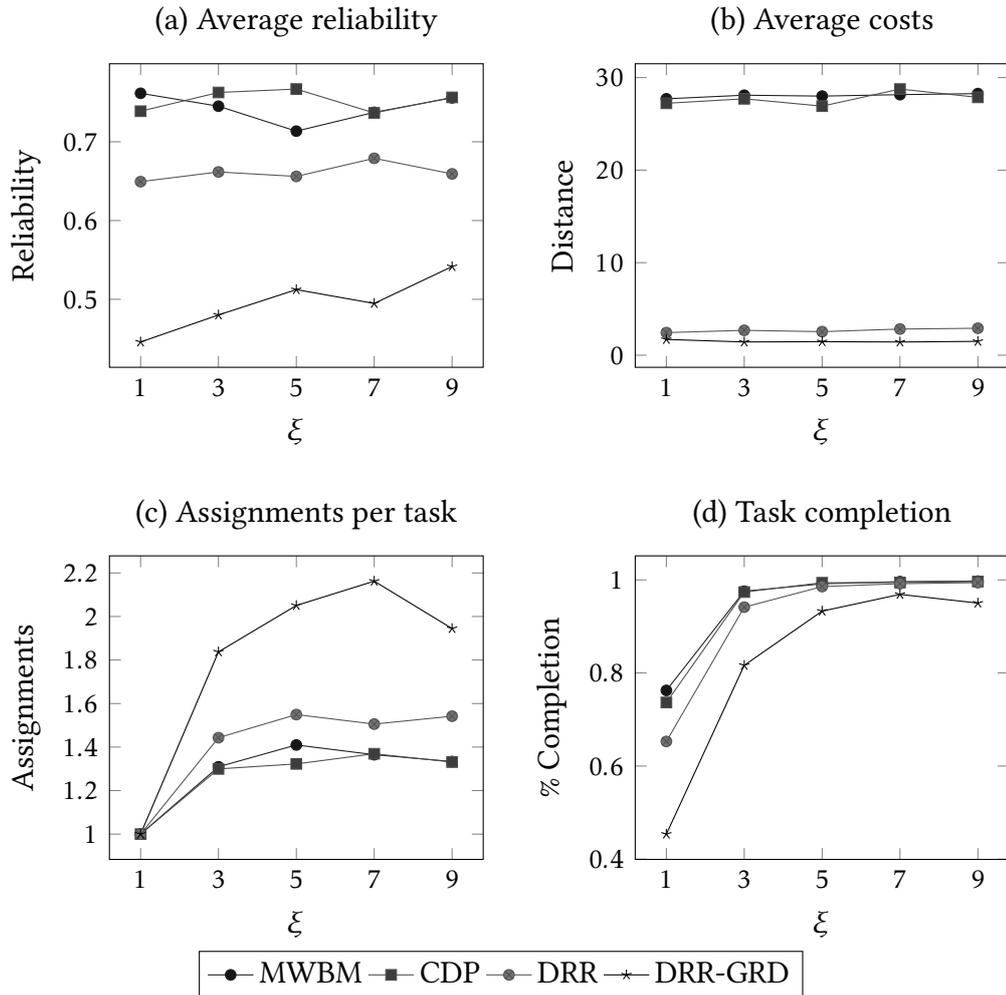


Figure 5.11: Effects of task expiry time (Skewed data)

(Figure 5.11a). The DRR-GRD algorithm improves in terms of average reliability per task as the expiry times increases. Algorithms based on DRR approach perform significantly better in terms of the travel costs (Figure 5.11b). Figure 5.11c shows the number of assignment per task for each algorithm. Understandably, the overall performance of all algorithms is degraded relative to the results of the Uniform distributions. Otherwise, the pattern of algorithmic performance remains the same. Figure 5.11d shows the percentage of tasks completed for all algorithms. Apart for DRR-GRD, all algorithms reach new maximum completion rate with

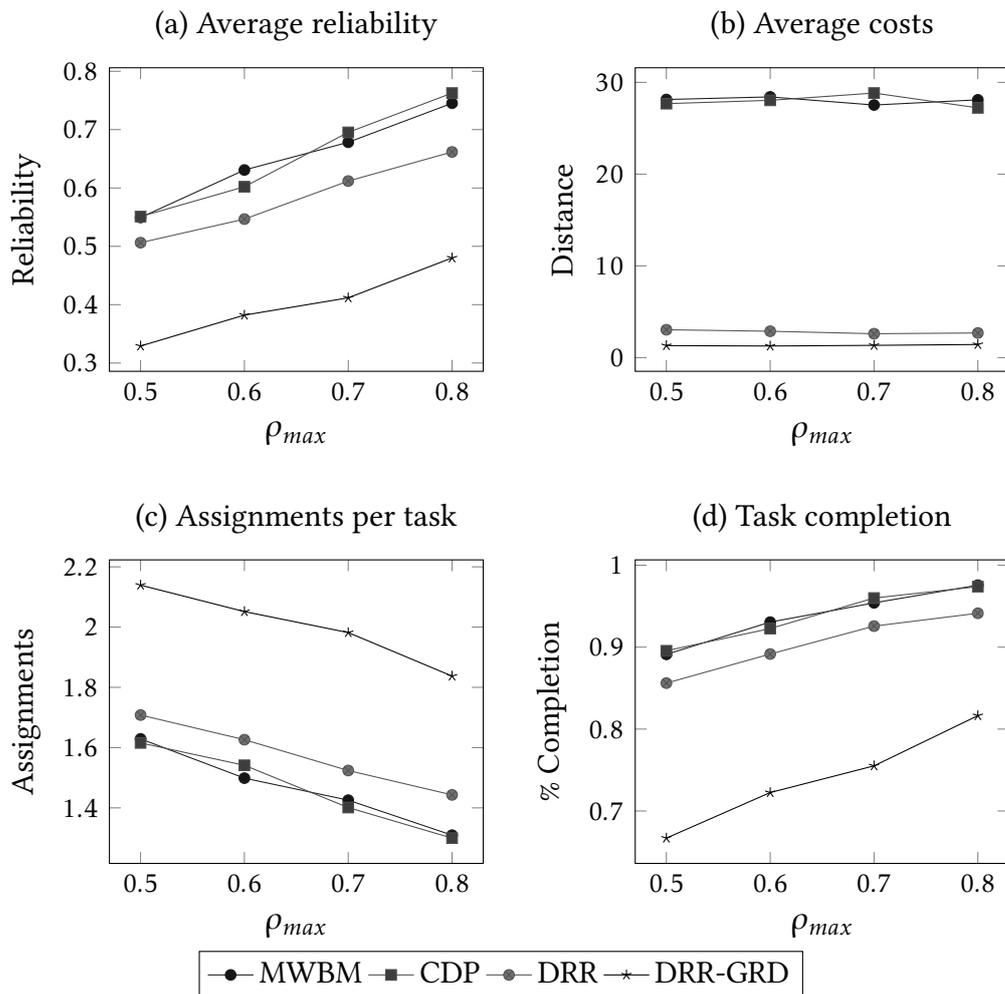


Figure 5.12: Effects of the range of worker reliability (Skewed Data)

more than 3 rounds of expiry time.

Effects of Workers' Reliability

Figure 5.12 shows the effects of the range of worker reliability on the comparative performance of algorithms. The reliability increases linearly with the increase in the range of worker reliabilities, for all workers (Figure 5.12a). The DRR based algorithms consistently achieve small travel costs with no effects due to the changes in worker reliabilities (Figure 5.12b). The relative performance, in terms of

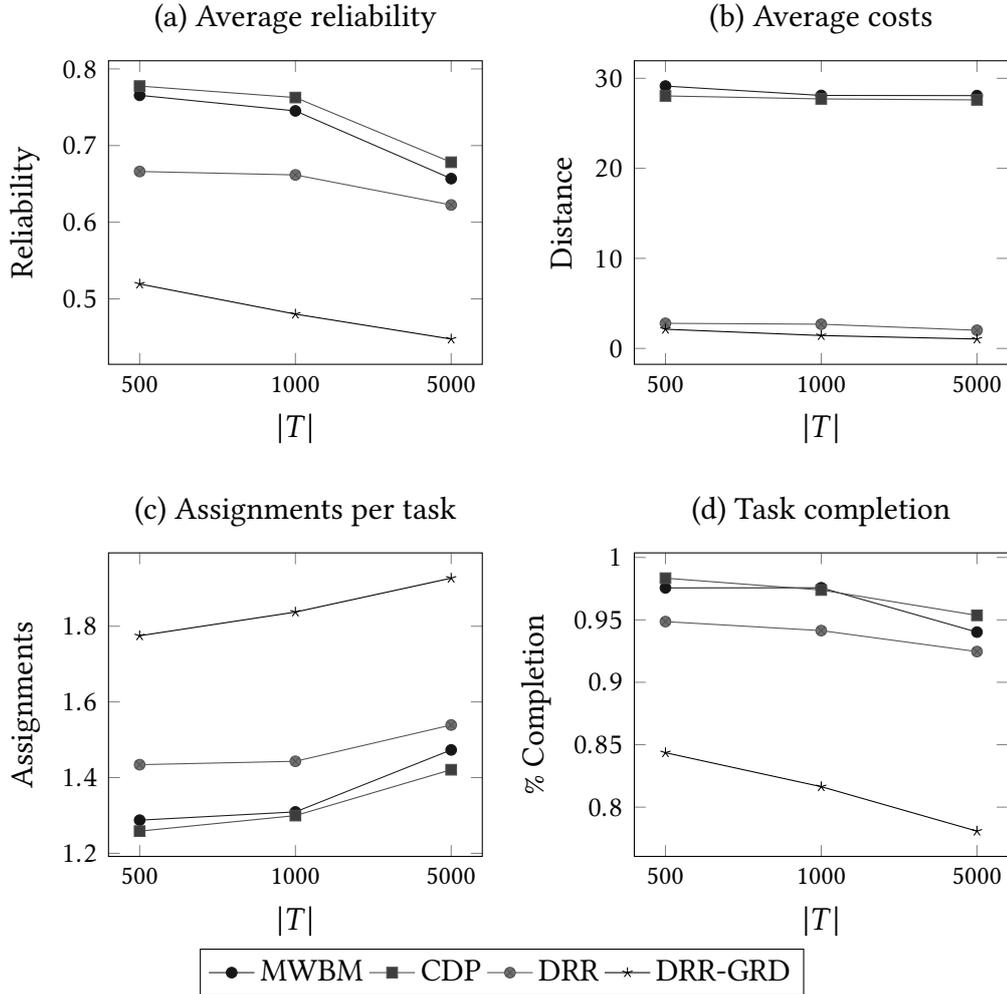


Figure 5.13: Effects of the number of tasks (Skewed Data)

the number of assignments, decreases constantly for all algorithms (Figure 5.12c). Similar, to the task reliability the percentage of completed tasks increases with increase in worker reliability range (Figure 5.12d).

Effects of Number of Tasks and Workers

Figure 5.13 shows the effects of varying the number of tasks with parameter $|T|$; therefore, there are more tasks available for assignment during each round. The task reliability decreases with increasing number of tasks (Figure 5.13a), where

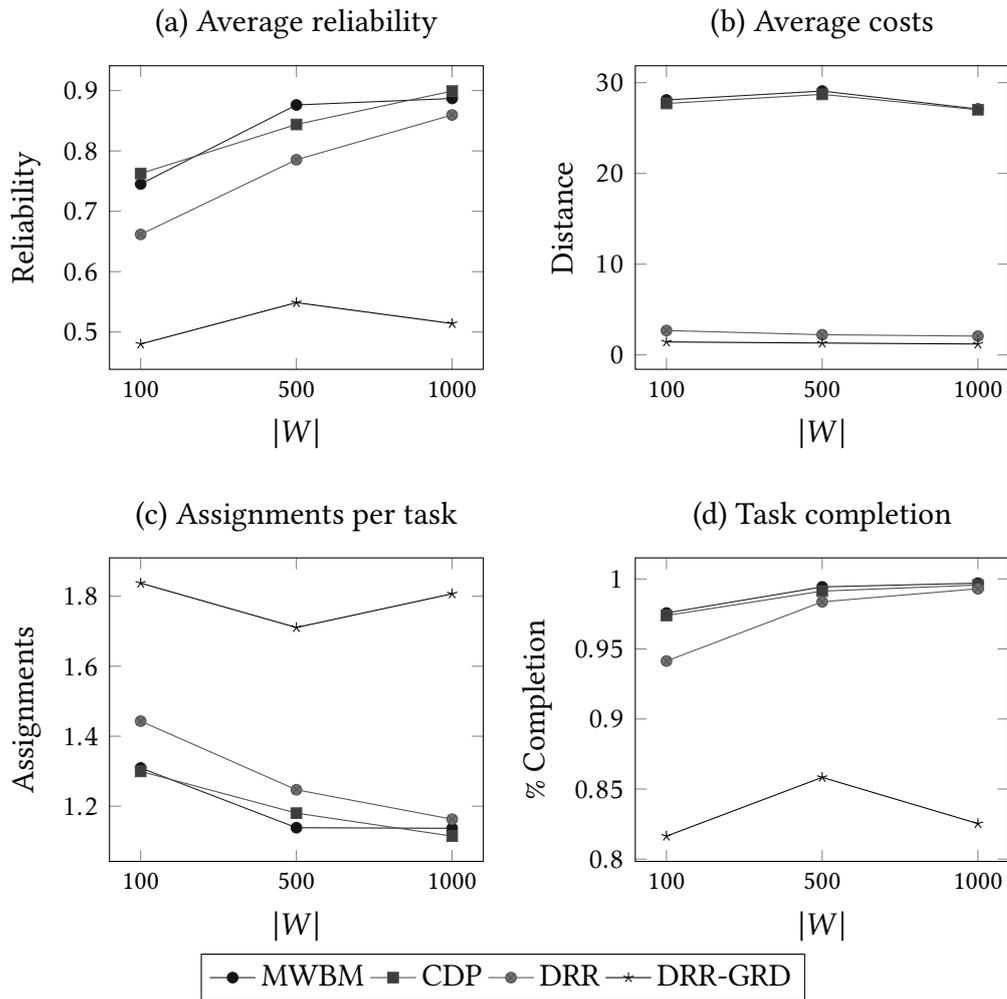


Figure 5.14: Effects of the range of the number of workers (Skewed Data)

the MWBM and the CDP approaches show the highest decrease for $n = 5000$. The average travel costs still remain the same even when there are ten times more tasks during a round (Figure 5.13b). The performance of algorithms in terms of the number of assignments per task is the opposite of the reliability performance (Figure 5.13c).

Figure 5.14 shows the effects of varying the number of workers with parameter $|W|$; such that, there are more alternatives available for assignment during each round. The average reliability of tasks increases due to more worker being assigned

with higher reliability, as the number of workers increases (Figure 5.14a). The number of workers does not have a significant effect on the percentage completion of tasks (Figure 5.14d).

5.6 Discussion

This section discusses the implications of the comparative performance results of the algorithms discussed in the chapter. The discussion is focused on how the MC-MRA problem and its associated algorithm address the research requirements. Furthermore, the limitations of the proposed approach are also discussed.

Formulation of Optimization Objectives

This chapter presents three approaches to addressing the MC-MRA problem. The CDP and DRR approaches aim to optimize the reliability as well as the travel costs. The proposed DRR approach is based on the linear-fractional programming formulation of the MC-MRA problem. The linear-fractional programming approach for optimization is a generalized case of the linear assignment problem; therefore, the proposed approach provides a more generalized solution to the MC-MRA problem. When travel costs sum to a unit in a round the DRR equals the MWBM approach. Other cost-reliability ratio assignment problems, in crowdsourcing, can also be formulated following the similar approach. Since the approach transforms the fractional optimization objective to an equivalent linear optimization objective, any existing algorithm for the linearized solution can be used to solve the problem. Subsequently, the computational complexity is dominated by the problem size and the computational complexity of the algorithm used for the linearized solution.

The MC-MRA problem presented in this chapter assumes one response

constraint per task, which means that the each task is assigned at most one worker. Consideration of multiple workers per tasks either for redundancy or diversity is a possible extension; however, such constraints increase the complexity of problem requiring approximate solutions. As a simple extension, each task that requires multiple responses can be considered as multiple instances of the same task. Similarly, a worker with more than one task capacity can be modeled as multiple instances of the same worker. The performance of the approaches proposed here needs to be validated with more complex constraints, such as the task diversity [38] or budget constraints [47]. Even worker constraints such as capacity or time duration can further enhance the assignment process. Note that, a linear-fractional program can be easily transformed into an equivalent linear program as far as the constraints matrix is unimodular on the right-hand side [192].

Online Learning Heuristics

Besides the DRR approach, the other main contribution of this chapter is addressing the explore-exploit trade-off for the estimation of worker reliabilities in the face of uncertainty. The real world is uncertain; therefore, it is essential to incorporate appropriate learning capabilities in spatial crowdsourcing process. The real world is also dynamic; hence, it is useful to update the learning in spatial crowdsourcing process. The proposed approach for the dynamic estimation of workers' reliabilities is based on the combinatorial bandits, which deals with uncertain decision making over time. Among the two learning approach proposed here for MC-MRA problem, the greedy approach performs reasonably well in comparison to the deterministic algorithms. However, the greedy approach suffers from poor short-term performance as the learning is scheduled in first few rounds. To overcome this issue, an alternative greedy approach based on semi-uniform

learning is proposed in literature where randomized exploration is scheduled during any round based on a probability parameter [64]. Both of these greedy approaches suffer from randomization error. By comparison, the upper confidence bound approach performs better in the long-term which underlines its utility for long running systems with dynamic worker populations.

Both DRR-GRD and DRR-UCB algorithms estimate the reliabilities of workers in each round of assignment. The exploration is based on the estimates plus an adjustment term, that depends on the number of times a worker has been chosen previously. This means that the more a worker is explored the more accurate the estimates are. Instead of estimating distribution parameters, the actual Bernoulli distribution for assignment successes of a worker can be estimated using Bayesian methods [193]. Similarly, probability matching approaches try to match the number of times a worker is chosen with the probability of that worker being the most reliable [63, 193].

Besides the feedback of task completion, other contextual information helps improve the learning process by providing additional data points. Existing approaches based on the multi-armed bandit problem tend to exploit this information for the purpose of improving the estimates of future rewards [66]. In the current formulation the locations of tasks and workers serve as the contextual information; however, this information is only used to optimize travel costs. Contextual algorithms [32] exploit side information including locations and other information such as task types, worker demographics, location properties, etc. The utility of such approaches in combinatorial optimization setting is yet to be investigated for spatial crowdsourcing.

Reliability in Spatial Crowdsourcing

The worker reliability as defined in this paper depends on both task and worker. This definition of reliability does not include the effects of external factors on the probability of success. Furthermore, the exact definition of a successful task completion is left to designers of spatial crowdsourcing system. For instance, a platform designer or requester might define the completion of a task in terms of the resolutions of the uploaded pictures. On the other hand, another requester might define the success in terms of a presence of specific objects in uploaded pictures. In this sense, the reliability of a worker might be thought of as the rate of successful task completion. Therefore, it is intuitive to maximize the reliability through intelligent assignment decisions.

The probabilistic definition of reliability is fairly recent in spatial crowdsourcing literature. The reliability is defined as the probability of task being completed by an assigned worker. A recent proposal defined probability as decreasing function of the distance between task and worker [37]. Additionally, it was assumed to be independent of task and worker. Another proposal defined fixed probability as the reliability of a worker [38]. The reliability in terms of the frequent routes visited by a worker was also considered for WST-based spatial crowdsourcing [26]. In general, all these definitions of reliability are independent of tasks. Instead, this paper assumes worker reliability being dependent on the assigned task.

Within the literature on non-spatial crowdsourcing, the reliability of a worker is defined in terms of the accuracy of responses provided by the worker. On one hand, the reliability of worker can be the closeness of response provided by a worker to the predicted values of a regression task [49]. On the other hand, it is the

reliability ratio of correct responses against all responses provided by a worker for binary classification tasks [52]. Domain specific definitions of worker reliability are common in the literature on non-spatial crowdsourcing.

Limitations & Strengths

One of major strength of the proposed approach is its applicability to more realistic spatial crowdsourcing scenario, as compared to existing task assignment approaches in spatial crowdsourcing. The platform being a participating agent in the multi-agent environment of spatial crowdsourcing, must adapt its behavior according to the outcomes of task assignments over time. In this regard, our proposed approach enable intelligent assignment decisions for optimizing spatial crowdsourcing, in the face of uncertainty. Our proposed approach is also applicable to other areas of expert and intelligent systems. It is specifically suited for application scenarios when optimization objective is to maximize cumulative rewards and minimize cumulative costs while assuming that the rewards are uncertain.

The adaptive task assignment approach faces limitations due to three underlying assumption. First, the multi-armed bandit formulation of the reliability approximation necessitates immediate observability of assignment outcomes. Second, it is assumed that the worker reliabilities are stochastic. This assumption ignores the strategic behavior of worker in responses of previously assigned tasks. Learning heuristics for adversarial outcomes of decisions have also been studied in the literature on multi-armed bandits. However, a detail investigation of such heuristics, for task assignment in spatial crowdsourcing, is out of the scope of this work. Finally, there could be situations when a worker might attempt an assigned task after the end of a round. Incorporating such delayed outcomes is another

limitation of this research work.

Assigning a chain of tasks to a worker such that assigned tasks are clustered within an area is also intuitive [20]. The DRR algorithms assign the nearest tasks to a worker over many rounds, which can also form a chain of tasks over time. However, a worker cannot plan travel path since future tasks are not revealed beforehand. Task chains can be considered in the case of multiple tasks per worker in a round. But this is a known hard problem and out of the scope of this work. Furthermore, it adds to the complexity of assignment decisions due to considerations of load balancing and social welfare. Social welfare entails that all workers are given the opportunity to perform tasks, instead of favoring worker with higher load capacity in a round. The goal of optimizing social welfare is to promote long-term engagement [16].

5.7 Chapter Summary

This chapter presented the minimum cost maximum reliability assignment problem in spatial crowdsourcing. This problem formalizes the spatial context and multi-criteria optimization requirements of adaptive task assignment. A distance reliability ratio based approach is proposed to address this problem. Experimental evaluation validates that the empirical performance of proposed approach. As compared to the baseline approaches, the proposed approach achieves high reliability and low travel costs.

The minimum cost maximum reliability assignment problem is extended for adaptive task assignment scenario. It formalizes the requirement of the explore-exploit trade-off for estimating unknown worker reliabilities. To address the trade-off, the distance reliability ratio approach is extended with two online

learning heuristics. The first semi-uniform approach achieves travel costs similar to the distance reliability ratio approach while achieving reliability within 15%. The second interval estimation approach achieves even better travel costs when compared to the semi-uniform approach. In general, the results suggest the online learning based assignment algorithms perform reasonably well even under varying conditions.

Chapter 6

Adaptive Assignment with Spatial Contextual Learning

Chapter 5 focused on adaptive assignment problem in spatial crowdsourcing under two simplifying assumptions. First, the worker's reliabilities over different tasks are non-identical Bernoulli trials. Second, the travel costs between tasks and workers are part of the optimization objectives. The intuition behind the second assumption was that assigning nearby tasks to workers would encourage task completion. The proposed algorithms used simple estimates of the worker reliabilities for making assignment decisions. In reality, the probability of the success of an assignment is not a simple coin toss. Teodoro et al. showed that the context of tasks and workers is important in workers' decisions to perform tasks [16, 20]. For instance, the socio-economic status of the task location influences the worker reliability [24]. In fact, workers chose not to perform tasks located in a low-income area of the city. It is clear that the assignment algorithms should also consider contextual information. So that the algorithms are able to learn the

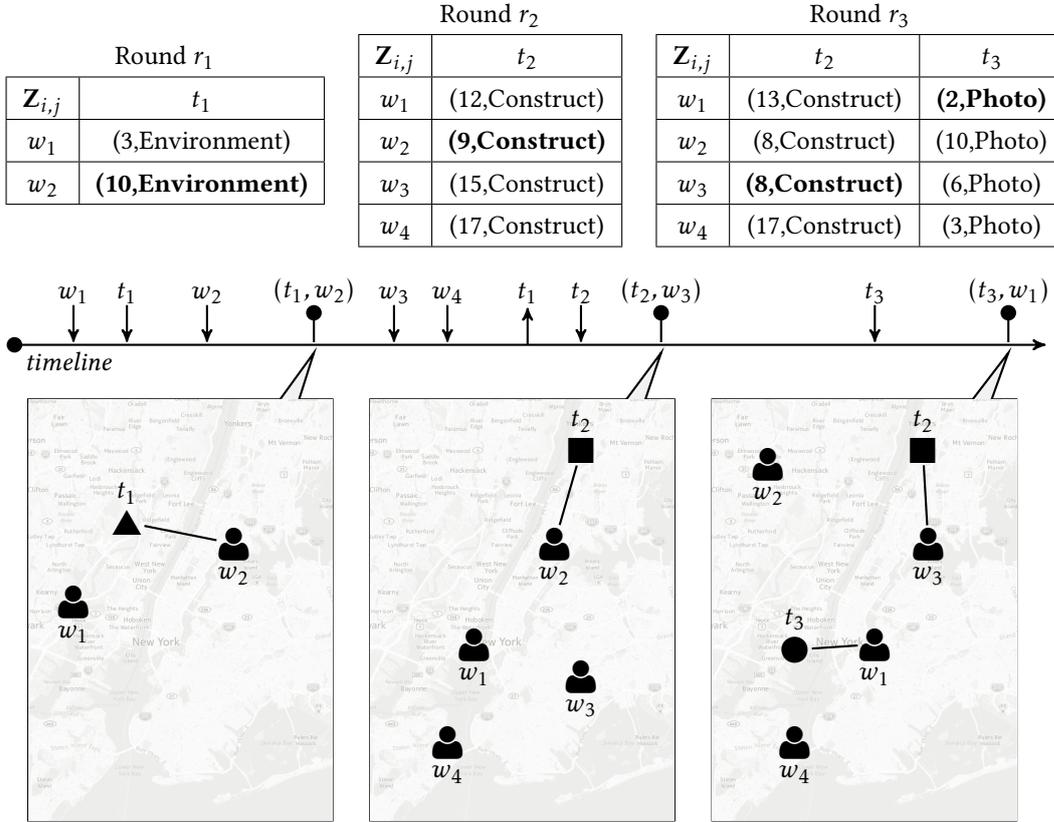


Figure 6.1: Three rounds of adaptive task assignment with contextual vectors.

influence of contextual variables on worker reliability.

Figure 6.1 motivates the need for adaptive task assignment based on contextual variables. In this scenario, the assignment algorithms have access to a contextual vector $Z_{i,j}$ for each task-worker combination. Each vector contains two variables: the first variable is the task type and the second variable is the distance between a task and a worker. Based on the observed knowledge assumption, the assignment algorithm must estimate the worker reliability. The estimation uses contextual variables and the outcomes of the previous assignment. This scenario highlights the three research requirements: explore-exploit trade-off, spatial context, and contextual learning.

This chapter introduces the contextual bandits approach for adaptive task

assignment with contextual learning in spatial crowdsourcing. The rest of the chapter is structured as follows. Section 6.1 extends the maximum reliability assignment problem from Chapter 4 with contextual learning. Section 6.2 analyzes the existing approaches to context-free learning. The analysis based on the literature on the bandit problem. Section 6.3 introduces the contextual bandits approach for assignment algorithm based on a linear regression model. Section 6.4 describes the adaptive task assignment algorithm that combines regression model and interval estimation. Section 6.5 presents an experimental evaluation of the proposed algorithms on real-world datasets. Section 6.6 discusses the implications of the results and Section 6.7 concludes the chapter.

6.1 Problem Description

Chapter 5 introduced the *minimum cost maximum reliability assignment* (MC-MRA) problem with probabilistic worker reliabilities and deterministic travel costs. The distances were treated as the contextual spatial information that was used to optimize the expected travel costs. In addition, no further assumptions were made about the availability of more contextual information at the time of assignment decisions. It is well-known that the success probability of an assignment is dependent on the context of tasks and workers [16, 24].

In general, there are two approaches to knowledge representation for adaptive task assignment in the literature: scalar and vector. In the scalar approach, as used in Chapter 5, the knowledge about a worker’s reliability is represented as a scalar value. Whereas the vector approach represents knowledge as a set of worker-specific coefficients that are used to generate estimates of reliability based on a weighted combination of contextual variables. The majority of existing research

on adaptive task assignment models the heterogeneous task types using the vector approach [51, 52]. This chapter develops a novel vector-oriented approach for the estimation of worker reliabilities. The proposed approach individualizes the linear regression models for workers. Such an approach allows the learning of worker-specific influences on assignment outcomes based on the contextual variables. The worker reliability is assumed to be a function of the contextual variables $\mathbf{Z}_{i,j}$, worker-specific coefficients \mathbf{U}_j , and random factors $e_{i,j}$:

$$p_{i,j} \sim f(\mathbf{Z}_{i,j}, \mathbf{U}_j, e_{i,j})$$

In general, the function f is unknown. However, it is common to assume that the function belongs to a class of probability-generating functions such as Logit and Probit. Furthermore, it is also reasonable to assume that the worker-specific coefficients define each individual worker's preferences of performing the tasks assigned to them. As described in Chapter 5, the overall objective is to design an algorithm that aims to maximize the reliability $P(r)$ over all rounds (see Equation 4.1).

Definition 6.1 (Maximum Reliability Assignment with Context). *The problem of maximum reliability assignment with context (MRA-C) is to assign workers to tasks such that the total reliability is maximized over all rounds i.e. $\max \prod_{r=1}^R P(r)$, while exploiting the contextual variables $\mathbf{Z}_{i,j}$.*

Similar to the original maximum reliability assignment problem, a global solution to the MRA-C problem is not feasible unless the assignment process is clairvoyant i.e. all information about rounds and probabilities is known beforehand. If the probability of success $p_{i,j}$ is known for each assignment, then the local optimization strategy based on the maximum weight bipartite matching

can be applied. As is the general case in this thesis, this chapter also focuses on the situations when $p_{i,j}$ are unknown and must be estimated using the contextual variables and observed assignment outcomes over time. Online learning is one of the fundamental requirements of the adaptive assignment problem as introduced in the Chapter 4, and the MRA-C problem is an instantiation of the adaptive assignment problem.

6.2 Baseline Approaches

Online learning in MRA-C is formulated according to the combinatorial bandits framework, which necessitates that the assignment algorithms must address the explore-exploit trade-off. In general, the existing research on combinatorial bandits employs heuristics to learn from previous outcomes and generate estimates for expected values of parameters of interest, i.e. worker reliability. The following list summarizes the high-level categories of these heuristics for online learning:

- The semi-uniform heuristics alternate between learning and optimization. The simplest example of an algorithm based on the semi-uniform heuristics is the ϵ -Greedy algorithm [63]. For the adaptive assignment, such an algorithm chooses random assignments with probability ϵ . Otherwise, the optimal assignments are chosen based on the existing estimates. The parameter ϵ controls the percentage of rounds dedicated to randomized assignments for the purpose of learning. Understandably, high values of ϵ result in more accurate estimates of success probabilities. Chapter 5 proposed the DRR-GRD algorithm for the minimum cost maximum reliability assignment problem that is also based on a semi-uniform heuristic for learning. In this case, the random assignments are scheduled in first

few rounds of assignment instead of being randomly scheduled. This two-phased approach of online learning is referred to as the ϵ -First algorithms in the literature [64].

- The probability matching heuristics make choices by estimating the likelihood of each choice being optimal. The Softmax algorithm employs one such heuristic based on the Boltzmann distribution [40]. For the adaptive assignment, such an algorithm chooses assignments with a probability that is proportional to the observed successes of assignment in previous rounds. It takes τ parameter, called the *temperature*, that moderates the degree to which the optimal choices are preferred. Higher values of τ mean more exploration of workers.
- The interval estimation based heuristics represent an optimistic approach by selecting less explored choices. The UCB algorithm is one of the most popular in general literature of the bandit problem, primarily due to its easier performance analysis [64]. For adaptive task assignment, this heuristic chooses the assignments that have the highest confidence bound on the expectation of worker reliability. Chapter 5 proposed the DRR-UCB algorithm which employs the UCB heuristic for the minimum cost maximum reliability assignment problem. A parameterized version of the UCB algorithm, known as the UCB2, uses the parameter λ to control the level of optimism by weighting the confidence interval.

This chapter considers the assignment algorithms based on the representative heuristics for each of these categories. The maximum weight bipartite matching serves as a local optimization strategy conjunction with ϵ -Greedy, Softmax, and UCB2 heuristics, as the baseline algorithms. All of the baseline algorithms are

categorized as the *context-free* algorithms. This is due to their entire reliance on the observed outcomes of previous assignments for the assignments in next round.

6.3 Contextual Bandits

This chapter formalizes the MRA-C problem, as inspired by the contextual bandits framework [66]. In this formalization, the pool of available workers W_r is considered as the multi-armed bandit such that each worker corresponds to an arm. An assignment $x_{i,j}$ of a task t_i to the worker w_j is equivalent to playing an arm and the outcome of the assignment $y_{i,j}$ is the resulting reward. For each dynamically arriving task, choosing a worker with the highest expectation of $y_{i,j}$ leads to the maximization of the assignment success. In the MRA-C problem, the spatial features of task and workers are revealed before assignment decisions. For instance, the location attributes of task and worker can be considered the contextual information that might be useful for making assignments.

The assignment process proceeds in a set of discrete rounds R . During each round r the following steps are performed:

1. The process considers the current incomplete tasks T_r and the pool of available workers W_r . Each pair of a task t_i and a worker w_j has an associated d-dimensional vector $\mathbf{Z}_{i,j}$. The vector $\mathbf{Z}_{i,j}$ contains the contextual variables defined according to the spatial and non-spatial attributes of the task and workers.
2. The algorithm calculates the reliability score matrix $[\hat{p}_{i,j}]^{n,m}$. Each score variable $\hat{p}_{i,j}$ approximates the reliability of worker w_j for the task t_i .
3. The algorithm chooses an assignment matrix $[x_{i,j}]^{n \times m}$ for the current round

based on the contextual variables and learning from the outcome of the previous rounds. The tasks are assigned to workers according to the assignment set.

4. The process waits until the end of the round and observes the outcomes $y_{i,j}$ for each assignment $x_{i,j}$.
5. The assignment algorithm updates its estimates for worker-specific coefficients \mathbf{U}_j of the individualized reliability model.

Figure 6.2 illustrates the workflow described above; in addition, it highlights the key steps where the contextual bandits approach improves the combinatorial bandits approach. A good assignment algorithm for the MRA-C problem aims to maximize the total reliability over all rounds i.e. $\max \prod_{r=1}^R P(r)$. The next section introduces one such algorithm that exploits the relationship between contextual variables and assignment outcomes to maximize reliability.

6.4 Estimating Reliabilities with Contextual Variables

The main source of uncertainty in the MRA-C problem is the unknown worker reliability $p_{i,j}$. An assignment algorithm could follow a simplistic approach by assuming that each worker has a fixed behavior of task acceptance, i.e., $p_{i,j} = \rho_j$ for all $t_i \in T_r$. In such a case, the success rate of each worker can be modeled as a Binomial process with parameter ρ_j . Given the contextual variables $\mathbf{Z}_{i,j}$, it is important to exploit this information to improve the assignment algorithm. The adaptive task assignment proposed in this chapter assumes that the worker

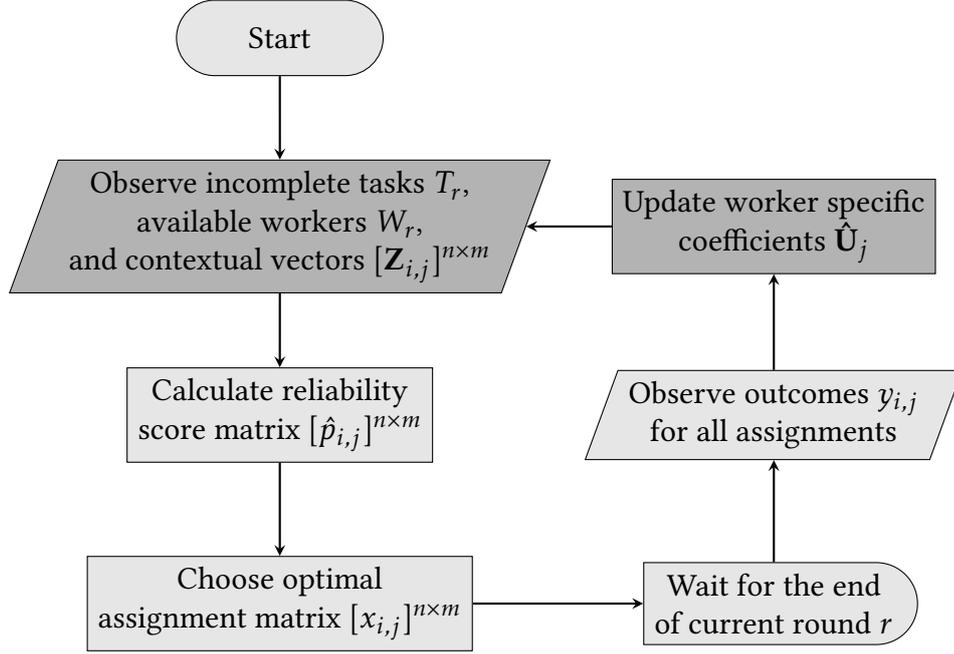


Figure 6.2: Work flow of the adaptive assignment problem with contextual bandits.

reliability is a linear function of contextual variables. Therefore, the expectation of assignment success is defined as below

$$\mathbf{E}[y_{i,j} | \mathbf{Z}_{i,j}] = \mathbf{U}_j^\top \mathbf{Z}_{i,j} + e_{i,j} \quad 6.1$$

As noted earlier, the vector \mathbf{U}_j defines the unknown vector of worker-specific coefficients that need to be learned from observed outcomes. On one hand, the objective of the assignment algorithm is to choose assignments with the highest expectation of $y_{i,j}$. On the other hand, the algorithm must estimate the coefficients \mathbf{U}_j for each worker by assigning tasks to unexplored workers. One heuristic to address this learning against optimization is the *linear upper confidence bound* (LinUCB), that uses ridge regression over contextual variables and outcome variables for learning the coefficients on the linear model [66, 194].

Algorithm 4 lists the LinUCB heuristic for the MRA-C problem. The algorithm

Algorithm 4 The SpatialUCB algorithm

Require: λ, R, T, W

```

1: for  $r \leftarrow 1$  to  $R$  do
2:    $T_r \leftarrow \text{Active}(T)$                                 {Set of incomplete tasks}
3:    $W_r \leftarrow \text{Available}(W)$                         {Set of available workers}
4:   for all  $w_j \in W$  do
5:     if  $w_j$  is new then
6:        $\mathbf{V}_j \leftarrow \mathbf{I}^{d \times d}$                     {Initialize covariance matrix}
7:        $\hat{\mathbf{U}}_j \leftarrow [0]^d$                         {Initialize coefficient vector}
8:     end if
9:   end for
10:   $n \leftarrow |T_r|$ 
11:   $m \leftarrow |W_r|$ 
12:   $[\mathbf{Z}_{i,j}]^{n \times m} \leftarrow \text{Context}(T_r, W_r)$       {Observe d-dimensional context vectors}
13:   $[\hat{p}_{i,j}]^{n \times m} \leftarrow \text{LinUCB}(\lambda, [\mathbf{V}_j]^m, [\hat{\mathbf{U}}_j]^m, [\mathbf{Z}_{i,j}]^{n \times m})$     {Estimated scores}
14:   $[x_{i,j}]^{n \times m} \leftarrow \text{Hungarian}([\hat{p}_{i,j}]^{n \times m})$     {Assignment matrix}
15:   $\text{Assign}(t_i, w_j) \quad \forall x_{i,j} > 0$                 {Assign tasks}
16:   $\text{Wait}(\tau)$                                         {Wait for the end of round}
17:  for all  $(x_{i,j} > 0)$  do
18:     $y_{i,j} \leftarrow \text{Complete}(t_i, w_j)$                 {Completion indicator}
19:     $\mathbf{V}_j \leftarrow \mathbf{V}_j + \mathbf{Z}_{i,j} \mathbf{Z}_{i,j}^\top$         {Update covariance matrix}
20:     $\hat{\mathbf{U}}_j \leftarrow \hat{\mathbf{U}}_j + y_{i,j} \mathbf{Z}_{i,j}$         {Update coefficient vector}
21:  end for
22: end for

```

requires one parameter λ that controls the intensity of the optimism when calculating reliability scores. For each round, the algorithm starts by finding the sets of incomplete tasks T_r and available workers W_r (Lines 2-3). The algorithm stores a covariance matrix \mathbf{V}_j and a vector $\hat{\mathbf{U}}_j$ for observed responses; both of which are initialized to defaults when a new worker arrives in the system (Lines 5-8). The vector of context variables is observed for each possible assignment

between task t_i to worker w_j (Line 12). The next step calculates the reliability scores $p_{i,j}$ for each assignment that approximates the actual reliabilities $p_{i,j}$ (Line 13). To be more precise, the reliability score for an assignment is defined according to the LinUCB heuristic:

$$\hat{p}_{i,j} = \hat{\mathbf{U}}_j^\top \mathbf{z}_{i,j} + \alpha \sqrt{\mathbf{z}_{i,j}^\top \mathbf{V}_j^{-1} \mathbf{z}_{i,j}}$$

where the estimated worker-specific coefficients are defined as $\mathbf{U}_j = \mathbf{V}_j^{-1} \hat{\mathbf{U}}_j$, and the second term quantifies the upper bound on the confidence interval for the estimated reliability score of the assignment. Given the scores matrix, the algorithm employs the Hungarian algorithm to calculate the optimal assignment matrix (Line 14). The algorithm then assigns tasks to workers and waits until the end of the current round (Lines 15-17). At the end, the algorithm observes the outcomes of assignments and updates the covariance matrix and the responses vector (Lines 18-22).

The SpatialUCB algorithm has some nice properties. First, the computational complexity is dominated by the complexity of the sub-routine used to implement the Hungarian algorithm in Line 14. Among the known implementations of the Hungarian method, the worst known complexity is cubic in terms of the size of score matrix [141]. The complexity of SpatialUCB is linear in the number of tasks and the number of workers; moreover, it is cubic in the number of contextual variables. Second, the algorithm also accommodates the dynamic pool of workers and remains efficient as long as the size of W is not too large. Third, the algorithm takes only one parameter λ that takes reasonably large values depending on the application scenario. However, tuning this parameter can improve the performance of the algorithm.

6.5 Experimental Evaluation

A set of experiments were performed on a real-world dataset to evaluate the performance of the SpatialUCB algorithm. Following is a set of non-contextual algorithms, each from one category of the heuristics discussed in Section 6.2, that were evaluated to serve as the baseline during the experiments:

- ϵ -Greedy: This algorithm is based on a semi-uniform heuristic that chooses assignments randomly with probability ϵ and with probability $1 - \epsilon$ the Hungarian algorithm is used to choose the assignment set. The parameter ϵ controls the rate of exploration. High values of ϵ result in high exploration.
- Softmax: This algorithm, based on a probability matching heuristic, estimates reliability scores according to the probability that it is proportional to the observed reliability of workers and uses the Hungarian algorithm to choose the assignment set [40]. The τ parameter moderates the degree to which the workers with the highest expectations are chosen for tasks. Higher values of τ mean more exploration of workers.
- UCB2: This algorithm estimates reliability scores according to a parameterized version of the upper confidence bound and uses the Hungarian algorithm to choose the assignment set [64]. The parameter λ controls the intensity of the confidence bound.

6.5.1 Evaluation Methodology

Similar to the evaluation methodologies in previous chapters, an agent-based simulation is used to compare and contrast the algorithms described in this

Table 6.1: Overview of dataset used for data-driven initialization

	Gowalla Dataset
Task locations, l_i	30,367 spots
Workers, W	19,183 users
Worker locations, $l_j(r)$	357,753 check-ins
Type 1 tasks	2,767 highlights
Type 2 tasks	357,753 check-ins

chapter. The agents in the simulation were instantiated, following a data-driven initialization approach, with a real-world dataset.

Datasets

The real-world dataset is based on data collected from a location-based social network: Gowalla [13, 37]. This dataset contains data about people who have voluntarily reported their visits to various locations. The data was collected for locations in the New York city during October 2011. A *spot* is a geo-referenced location in New York city. A *highlight* represents a particular tag associated with a spot by a user. A *check-in* represents the visitor relationship between a user and a spot. The dataset contains 19,183 users, 30,367 spots, 2,767 highlights, and 357,753 check-ins. The relationship of this dataset with the experimental setup is summarized in Table 6.1.

A worker agent is initialized according to the check-ins and highlights of a user, where the home location of the worker is set to the last check-in or highlight of the corresponding user in the dataset. Figure 6.3a shows the distribution of the number of check-ins by each user on a logarithmic scale. The distribution shows the Zipf distribution for the number of unique check-ins by a user; the majority of users have very low activity. This behavior is commonly observed in various

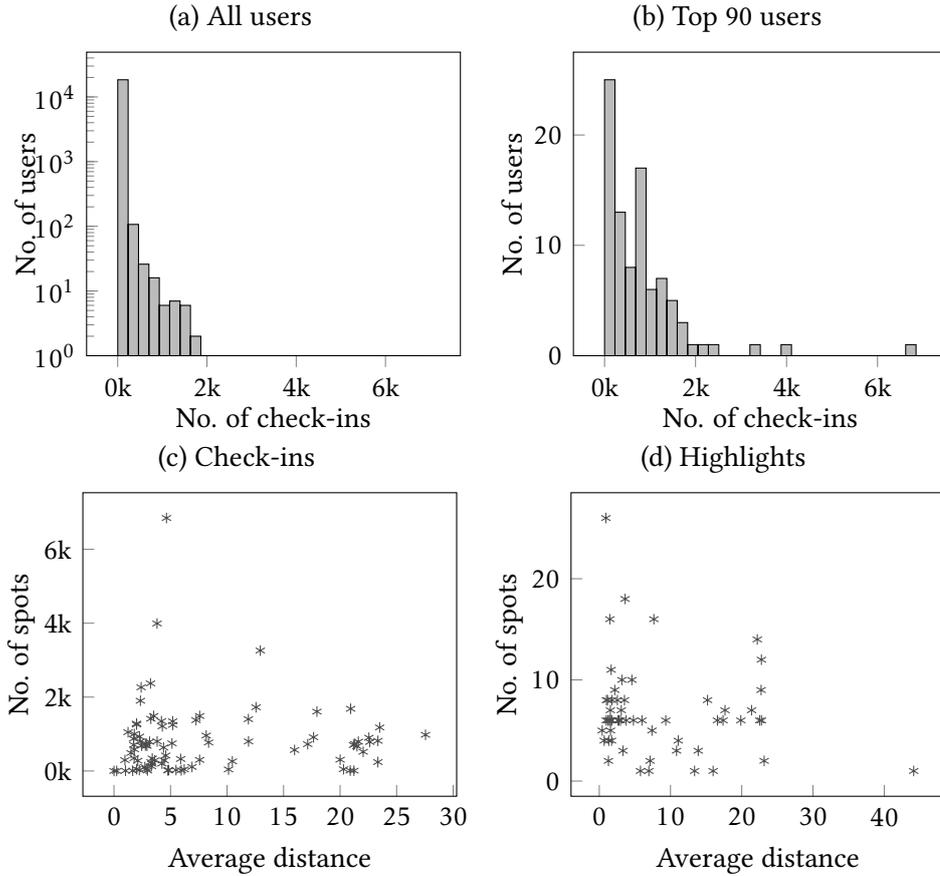


Figure 6.3: Distributions of users against check-ins in the Gowalla dataset and comparison of distance with check-ins and highlights.

physical and social phenomena. Low-activity users, from the long tail distribution, were excluded from the simulated pool of workers. This was achieved by selecting 90 top-ranked users based on their check-ins and highlights. The resulting dataset had 90 users with relatively high levels of activity and mobility. The distribution of check-ins, for selected users, is shown in Figure 6.3b. This group of users forms the pool of workers in the experimental evaluations.

Figure 6.3c shows the number of check-ins against the average distance from user's current spot. Clearly, the check-in behavior varies across users. Some users have higher number of check-ins within 5-10 kilometers, while other users have

visited spots as far as 25 kilometers away. Conversely, there are users who visit a very small number of spots irrespective of the distance. The average distance shows a negative correlation with the number of check-ins. Overall, this behavior is representative of worker dynamics in spatial crowdsourcing; more tasks tend to be completed in the near vicinity of workers [13]. The behavior for highlights is similar for all users, as shown in Figure 6.3d. Again, the majority of users are clustered around the bottom left corner to indicate that the majority of highlights are in the near vicinity of workers.

A requester was simulated by randomly sampling locations from all the spots in the dataset. For the sake of simplicity, each round consisted of exactly one task and the expiry time for each task was set to one round. During the simulation, a highlight, and a check-in are considered two different types of tasks. The contextual vector $Z_{i,j}$ for each assignment consisted of two variables i.e. distance and task type. The type variables is represented as a binary indicator variable, 1 for check-in and 0 for highlight. The distance feature is based on the Euclidean distance between coordinates of the task and the worker. The outcomes of assignments were simulated according to the corresponding check-ins and highlights for workers[37].

Evaluation Metrics

Two metrics were used for the evaluation of assignment algorithms as described below:

- *Task completion rate* (TCR) is the percentage of tasks completed with high quality after the end of all rounds. From a requester’s perspective, the task completion is the primary success criteria.

Table 6.2: Experiment settings used for experimental evaluation of SpatialUCB algorithm.

Agent	Parameter	Description	Values
Worker	$ W $	Number of workers	90
Requester	ξ	Duration of tasks in terms of rounds	1
	$ T $	Number of tasks	5000, 30367
Platform	R	Number of rounds	5000, 30367
	ϵ	Exploration parameter of semi-uniform learning	[0.1, 0.9]
	τ	Temperature parameter of Softmax heuristic	[0.1, 0.9]
	λ	Balance parameters for interval estimation heuristic	[0.1, 0.9]

- *Average travel costs (ATC)* is the average of the distances to be traveled by workers assigned to tasks. Only the distances for the completed tasks were considered for the purpose of reporting.

Experiment Settings

Each experiment was performed by changing a single parameter value while keeping others fixed. Table 6.2 shows the default values for experiment parameters. All of the experiments were run on an Intel Core i7-4600 CPU @2.90 GHz with 16 GB RAM. The algorithms were implemented using the SciPy¹, NumPy², and Pandas³ open source libraries in Python. The Jonker and Volgenant variant of the Hungarian algorithm, as implemented using Pymatgen⁴ library [188], was used for implementing the local optimization strategy. All reported metrics are based

¹<http://www.scipy.org/>

²<http://www.numpy.org/>

³<http://pandas.pydata.org/>

⁴<http://pymatgen.org/>

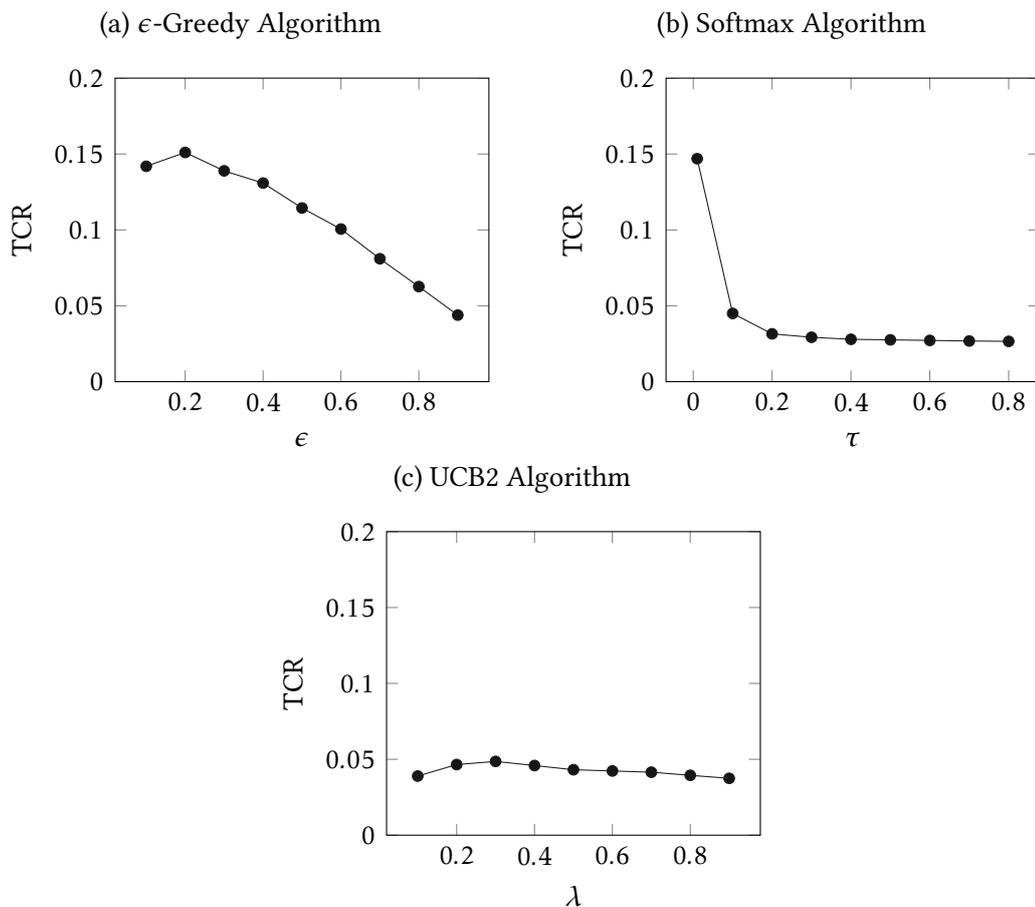


Figure 6.4: Results of context-free algorithms where workers are simulated as Binomial stochastic processes.

on the average of 10 runs for same settings in an experiment.

6.5.2 Experiments without Spatial Context

The first experiment compares the performance of the context-free algorithms. For this purpose, the assignment success behavior of workers was simulated as the Binomial process with worker reliability ρ_j . The worker reliability ρ_j was initialized according to the of percentage check-ins of the top 90 workers in the dataset. Each algorithm was run for 5000 rounds.

Figure 6.4 shows the comparison of algorithm parameter values against the task

Table 6.3: Comparison of algorithms for 5000 tuning tasks.

Assignment Algorithm	TCR	ATC
<i>Context-free algorithms</i>		
Softmax($\tau = 0.01$)	0.09204	18.52864
Softmax($\tau = 0.1$)	0.020880	15.950382
ϵ -Greedy($\epsilon = 0.2$)	0.06326	17.058638
ϵ -Greedy($\epsilon = 0.3$)	0.06232	18.604301
<i>Contextual algorithms</i>		
SpatialUCB($\lambda = 0.2$)	0.1465	16.28432
SpatialUCB($\lambda = 0.5$)	0.1465	15.85458
SpatialUCB($\lambda = 1.0$)	0.09814	13.858756
SpatialUCB($\lambda = 2.0$)	0.06608	14.271388

completion rate. The results indicate that the ϵ -Greedy and Softmax algorithms achieve the highest task completion rate. The ϵ -Greedy algorithm achieves the highest task completion rate for small values of ϵ . Similarly, the highest task completion rate for Softmax is achieved at $\tau = 0.01$. These results suggest that high exploration has negative effects on the success rate. The better performing algorithms were quickly able to identify the small set of workers with higher reliability. By comparison, the optimistic exploration strategy of the UCB2 algorithm resulted in the unnecessary selection of workers with low-reliability. In general, task completion rate is low for all algorithms. This is primarily due to the small initialization values of worker reliability ρ_j . The context-free algorithms were limited to ϵ -Greedy and Softmax, for the rest of the experiments, respectively with parameter values of $\epsilon \in \{0.2, 0.3\}$ and $\tau \in \{0.01, 0.1\}$.

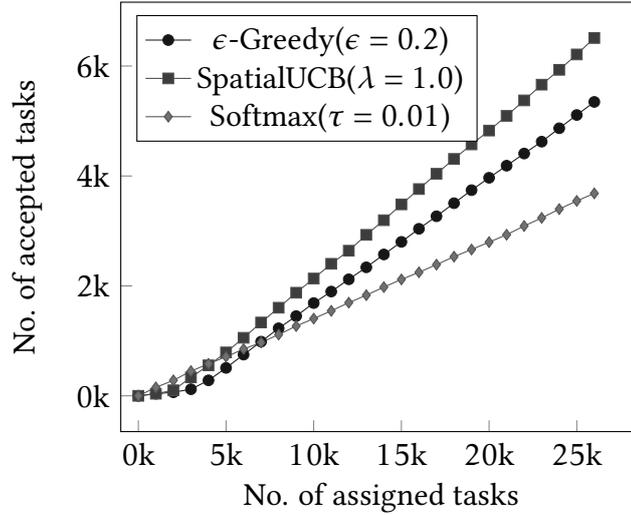


Figure 6.5: Comparison of algorithms on 26,662 test tasks.

6.5.3 Experiments with Spatial Context

The second set of experiments evaluated the performance of assignment algorithms based on contextual variables. As a first step, the parameter values of algorithms were fine-tuned on a random set of 5000 tasks from the dataset. During this step, the SpatialUCB algorithm was compared with the algorithms selected in the previous experiment. The parameter value of SpatialUCB was set according to the commonly found values in the literature i.e. $\lambda \in \{0.2, 0.5, 1, 2\}$. Table 6.3 lists the resulting task completion rate on the tuning tasks. The results indicate that the SpatialUCB performs better, both in terms of the task completion rate and the average travel costs, as compared to the other algorithms. This highlights the fact that the SpatialUCB algorithm was effective in learning the linear relationship between contextual variables and the task acceptance behavior of a worker.

As a second step, the performance of the tuned algorithms was evaluated on the rest of the 26,662 tasks in the dataset. Figure 6.5 shows the performance of ϵ -Greedy, Softmax, and SpatialUCB algorithm with tuned parameters. We ran

each algorithm 10 times on and report the number of completed tasks after each round of assignments. The results show that the number of completed tasks increases linearly with the number of assigned tasks. The tuned SpatialUCB algorithm performs consistently better than other algorithms. The difference in the performance is not in multiple orders of magnitude.

6.6 Discussion

The primary goal of this chapter was to present contextual learning for adaptive task assignment. This goal was achieved through the description of the SpatialUCB algorithm. A key strength of the proposed approach is that it effectively leverages the contextual information about tasks and worker for the estimation of worker reliabilities. Due to this, the performance of SpatialUCB improves significantly against context-free algorithms. As shown in the results, the relative performance of contextual algorithm improves over time. A similar approach can be applied to the other forms of crowdsourcing, including but not limited to expertise-based crowdsourcing.

The reliance on context information can also be considered a limitation. Worker-specific data collection can easily become a privacy bottleneck. In terms of the evaluation, this paper considers a linear regression model for the relationship between assignment outcome and contextual variables. Consideration for other regression models, such as Logistic and Probit, is an area for further research. Another limitation is the fixed cardinality constraints of one worker per task. This may result in a low task completion rate when the worker population is reluctant in general. Instead of a fixed cardinality constraint, redundancy for assignments can be introduced to improve the task completion rate.

6.7 Chapter Summary

This chapter introduced the maximum reliability assignment with context problem in spatial crowdsourcing. The problem formalizes three requirements of adaptive task assignment: explore-exploit trade-off, spatial context, and contextual learning. A set of baseline context-free algorithms is described that address the explore-exploit trade-off. This chapter proposes the SpatialUCB algorithm. The proposed algorithm employs an interval estimation heuristic to address the explore-exploit trade-off. The heuristic uses individualized linear models to estimate worker reliabilities. The SpatialUCB algorithm also addresses the spatial context and contextual learning requirements. The SpatialUCB algorithm is evaluated against a set of context-free algorithms on a real-world dataset. The results suggest that the SpatialUCB achieves higher task completion rate. The results validate the use of contextual learning in adaptive task assignment.

Chapter 7

Adaptive Semi-Assignment with Location Diversity

The instances of the adaptive assignment problem in previous chapters considered the formulations where assignments were constrained with one worker per task. Recent studies have established that tasks in densely populated areas are more likely to be performed as opposed to tasks in sparsely populated areas [31]. Due to this, the coverage of tasks spread over a large area becomes a difficult problem. This problem is exasperated by the fact that the socio-economic status of a location also affects the worker reliability to perform a task [24]. One solution is to design incentive mechanisms that reward workers for performing tasks depending on the location [195]. However, such a solution is not applicable in the case of volunteered crowdsourcing.

Given that the assignment success is probabilistic in reality, it is better to assign multiple workers to a task. This requires the design of assignment algorithms that

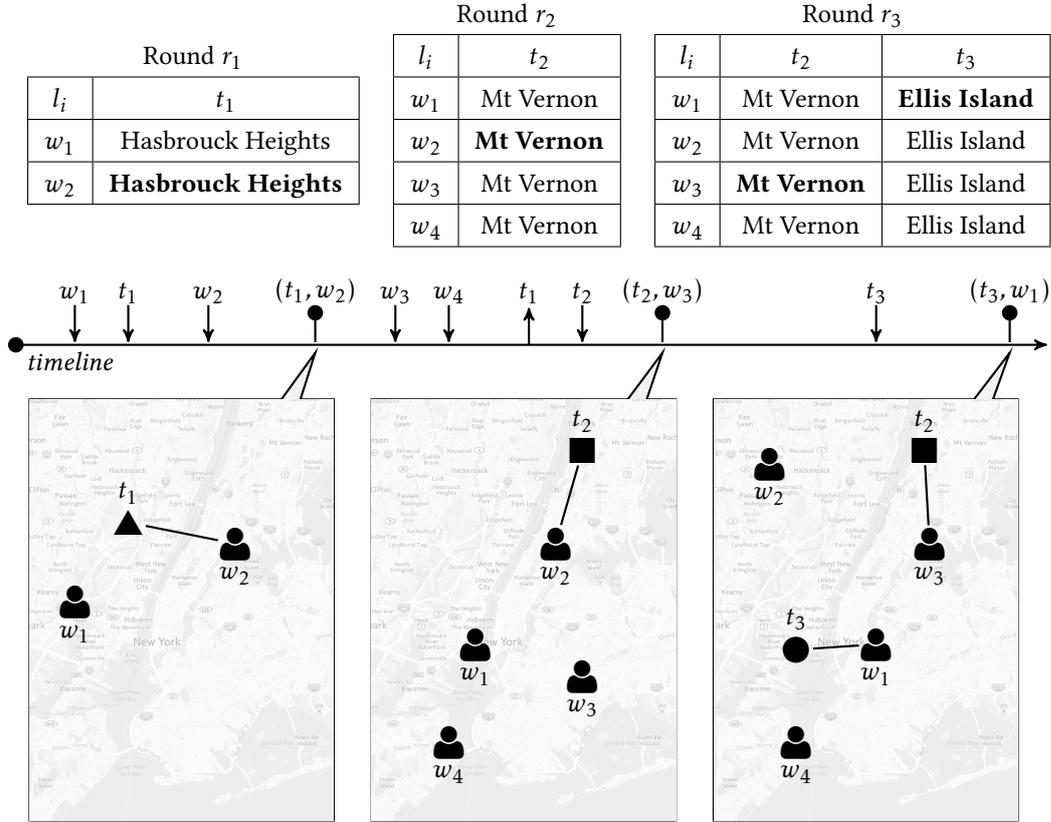


Figure 7.1: Three rounds of adaptive task assignment with task location.

choose a set of workers such that some of them might perform the task [13, 32]. Hence, it is essential to determine the set of workers to be assigned to a task based on the location characteristics and worker reliabilities. Figure 7.1 illustrates this scenario in further detail. During each round, the assignment algorithm has access to the location tasks which can be used to determine the number of workers to be assigned to a task and the estimated worker reliability can be used to determine the set of workers to be assigned to the task.

In Section 7.1, the *maximum reliability semi-assignment* (MRSA) problem is introduced that is a generalization of the *maximum reliability assignment* (MRA) problem. The MRA focuses on the *selection issue* in adaptive task assignment, where an algorithm is primarily concerned with choosing optimal assignments

over multiple rounds while learning from the outcomes of decisions. The MRSA problem also considers the *quantity issue* in adaptive task assignment, where an adaptive algorithm also determines the number of workers to be assigned to a task depending on spatial heuristics. The MRSA formalizes the explore-exploit trade-off, spatial context, and multi-criteria optimization requirements of adaptive task assignment in spatial crowdsourcing.

Generally, existing research addresses the quantity issue by considering fixed constraints on the number of workers per task [13, 25]. Section 7.2 summarizes the baseline approaches of addressing the maximum reliability semi-assignment problem. Section 7.3 proposes a spatial heuristic to determine the number of workers to be assigned to a task. The heuristic is based on the location diversity of a task. Section 7.4 describes the proposed assignment algorithm that uses the location diversity heuristic for the quantity issue and a probability matching heuristic for the selection issue. Section 7.5 presents an experimental evaluation of the proposed algorithms against baseline algorithms using an agent-based simulation methodology and real-world datasets. Section 7.6 discusses the implication of the proposed algorithm and Section 7.7 concludes the chapter with key findings.

7.1 Problem Description

Chapter 5 introduced the *maximum reliability assignment with context* (MRA-C) problem with unknown worker reliabilities and contextual variables. However, the assignments had cardinality constrained of one worker per task. The success probability of an assignment is assumed to be a function of the contextual variables

$\mathbf{Z}_{i,j}$, worker specific coefficients \mathbf{U}_j , and random factors $e_{i,j}$.

$$p_{i,j} \sim f(\mathbf{Z}_{i,j}, \mathbf{U}_j, e_{i,j})$$

In general, the function f is not known and the cardinality constraint results in a low task completion rate when the worker population is generally reluctant. This chapter relaxes the cardinality constraint and allows multi-worker assignments per task. The overall goal is to design an algorithm that aims to maximize the reliability $P(r)$ over all rounds (see. Equation 4.1).

Definition 7.1 (Maximum Reliability Semi-Assignment). *The problem of maximum reliability semi-assignment is to determine the number of unique workers k_i to be assigned to a task and select the assignment set such that the total reliability is maximized over all rounds i.e. $\max \prod_{r=1}^R P(r)$.*

A global solution to the MRSA problem is not also feasible unless the assignment process is clairvoyant. If the worker reliabilities $p_{i,j}$ are estimated at the start of each round, then a local optimization strategy can be employed. However, even the local version of the MRSA problem is NP-Hard by reduction to the number partitioning problem [38]. An alternative formulation of the local MRSA problem assumes that the number of workers for each task is known, which

is represented by the following integer program:

$$\begin{aligned}
 & \max \prod_{t_i \in T_r} \prod_{w_j \in W_r} p_{i,j} \cdot x_{i,j} & 7.1 \\
 & \text{s.t.} \quad \sum_{w_j \in W_r} x_{i,j} = k_i \quad \forall t_i \in T_r \\
 & \quad \quad \sum_{t_i \in T_r} x_{i,j} = 1 \quad \forall w_j \in W_r \\
 & \quad \quad x_{i,j} \in \{0, 1\} \quad \forall t_i \in T_r, w_j \in W_r
 \end{aligned}$$

This chapter considers the scenario when both the k_i and $p_{i,j}$ are unknown; therefore, requiring multi-criteria optimization and online learning with explore-exploit trade-off. The next section discusses some baseline heuristics aimed at addressing these issues.

7.2 Baseline Approaches

To address the quantity issue, a simple heuristic is to choose an empirically tested fixed value for the number of workers per task. On one hand, the assignment algorithm can set $k_i = \gamma m$, where γ is a scaling parameter and $m = |W_r|$ is the number of available workers. In which case, some tasks might get an unnecessarily high number of workers assigned. On the other hand, the algorithm can assign only one worker which might result in a lower completion rate for tasks. Instead, as a baseline the assignment size can be fixed after empirically testing different values such that $k_i = b$. To address the selection issue, a baseline heuristic is to choose the worker nearest to the task. Understandably, such an approach would require the knowledge about the distance between tasks and workers. This

distance-based worker selection approach may be suboptimal since it does not consider the worker reliabilities.

7.3 Location Diversity

This thesis proposes a novel heuristic that aims to optimize the quantity issue using the spatial characteristics of task location. Given that the location of a task is l_i and the maximum allowance of workers for the task is b , the goal of such a heuristic is to define a function $g(l_i) \rightarrow k_i$ such that $k_i \in [1, b]$. As discussed earlier, the location of a task is one of the major factors that determine the reliability of workers to perform a task. One heuristic that can be exploited for this purpose is the distribution of workers visiting a location, also known as location diversity [196]. A task is more likely to be completed if it is located in a frequently visited area, whereas the task in a sparsely visited area might require more workers.

The notion of *location entropy* is used to represent the location diversity [13, 196]. A location has high-entropy when many workers visit that location with equal proportions; conversely, a location has low-entropy if a limited number of workers visit the location. The underlying intuition is that a location with high-entropy will require less workers to ensure coverage. Let $\Omega(l)$ be the number of visits to a location l and $\Omega(w_j, l)$ be the number of visits made by a worker w_j to the location l . The probability that a random visit to location l is also a visit by worker w_j is defined as

$$Pr(w_j, l) = \frac{\Omega(w_j, l)}{\Omega(l)} \tag{7.2}$$

which is the fraction of visits to l made by worker w_j . The *location entropy* of

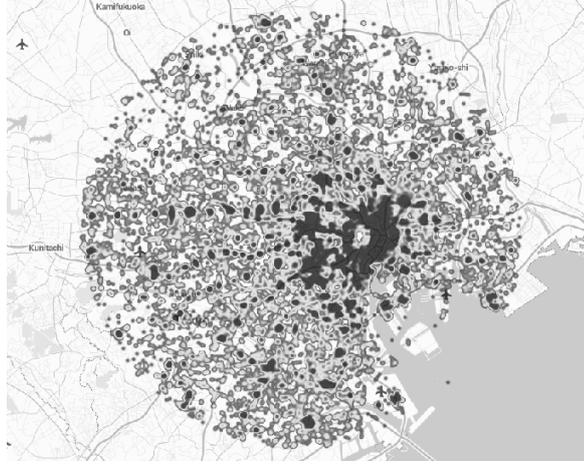


Figure 7.2: The heat map of Tokyo dataset based on location diversity of locations.

a location is formally defined as follows

$$H(l) = \begin{cases} -\sum_{w_j \in W_l} Pr(w_j, l) \cdot \log Pr(w_j, l) & \text{if } \Omega(l) \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad 7.3$$

where W_l is the set of workers who have visited the location l . The function to determine the size of assignment of a location is defined as follows

$$g(b, l) = \left\lceil b \cdot \left(1 - \frac{H(l)}{\max_{l \in L} H(l)} \right) \right\rceil \quad 7.4$$

Henceforth, the quantity issue is addressed by using $k_i = g(b, l_i)$ for task t_i . This heuristic allows a better trade-off between the number of workers per task and the expected completion of a task. The location diversity can be quantified with the help of data from *mobile phone networks*, location-based social networks, or similar geo-location services. Section 7.5 discusses the datasets, from a location-based social network, that are used to approximate location diversity during experimental evaluation. Figure 7.2 shows an example of the distribution of

location diversity based on data from Tokyo dataset.

7.4 Estimating Reliabilities

Given the value for k_i for task t_i , the selection issue is addressed by estimating the reliability of workers and solving Equation 7.1 using the k_i . This requires an online learning heuristic to address the explore-exploit trade-off. Figure 7.3 illustrates the workflow for the combinatorial bandits approach with dynamic assignment size and highlights the improvements compared to the combinatorial bandits approach discussed in Chapters 4 and 5. Previous chapters have established that the upper confidence bound heuristic serves well for this purpose, with or without the contextual variables. This chapter introduces another heuristic that has been shown to perform well both empirically and theoretically [197].

7.4.1 Thompson Sampling

Thompson sampling is a Bayesian heuristic for decision making that chooses actions such that the expected reward is maximized with respect to a randomly drawn belief [198]. In this approach, two counters are maintained for each worker: θ_j is the number of tasks assigned and ϑ_j is the number of tasks successfully completed. The selection of an assignment is primarily governed by a Bayesian approximation approach for the expected reliability. The expectation of the reliability of an assignment of a task t_i to a worker w_j is approximated with the $\hat{p}_{i,j}$ variable. In each round, the $\hat{p}_{i,j}$ is sampled from Beta distribution with parameters that are specific to worker w_j . Based on two prior parameter values α_0 and β_0 , the shape of Beta distribution is adjusted using the known successes ϑ_j and failures $(\theta_j - \vartheta_j)$ of assignments. After observing the outcome variables $y_{i,j}$, the counters

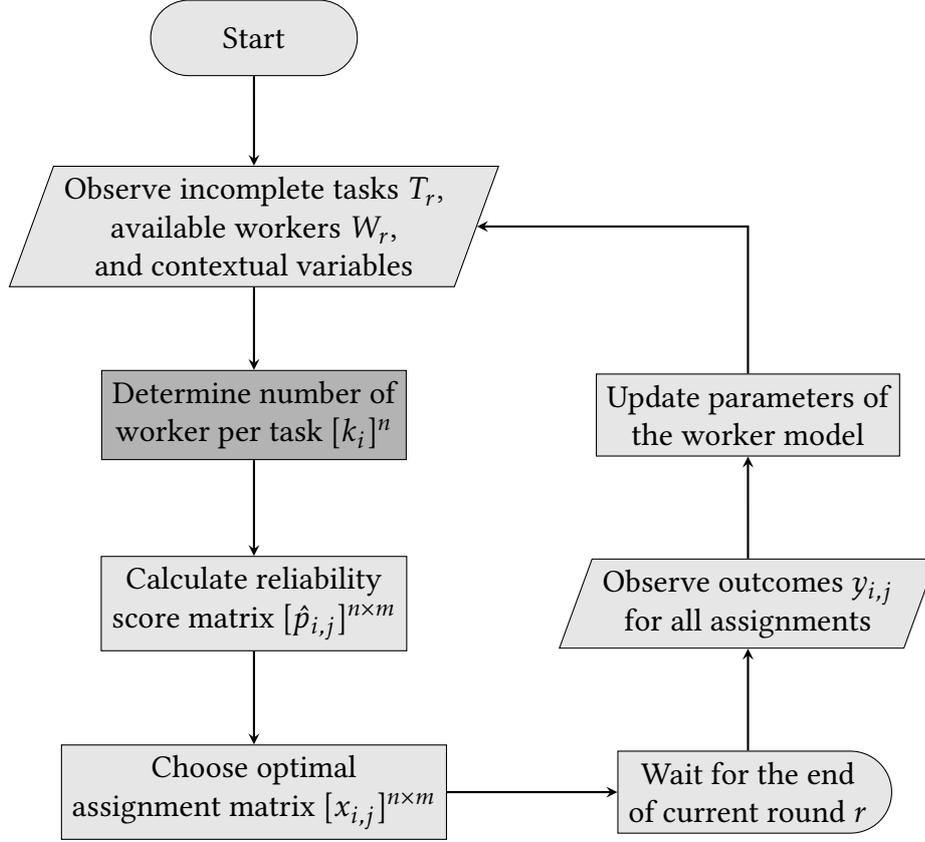


Figure 7.3: Workflow of the adaptive assignment problem with the dynamic number of workers per task.

are incrementing appropriately for selected workers .

7.4.2 DynTS Algorithm

Algorithm 5 provides the primary steps of the algorithm. The algorithm requires three parameters: b the maximum limit of assignments per task and the parameters for the prior Beta distribution α_0, β_0 . The algorithm starts by initializing the success and task counters for each new worker (Lines 4-8). Next, the algorithm determines the number of workers to be assigned k_i to each task using Equation 7.4 (Line 12). The algorithm estimates the reliability score for each assignment by sampling from the adjusted Beta distributions for all workers i.e. $\hat{p}_{i,j} \sim \text{Beta}(\beta_0 +$

$\vartheta_j, \alpha_0 + \theta_j - \vartheta_j$) (Line 13). The set of workers to be assigned to a task is dependent on the reliability scores $\hat{p}_{i,j}$ and the number of workers k_i . The algorithm chooses the assignment matrix by solving the following integer program.

$$\begin{aligned}
& \max \sum_{t_i \in T_r} \sum_{w_j \in W_r} p_{i,j} \cdot x_{i,j} && 7.5 \\
& \text{s.t.} \sum_{w_j \in W_r} x_{i,j} = k_i && \forall t_i \in T_r \\
& \sum_{t_i \in T_r} x_{i,j} = 1 && \forall w_j \in W_r \\
& x_{i,j} \in \{0, 1\} && \forall t_i \in T_r, w_j \in W_r
\end{aligned}$$

The structure of the above integer program allows polynomial time solutions using shortest path approaches [199] (Line 14). The algorithm then assigns tasks to workers and waits until the end of the current round (Lines 15-17). At the end, the algorithm observes the outcomes of assignments and updates the covariance matrix and the response vector (Lines 18-22).

The computational complexity of the DynTS algorithm is dominated by the complexity of the sub-routine used to solve Equation 7.5, which can be reduced to the *semi-assignment problem*. Volgenant noted that by taking advantage of identical rows in the assignment matrix it is possible to find solutions quickly even for large dimensions [200]. The algorithm also accommodates a dynamic pool of workers and remains efficient as long as the size of the total set of workers W is not too large.

Algorithm 5 The DynTS algorithm

Require: $\alpha_0, \beta_0, b, R, T, W$

```

1: for  $r \leftarrow 1$  to  $R$  do
2:    $T_r \leftarrow \text{Active}(T)$                                 {Set of incomplete tasks}
3:    $W_r \leftarrow \text{Available}(W)$                         {Set of available wokrers}
4:   for all  $w_j \in W$  do
5:     if  $w_j$  is new then
6:        $\theta_j \leftarrow 0$                                 {Initialize assignment counter}
7:        $\vartheta_j \leftarrow 0$                             {Initialize success counter}
8:     end if
9:   end for
10:   $n \leftarrow |T_r|$ 
11:   $m \leftarrow |W_r|$ 
12:   $[k_i]^n \leftarrow [g(b, l_i)]^n$                         {Workers per task}
13:   $[\hat{p}_{i,j}]^{n \times m} \leftarrow \text{TS}(\alpha_0, \beta_0, [\theta_j]^m, [\vartheta_j]^m)$     {Estimated reliability scores}
14:   $[x_{i,j}]^{n \times m} \leftarrow \text{SemiAssignment}([s_{i,j}]^{n \times m})$     {Assignment matrix}
15:   $\text{Assign}(t_i, w_j) \quad \forall x_{i,j} > 0$                 {Assign tasks}
16:   $\text{Wait}(\tau)$                                         {Wait for the end of round}
17:  for all  $(x_{i,j} > 0)$  do
18:     $y_{i,j} \leftarrow \text{Complete}(< t_i, w_j >)$         {Completion indicator}
19:     $\theta_j \leftarrow \theta_j + 1$                         {Update assignments counter}
20:     $\vartheta_j \leftarrow \vartheta_j + y_{i,j}$                     {Update success counter}
21:  end for
22: end for
    
```

Table 7.1: Overview of baseline and proposed algorithms

Workers per Task	Assignment Strategy		
	<i>Random</i>	<i>Distance</i>	<i>Thompson sampling</i>
<i>Fixed, $k_i = b$</i>	FixRAN	FixDIST	FixTS
<i>Dynamic, $k_i = g(b, l_i)$</i>	DynRAN	DynDIST	DynTS

7.5 Experimental Evaluation

This section presents an empirical evaluation of the proposed DynTS algorithm against baseline algorithms. The purpose of evaluation is to establish that the algorithm addresses both the quantity issue and the selection issue in the MRSA problem. Table 7.1 lists the algorithms compared during the experiments, each of which is summarized below:

- The FixRAN algorithm assigns a fixed number of workers per task and selects the assignment set randomly. It serves as a worst-case baseline algorithm.
- The FixDIST algorithm assigns a fixed number of workers per task. It chooses the assignment set according to the closeness of workers to task locations.
- The FixTS algorithm assigns a fixed number of workers per task. It selects the assignment set using Thompson sampling; it should highlight the relative benefit of Thompson sampling.
- The DynRAN algorithm dynamically decides the number of workers for a task using location diversity. It selects the assignment set randomly. This algorithm should highlight the relative benefit of using location diversity.
- The DynDIST algorithm employs location diversity and chooses the assignment set according to the closeness of workers to task locations.
- The DynTS algorithm employs location diversity and selects the assignment set using Thompson sampling.

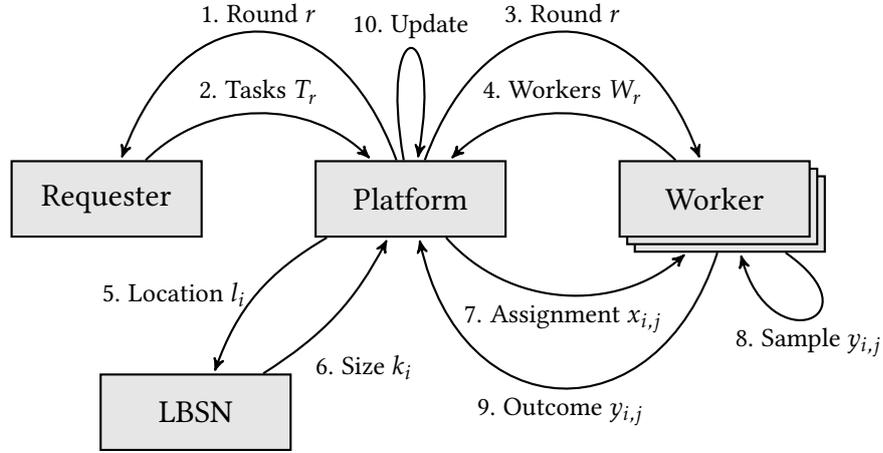


Figure 7.4: Overview of the interaction between simulated agents

The evaluation methodology is detailed next to clarify the specifics of the experiments, which is followed by the results of experiments.

7.5.1 Evaluation Methodology

An agent-based simulation methodology was followed for evaluating the performance of algorithms. Figure 7.4 provides an overview of the simulated agents and their interactions, based on the extension of the simulation methodology discussed in Chapter 4. The simulation workflow is implemented in a platform agent that also implements a periodic-assignment protocol and all of the compared assignment algorithms. The requester agent implemented a queue of spatial tasks with their associated locations and the platform dequeued tasks by calling the requester agent for each iteration. An LBSN agent was created to represent a geo-location service which was queried by the platform agent to determine the entropy of locations. Many worker agents were created along with their specific parameters and mobility patterns. The mobility pattern of each worker was defined according to a set of locations and the time of visit at that location. Each

Table 7.2: Overview of dataset used for data-driven initialization

	New York Dataset	Tokyo Dataset
Workers, W	1,083 Users	2,293 Users
Worker locations, $l_j(r)$	227,428 check-ins	573,703 check-ins
Task locations, l_i	38,333	61,858

worker agent was initialized with coefficients for a set of contextual variables which defined the reliability of tasks assigned to the worker agent. The contextual vector $\mathbf{Z}_{i,j}$ consisted of variables: the distance between task and worker $d_{i,j}$ and the socio-economic status of the task location $SES(l_i)$. The distance variable was calculated using the last location of the worker at the time when the platform agent assigned a task to the worker.

Based on the observation that highly visited locations are generally located in high economic activity areas [24, 31], the socio-economic status variable of task location was approximated using the popularity of a location i.e. number of unique people who visited the location. The reliability of assignment was simulated as follows:

$$p_{i,j} \sim \frac{e^{\mathbf{Z}_{i,j}\mathbf{U}_j}}{1 + e^{\mathbf{Z}_{i,j}\mathbf{U}_j}}$$

During simulation, the platform agent queried appropriate worker agents for assignment outcomes. The worker agent sampled the assignment outcome variables $y_{i,j}$ by a Bernoulli trial with parameter $p_{i,j}$. The worker specific coefficients for distance and socio-economic variables were sampled from Standard Normal distributions with means u_{dist} and u_{ses} , respectively. At the end of an iteration, the platform agent updated its assignment policy based on the observed outcomes.

Datasets

Real-world data from a location-based social network was used to initialize the requester, LBSN, and worker agents. For this purpose, two datasets extracted from a location-based social network: FourSquare¹ [201]. The datasets contain voluntarily reported visits to various locations by the residents of New York and Tokyo cities from April 2012 to February 2013. Each *spot* is a geographically referenced location and a *check-in* represents the visitor relationship between a user and a spot. Table 7.2 lists the properties of two datasets: the New York Dataset and the Tokyo Dataset. Figure 7.2 shows the location entropy of spots in New York Dataset based on user check-ins. The requester agent was initialized by randomly sampling locations from the dataset, each of which was queued according to the timestamp of associated a check-in. The worker agents were initialized according to the check-ins of individual users in a dataset. The LBDN agent was initialized by calculating the location entropy for all spots based on their associated check-ins in a dataset.

Metrics

The following metrics were used to compare and contrast the performance of algorithms:

- *Task completion rate* (TCR) is the percentage of tasks completed with high quality after the end of all rounds. From a requester's perspective, the task completion is the primary success criteria.
- *Relative reduction in assignments* (RAR) is the percentage reduction in the total number of assignments for a dynamic k_i against a fixed k_i .

¹<http://foursquare.com/>

Table 7.3: Experiment settings used for experimental evaluation

Agent	Parameter	Description	Values
Worker	u_{dist}	Sampling parameter for the co-efficients of distance variable	-3, -2, -1
	u_{ses}	Sampling parameter for the co-efficients of socio-economic variable	0.5, 1, 1.5
Requester	$ W $	Number of workers	50, 100, 500
	ξ	Duration of tasks in terms of rounds	1
	$ T $	Number of tasks	1000, 5000, 50000
Platform	R	Number of rounds	1000, 5000, 50000
	b	Maximum budget of worker to be assigned to a task	1, 3, 5, 7, 9
	α_0	First parameter of Thompson sampling heuristic	0.5, 1, 2
	β_0	Second parameter of Thompson sampling heuristic	0.5, 1, 2

This secondary metric defines the overhead problem due to unnecessary assignments in fixed algorithms.

- *Number of completed tasks* (NCT) is a secondary metric that is used to show the progression of assignment process over time.

Experiment Settings

Each experiment was performed by changing a single parameter value, while keeping others fixed. All of the experiments were run on an Intel Core i7-4600 CPU @2.90 GHz with 16 GB RAM. The algorithms were implemented using the open source libraries in Python. The Jonker and Volgenant variant of the Hungarian

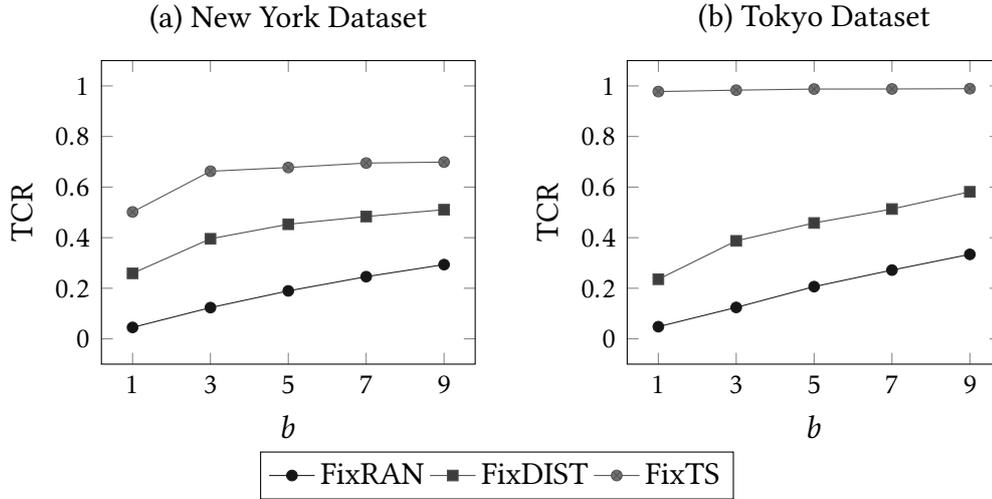


Figure 7.5: Comparison of algorithm performance in terms of task completion rate.

algorithm, as implemented in Pymatgen² library [188], was used for implementing the FixDIST, FixTS, DynDIST, and DynTS algorithms. Table 7.3 lists the range of values for the experimental settings and algorithm parameters, with default values in bold font. All reported metrics are based on the average of 10 runs for the same dataset in an experiment.

7.5.2 Experiments on Task Completion

Figure 7.5 reports the results of the comparison of algorithms based on a fixed number of assignments for each task. Clearly, the FixTS algorithm performs consistently better than both the baseline FixRAN and FixDIST. This highlights that the adaptive approach of learning worker reliability is indeed better than the non-adaptive approach of choosing workers based on their distance from the task location. The distance-based non-adaptive approach does reach more than 50% coverage for more than 5 assignments per task. By comparison, the algorithm based on Thompson sampling achieves more than 95% coverage with

²<http://pymatgen.org/>

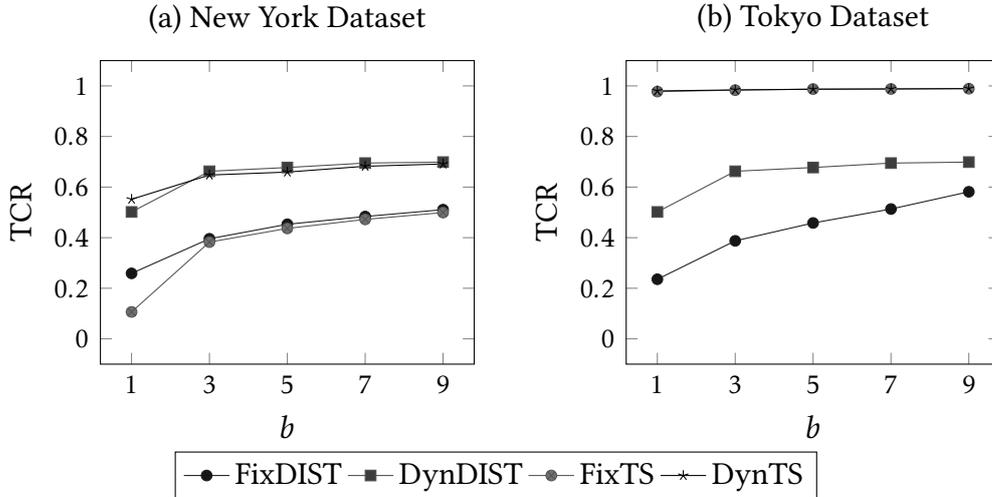


Figure 7.6: Comparison of algorithms between fixed and dynamic assignment size.

only 2 assignments on the Tokyo Dataset. This is achieved without the knowledge of the distance between task and worker locations.

Further analysis revealed that the average number of consents was higher for the FixTS algorithm, while other algorithms remain relatively low for the majority of b . This suggests that the previously observed outcomes are a strong indicator of future reliability. On the other hand, the distance might correlate with reliability but that is dependent on the individual worker's decision model. The results also suggest that reasonably small values of b suffice for achieving more than 95% coverage.

7.5.3 Experiments on Dynamic Assignment Size

Figure 7.6 shows the comparison of algorithms with fixed and dynamic assignment algorithms. A dynamic algorithm tries to reduce the number of assignments by using Equation 7.4. The relative performance of the algorithms does not change significantly between fixed and dynamic assignment sizes. Figure 7.7 shows the relative reduction in the number of assignments for a dynamic policy as compared

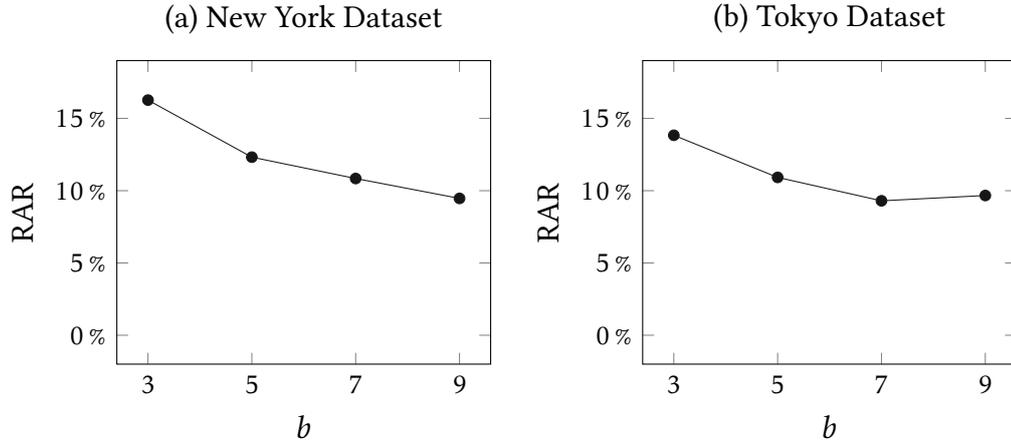


Figure 7.7: Relative reduction in assignments for dynamic size policy against fixed size policy.

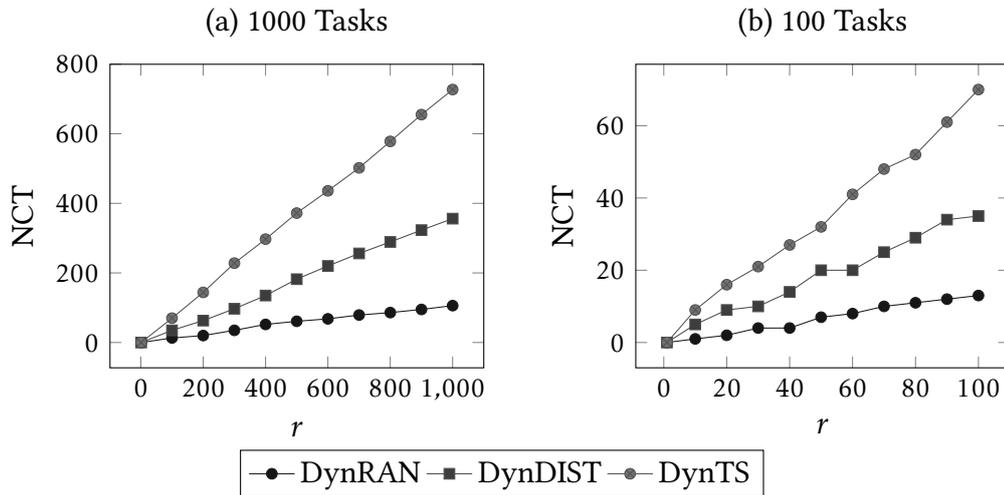


Figure 7.8: Comparison of assignment policies over time for the New York Dataset

to a fixed-size policy. The relative number of workers per task between FixTS and DynTS was compared. The DynTS algorithm required between 10%-15% fewer workers per task while achieving a coverage similar to FixTS. The location diversity for tasks was distributed according to an exponential distribution, which means the majority of locations have a near-zero diversity. This is understandable given that the relative reduction will increase if the diversity of the task location is higher, for instance in dense areas of a city.

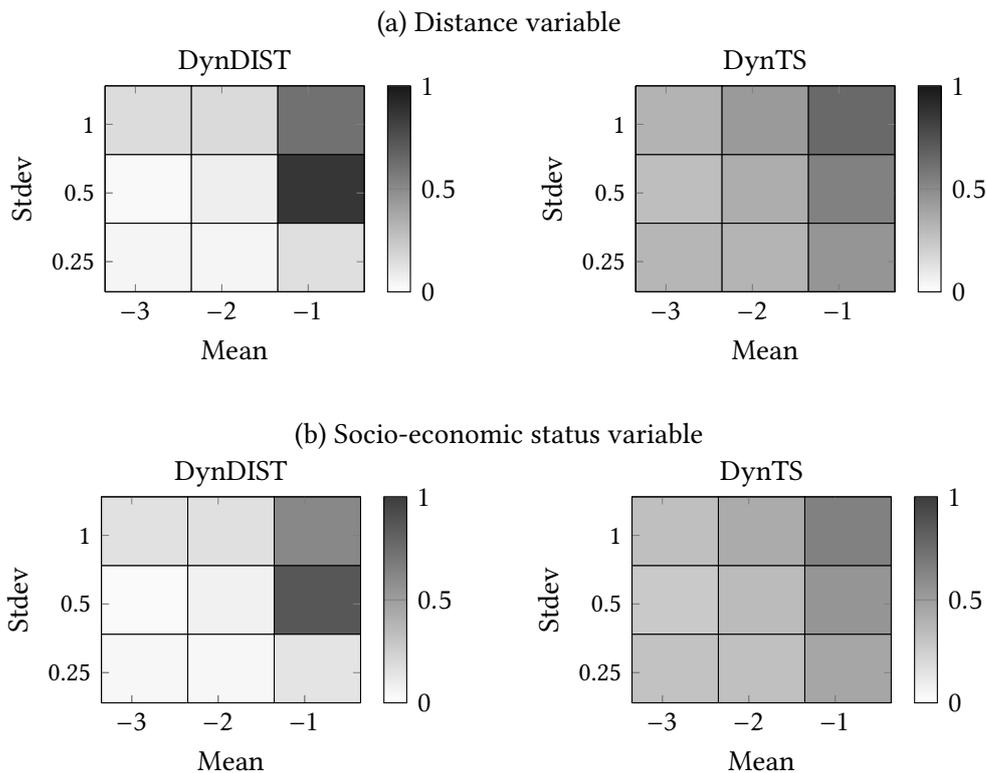


Figure 7.9: Comparison of task completion rate under varying distributions of worker model coefficients for the New York dataset.

7.5.4 Experiments on Task Completion over Time

Figure 7.8 shows the comparison of dynamic assignment algorithms over multiple rounds. All algorithms follow a linear pattern for the number of completed tasks as more rounds are completed. Furthermore, the relative performance of the algorithms is the same over time. Figure 7.8 also shows the comparison on first 100 rounds, which highlights the fact that the DynTS algorithm was quickly able to learn the reliability of workers.

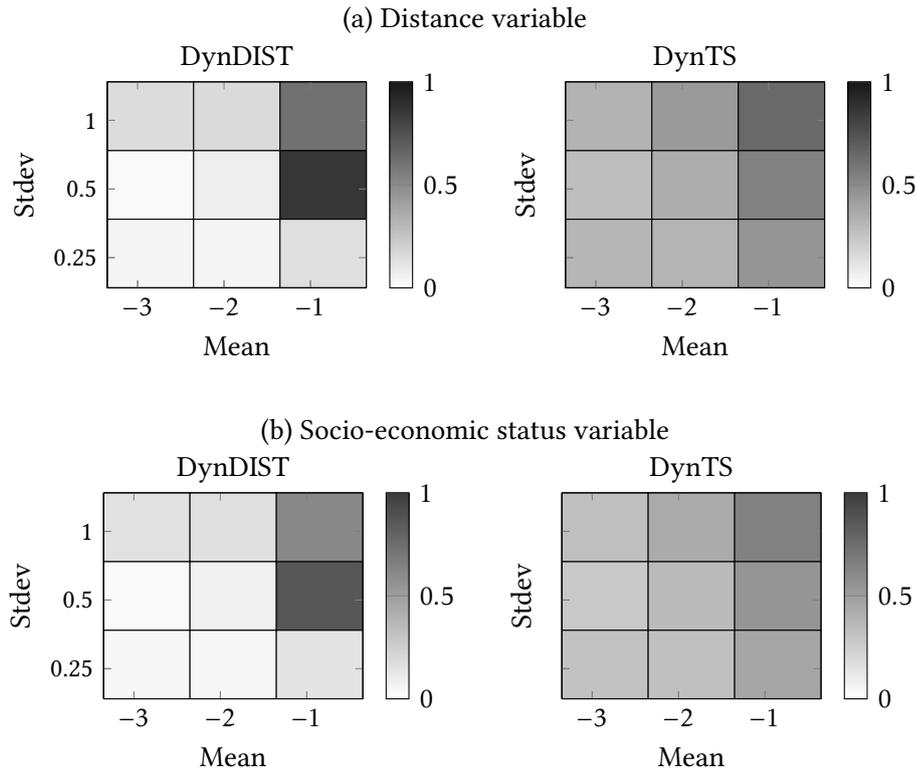


Figure 7.10: Comparison of task completion rate under varying distributions of worker model coefficients for the Tokyo dataset.

7.5.5 Experiments on Worker Distribution

Different distributions of coefficients of workers for the contextual variables were also tested. Figure 7.9 and Figure 7.10 shows the comparison of the DynDIST and DynTS algorithms with varying mean and standard deviation of the coefficients. It is clear that the DynTS algorithm is more sensitive to the distance coefficient. When the coefficients of the distance variable are small and diverse then the DynTS algorithm performs particularly well. This means that the Thompson sampling is effective in adapting the assignment process for diverse and relatively willing workers. By comparison, the non-adaptive DynDIST algorithm is less sensitive to the distribution of distance coefficients. However, the performance does increase with an increase in the number of workers.

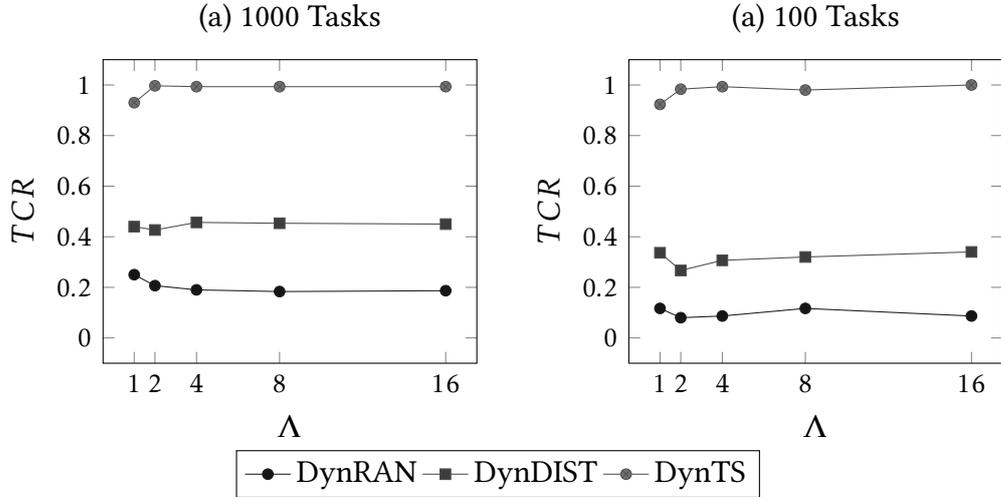


Figure 7.11: Comparison of algorithms in terms of task completion rate against mean inter-arrival time of tasks.

7.5.6 Experiments on Task Arrival Rate

Task arrival times also affect the performance of dynamic algorithms. For this purpose, the requester agent was parameterized with the average inter-arrival rate for tasks. The parameter Λ defines the average interval between two tasks in hours. Figure 7.11 shows the comparison of DynRAN, DynDIST and DynTS algorithms with varying $\Lambda \in \{1, 2, 4, 8, 16\}$. The changes in inter-arrival rate do not have a direct effect on the relative performance of algorithms. This suggests that the distance heuristic and the estimated reliability are not dependent on the frequency of task arrivals.

7.6 Discussion

This section discusses some strengths and limitations of the algorithms and results presented in this chapter. A key strength of the DynTS algorithm is the relaxation of the hard constraint on the number of workers per task. This allows for

adjustment in the assignment strategy to ensure a high task completion rate. Although this chapter proposes a location diversity for assignment size calculation, a similar approach can be applied to expertise-based crowdsourcing. In this case, more workers might be assigned to tasks that have limited expertise in worker population. The DynTS algorithm uses the Thompson sampling heuristic which has two distinct advantages. First, it does not require any specific parameter to control the effectiveness of learning. Second, it is shown to be empirically effective for low time periods [197].

In terms of the evaluation, the worker's decision to complete a task is modeled using the distance between the last check-in of worker and the location of the task. In real-world scenarios the worker might have moved to a new location since the last check-in. One possible option to overcome this limitation is to weight the distance variable according to the recency of the last check-in. Nonetheless, it is expected that the randomness due to the recency will not significantly affect the relative performance of the DynTS algorithm. It is also assumed that the worker reliability is independent of previous assignments. In real-world settings, a sequence of assignments to a worker might be correlated. If the worker is busy with another activity, then it is likely that the reliability of worker for current and future assignments will be lower. This highlights the need to model the worker reliability as a function of time and historical assignments.

The assignment outcome model in this work assumes binary completion of tasks. More specifically, if a worker does not complete a task then it is assumed that the worker is not interested in the task. This assumption can be relaxed by considering probabilistic completion of tasks. For instance, the mobile application on a worker's device can respond with probabilistic estimates of task completion [202]. Furthermore, the online learning process of DynTS can be

adjusted to account for delayed completion of tasks.

The incentives used in spatial crowdsourcing also affect the worker's reliability and participation. Since the goal of experimental evaluation is to establish the utility of location diversity heuristic, the discussion on the incentives is out of the scope. It is reasonable to assume that if higher incentives are offered then workers would be willing to perform more tasks. By comparison, a fair load balancing policy might be required for volunteered crowdsourcing. This is especially true when workers are aware of the workload assigned to each other.

7.7 Chapter Summary

This chapter establishes the utility of heuristics based on the task location for optimizing spatial crowdsourcing. The chapter introduced the maximum reliability semi-assignment problem that generalizes the maximum reliability assignment problem with multiple assigned workers per task. The maximum reliability semi-assignment formalizes the explore-exploit trade-off, spatial context, and multi-criteria optimization requirements of adaptive task assignment in spatial crowdsourcing. A novel adaptive assignment algorithm DynTS is proposed that addresses the maximum reliability semi-assignment problem and its research requirements. The proposed algorithm uses Thompson sampling to estimate the reliability of assignments. It also employs location diversity to optimize the number of assignments per task, while keeping the coverage at the maximum. Empirical evaluation suggests that the proposed algorithm performs better than the baseline non-adaptive and fixed-size algorithms. The DynTS algorithm achieves between 50% to 30% higher task completion rate than the baseline non-adaptive algorithm while requiring 10%-15% fewer assignments.

Chapter 8

Adaptive Assignment using Spatial Expertise

Chapters 4 to 7 focused on the adaptive task assignment in spatial crowdsourcing from an online learning perspective, where assignment decisions are made while learning from the outcomes of previous assignments. Usually, the reliabilities of new workers are initialized with parameterized prior values. This can lead to unnecessary assignments to the unreliable workers as they join the worker pool. Conversely, reliable new workers might not be assigned tasks due to conservative prior values. This *cold-start* problem becomes more evident when worker pools are large and dynamic [20]. The cold-start problem formalizes the explore-exploit trade-off and contextual learning requirements of adaptive task assignment using offline learning. This chapter addresses the cold-start problem by proposing an offline approach for learning worker reliabilities, conveniently referred to as the adaptive assignment with *warm-start*. In this approach, a set of test tasks is given

Table 8.1: Example of vectors for contextual variables and worker expertise in the city maintenance scenario of spatial crowdsourcing.

Topics	Context vectors $\mathbf{Z}_{i,j}$			Worker expertise coefficients \mathbf{U}_j		
	t_1	t_2	t_3	w_1	w_2	w_3
Environment, s_1	0	1	1	0.5	0.2	0.8
Construction, s_2	1	0	0	0.8	0.3	0.5
Photography, s_3	0	0	1	0.2	0.9	0.5

to new workers before they are added to the pool of available workers. Hence, this approach supports adaptive task assignment through an initialization process.

The proposed approach is specifically designed to address the cold-start problem for expertise-based spatial crowdsourcing, where each task requires a specific expertise to be performed correctly. For example, a spatial crowdsourcing task might require a worker to have expertise on “Photography”. The expertise required for performing a task is defined in terms of binary contextual variables, where each variable represents a topic, as shown in Table 8.1. As discussed in Chapter 6, the reliability of workers is represented and estimated using a vector coefficients for the contextual variables, in the case of expertise-based crowdsourcing the vector of coefficients defined expertise level for each topic. This chapter proposes a combined heuristic based self-assessment with task-based assessment for offline learning of worker specific expertise coefficients. The proposed approach is evaluated on data collected by posting tasks on a real-world crowdsourcing platform.

The rest of this chapter is organized as follows. Section 8.1 introduces the cold-start problem for adaptive task assignment in spatial crowdsourcing. Section 8.2 summarize the baseline approaches that address the cold-start problem. Section 8.3 describes the combined assessment heuristic to address the cold-start Section 8.4

presents a greedy algorithm for adaptive task assignment with warm-start. Section 8.5 presents an experimental evaluation of the proposed algorithms based data collected from real-world crowdsourcing platform. Section 8.7 summarizes the chapter with key findings.

8.1 Problem Description

Assume that T is the set of tasks that will dynamically arrive and W be the set of workers that will be joining the worker pool, on the platform during R rounds of spatial crowdsourcing. Let S be the set of domain topics that define the expertise required to perform certain tasks. Each task $t_i \in T$ has an associated set of topics S_i . The context vector $\mathbf{Z}_{i,j}$ defines the expertise requirements of the task using binary variables for each topic. The reliability of a worker w_j in performing the task t_i is defined as:

$$p_{i,j} \sim f(\mathbf{Z}_{i,j}, \mathbf{U}_j, e_{i,j})$$

Where \mathbf{U}_j is the worker specific vector of latent coefficients that defines the expertise; whereas, the $e_{i,j}$ vector represents the unobserved random effects. Table 8.1 further illustrates the vectors for contextual variables and worker expertise with the help of example topics. For instance, the context vector for task t_1 indicates the need for expertise on the topic of “Construction” and the coefficient vector for worker w_1 who has the higher expertise on the same topic. Chapter 6 proposed an online learning approach for adaptive task assignment using contextual variables. In this chapter, an offline learning approach for estimating the worker coefficients \mathbf{U}_j . This approach initializes prior estimates for \mathbf{U}_j using test tasks before a worker becomes available for real tasks.

Let Q be the set of test tasks available for estimating worker coefficients

through offline learning and S_i be the set of topics associated with a task $t_i \in Q$. It is assumed that test tasks cover all of the topics:

$$\left| \bigcup_{t_i \in Q} S_i \right| = |S| = |S|$$

The main goal of the warm-start approach is to efficiently generate the estimated coefficients \hat{U}_j for each new worker joining the platform w_j . This two-phased process of offline learning and online assignment allows adaptive initialization of worker coefficients leading towards better assignment decisions afterward. It is to be noted that the workers are not made aware that the assigned tasks are a test to ensure their normal behavior.

8.2 Baseline Approaches

The majority of existing research on general crowdsourcing uses the test tasks to estimate worker parameters. Generally, a fixed set of text tasks is used for this purpose. However, accurate estimates may require a large number of test tasks when there are several topics that cover actual tasks. Estimating worker parameters with a large number of test tasks is both expensive and time-consuming. It becomes important to minimize the workload of offline learning while maximizing the performance of online optimization.

Liu et al. proposed some simple rules of thumb for deciding the number of test tasks in online crowdsourcing [203]. The proposed rules were limited to simple tasks and do not consider the expertise of workers. Fan et al. proposed an influence based approach for warm-start in non-spatial crowdsourcing [130]. The proposal is based on the similarity between test tasks and actual tasks. Ho et al. also utilized

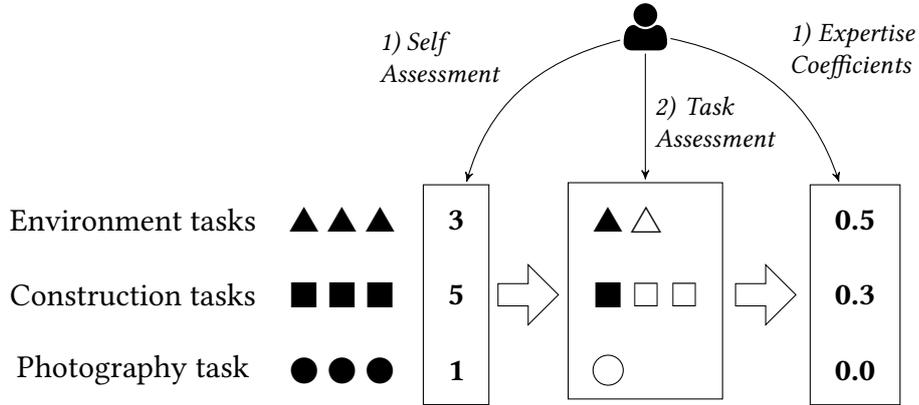


Figure 8.1: Overview of combined assessment approached for estimated worker specific expertise coefficients.

offline learning for a multi-skill scenario in non-spatial crowdsourcing [52]. Their proposal was based on employing a number of test tasks that is in multiples of the number of topics, where the set of tasks belonging to a topic was mutually exclusive from any other topics. This means that no task was associated with multiple topics. In general, these approaches are limited due to the assumption that the pool of actual tasks is fixed and known. By comparison, we consider that scenario where the set of topics is known but there is no knowledge about the actual tasks. Similar to the multi-skill scenario of [52], this chapter considers a fixed set of test tasks as the baseline approach for offline learning. Therefore, the number of test tasks $|Q|$ is a multiple of the number of domain topics $|S|$.

8.3 Estimating Expertise

This chapter proposes a combined assessment approach for estimating worker parameters, as illustrated in Figure 8.1. In the proposed approach, the workers are asked to provide self-assessment of their expertise on each topic before assigning test tasks [204]. An ordered belief scale is used for rating expertise of workers

through self-assessment. A 5-level belief scale is defined for expertise rating of a topic; with ordered levels of *None*, *Poor*, *Fair*, *Good*, and *Excellent*. The chosen expertise level was converted to a normalized value.

The primary purpose of self-assessment is to control the number of test tasks used during offline learning. The combined assessment approach filters the test tasks based on the self-assessed expertise of worker on topics; therefore, guiding the offline learning. The worker is asked to rate their expertise on topics, followed by task assessment on a subset of test tasks (filtered according to expertise level of topics). Since the offline learning is independent of the online assignment algorithm, the next section introduces a purely greedy algorithm for adaptive task assignment.

8.4 Greedy Assignment Algorithm

Algorithm 6 provides the primary steps of the proposed approach. The algorithm requires the number of assignments per task b ; the set of test tasks Q ; and the set of topics S . The algorithm starts by initializing expertise estimates using the warm-start approach discussed earlier and task counters for each new worker (Lines 4-8). The algorithm estimates the reliability score for each assignment by following a greedy approach, i.e. $\hat{p}_{i,j} = \mathbf{Z}_{i,j} \cdot \hat{\mathbf{U}}_j$ (Line 13). The set of workers to be assigned to a task is dependent on the reliability scores $\hat{p}_{i,j}$ and the number of workers k_j . The algorithm determines the assignment matrix by solving Equation 7.5 (Line 13). The algorithm then assigns tasks to workers and waits until the end of the current round (Lines 14-16). At the end, the algorithm observes the outcomes of assignments and updates worker specific expertise coefficients based on the topics associated with the task (Lines 17-20). Similar to the DynTS algorithm,

Algorithm 6 The WS-GRD algorithm**Require:** R, b, T, W, Q, S

```

1: for  $r \leftarrow 1$  to  $R$  do
2:    $T_r \leftarrow \text{Active}(T)$                                 {Set of incomplete tasks}
3:    $W_r \leftarrow \text{Available}(W)$                         {Set of available wokrers}
4:   for all  $w_j \in W$  do
5:     if  $w_j$  is new then
6:        $\hat{U}_j \leftarrow \text{WarmStart}(Q, S)$                 {Initialize parameter estimates}
7:     end if
8:   end for
9:    $n \leftarrow |T_r|$ 
10:   $m \leftarrow |W_r|$ 
11:   $[Z_{i,j}]^{n \times m} \leftarrow \text{Topics}(T_r)$                 {Observe contextual vectors}
12:   $[\hat{p}_{i,j}]^{n \times m} \leftarrow \text{Greedy}([Z_{i,j}]^{n \times m}, [\hat{U}_j]^m)$  {Estimated reliability scores}
13:   $[x_{i,j}]^{n \times m} \leftarrow \text{SemiAssignment}([\hat{p}_{i,j}]^{n \times m}, b)$  {Assignment matrix}
14:   $\text{Assign}(t_i, w_j) \quad \forall x_{i,j} > 0$                     {Assign tasks}
15:   $\text{Wait}(\tau)$                                             {Wait for the end of round}
16:  for all  $(x_{i,j} > 0)$  do
17:     $y_{i,j} \leftarrow \text{Complete}(< t_i, w_j >)$                 {Completion indicator}
18:     $\hat{U}_j \leftarrow \text{UpdateCoefficients}(y_{i,j})$             {Update estimates}
19:  end for
20: end for

```

the computational complexity of the WS-GRD algorithm is dominated by the complexity of the sub-routine used to solve Equation 7.5.

8.5 Experimental Evaluation

This section presents an empirical evaluation of the proposed offline learning heuristics based on combined assessment. The purpose of evaluation is to establish that the worker parameter estimation with combined assessment is effective in

warm-start of adaptive task assignment. Following approaches are compared in terms of warm-start, where each heuristic results in a different estimate for μ_j that is employed in the WS-GRD algorithm.

- RND generates random estimates for μ_j during warm-start.
- SA calculates the estimates for μ_j by using the normalized self-assessment provided by the worker.
- TA estimates the μ_j purely using the test tasks for assessment.
- CA generates the estimated μ_j through pair-wise multiplication of estimates generated by SA and TA.
- CA-P generates the estimated μ_j through pair-wise multiplication of estimates generated by SA and TA with test tasks filtered for topics with “poor” or higher rating.
- CA-F generates the estimated μ_j through pair-wise multiplication of estimates generated by SA and TA with test tasks filtered for topics with “fair” or higher rating.
- CA-G generates the estimated μ_j through pair-wise multiplication of estimates generated by SA and TA with test tasks filtered for topics with “good” or higher rating.
- CA-E generates the estimated μ_j through pair-wise multiplication of estimates generated by SA and TA with test tasks filtered for topics with “excellent” or higher rating.

8.5.1 Evaluation Methodology

A user study was conducted with real workers to evaluate the relative utility of previous discussed offline learning heuristics. The tasks required expertise on specific topics for improving the data quality of a database. The DBpedia¹ project serves as an appropriate use case for this purpose [205]. DBpedia aims at creating a database of facts about real-world entities such as cities, actors, books, games, etc. However, DBpedia suffers from data quality issues such as incorrect values, incorrect mappings, and missing values. Consequently, applications that use DBpedia need to review the data with the help of humans or experts. A set of knowledge intensive tasks were created from entities in DBpedia. Each task simply required human verification of a fact in DBpedia. Tasks were created for two types of entities in DBpedia: Movies and Actors. For example, a task verifies birthplace of an actor by stating the fact “Tom Hanks was born in Concord, California.” and asking for a response from three options “Agree, Don’t Know, Disagree”. Some of the topics associated with this fact that could be used for the definition of expertise include famous actors, Oscar winners, California, etc. Generally speaking, the topics can be any keyword or label that describes the expertise requirements of a task and expertise of a worker. In this context, the task-related topics are based on Movies and Actors classification schemes used within DBpedia.

Datasets

The characteristics of two datasets used in the user study are summarized in Table 8.2. Tasks in each dataset were related to films. Fact verification tasks are based on attribute values of the entities. The topics associated with each task are

¹<http://www.dbpedia.org>

Table 8.2: Characteristics of the Movies and Actors datasets describing entities, movies, and actors in DBpedia.

Dataset Characteristics	Movies Dataset	Actors Dataset
Total entities	724	14
Total concepts	42	14
Total tasks	230	120
Average tasks per concept	9	8.6
Average concepts per task	1.64	1

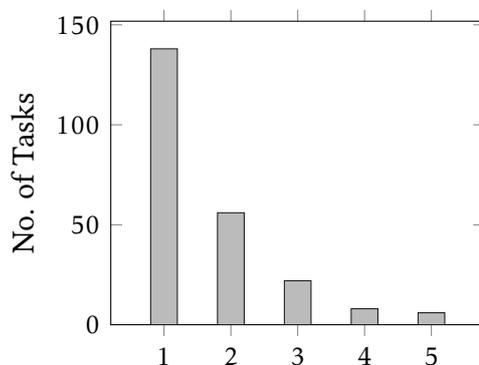


Figure 8.2: Distribution of the number of tasks against the number of topics in the Movies dataset.

based on genres for the Movies dataset and actor names for the Actors dataset. The choice of creating datasets from films has two advantages; it is relatively easy to recruit workers for the study and people have varying degree of knowledge about films depending on various factors such as genre, language, actors, etc. The Movies dataset was created by selecting Academy Award winning, Indian FilmFare Award winning, and top 100 grossing movies (from both Hollywood and Bollywood). DBpedia provides a variety of topic hierarchies for entities. We chose the 42 genres associated with the movies to serve as expertise topics. Each task consisted of a fact about a movie entity, where the genres of the movie served as the topics related to the task. The distribution of tasks against the number of topics is shown in Figure 8.2. The Actors dataset was also generated manually by

Table 8.3: Details of workers and tasks for the Movies and the Actors datasets.

	Movies Dataset	Actors Dataset
No. of workers	11	26
No. of topics	42	14
No. of test tasks (offline learning)	100	56
No. of new tasks (online assignment)	130	64

selecting 20 popular celebrities from Hollywood and 4 from Bollywood. In this case, the names of movie stars served as topics thus providing close relationship with their associated tasks. The objective of this selection was to facilitate easy association of expertise with the task response, during assessment. Similar to the Movies dataset, a task required feedback on a fact about actor entity.

Crowd Workers

The participants of the user study were recruited through an open call in a research institute. Separate calls were made for Movies and Actors datasets. The resulting two groups of workers consisted of participants coming from countries in Asia, Europe, and America. Since some workers were from South Asian countries, they possessed a higher expertise about topics and tasks related to Bollywood films. Table 8.3 summarizes both datasets in terms of (i) the number of volunteer workers recruited, (ii) the number of topics, (iii) the number of test tasks used for the offline learning phase, (iv) and the number of tasks assigned to workers during online assignment phase. During the data collection exercise, each worker was asked to perform both self-assessment and test tasks. Additionally, workers had to provide responses to the new tasks assigned to them after offline learning phase. The workers were asked to respond quickly and truthfully without looking up answers on the Web.

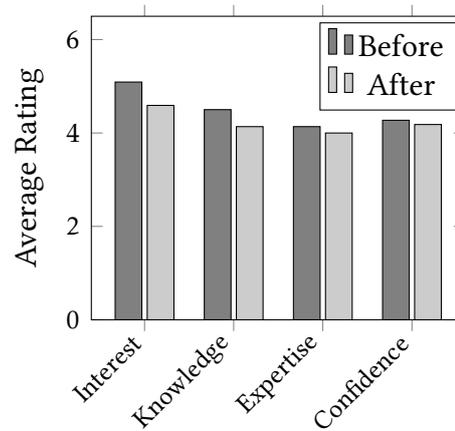


Figure 8.3: The average level of interest, knowledge, expertise and confidence of workers before and after the experiment for the Actors dataset.

Short surveys were performed before and after the experiment to measure changes in participant belief. In the both surveys 22 workers of the Actors dataset were asked to indicate, on 10 point belief scale, their level of (i) interest in tasks, (ii) knowledge about tasks, (ii) expertise of topics of tasks, and (iv) the confidence in their responses. Figure 8.3 shows the comparison of results of the survey. A paired t-test was performed to determine if the belief level of workers, for each question, changed after the experiment. The average loss in interest (mean=0.5, standard deviation=1.06, count=22) was significantly greater than zero, where $t(21)=2.22$ and $p=0.04$, providing the evidence that the experiment resulted in a decrease in interest of workers. A 95% confidence interval for average loss in interest is (0.03, 0.97). The average difference in the level of knowledge, expertise, and confidence was statistically not significant. Some participants indicated that they lost interest in the experiment due to a large number of tasks. This can also be attributed to the volunteered nature of tasks, although participants were given non-monetary rewards.

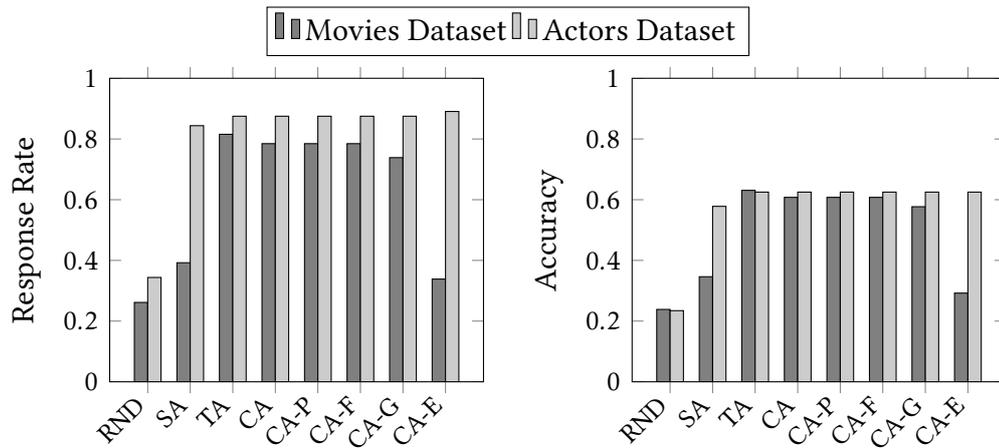


Figure 8.4: Graphs for the comparison of response rate and accuracy of all heuristics used for Movies and Actors datasets.

Evaluation Metrics

Following metrics were used to compare and contrast the utility of offline learning heuristics.

- **Response Rate** is the percentage of tasks with either “agree” or “disagree” response, out of all the tasks during the online assignment phase.
- **Accuracy** is the percentage of tasks with correct responses, out of all the tasks during the online assignment phase.
- **Workload** is the cognitive load on an individual worker during offline learning, in terms of decisions made by her. A decision is either self-rating of the expertise on a topic, or providing a response to a test task.

Next section discusses the results of the experiments performed using the data collected during user study. The number of workers per task was fixed such that $b = 1$.

8.5.2 Comparison of Accuracy & Response Rate

Figure 8.4 shows the comparison of all offline learning heuristics in terms of response rate and accuracy, for both datasets. For each heuristic, a two-sample t-test between metrics was performed to determine whether there was a significant difference between two datasets. As expected the response rate and accuracy of RND were minimum and TA strategies were maximum, with no significant difference between the two datasets. However, the metrics had statistically significant differences between datasets for the SA heuristic. The semantic relationship of topics (movie genres) and tasks (movie facts) was not strong for the Movies dataset; therefore workers claiming high knowledge were unable to respond correctly to the assigned tasks. Despite this observation, the performance of CA strategy was similar to that of TA, with no significant difference between datasets.

8.5.3 Comparison of Offline Learning Workload

The workload is quantified in terms of the number of decisions made by a worker during the offline learning. For example, the offline learning in SA required a worker to make 42 decisions rating their expertise on topics in the Movies dataset. The TA heuristics required 100 decisions of providing responses to the test tasks. Therefore, the workload required for expertise estimation by CA includes 42 topic rating decisions and 100 task responses. Clearly, there is an overhead associated with the combined-assessment approach. As shown in Figure 8.5, the maximum workload is attributed to the CA approach, .

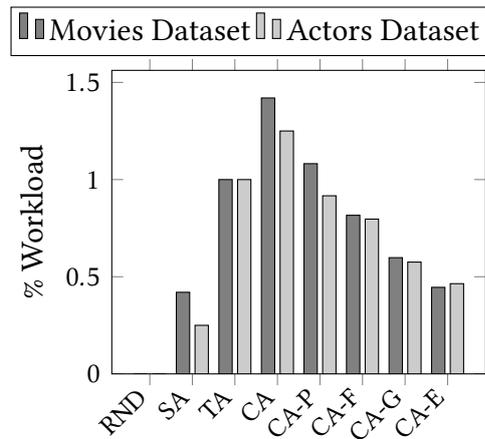


Figure 8.5: Relative workload, as compared to the TA approach, during the offline learning phase.

8.5.4 Effects of Filtering on Workload

To compensate for the extra workload, due to test tasks, filters were applied according to various levels of topic knowledge in combined approach. As a result, workers were not given test tasks with topics below a certain expertise level. As shown in Figure 8.4, the performance of filtering based heuristic is to TA, except for the CA-E. Table 8.4 compares the workload per worker for each heuristic (in terms of percentage as compared to CA) with the response rate, for both the datasets. The filtered CA assessment heuristics required comparatively less workload while achieving the accuracy close to TA. The response rate for CA-F and CA-G filtering does suffer during the online assignment phase. Nonetheless, the comparative decrease is less as compared to the rate of decrease in the average workload. However, in case of Movies Dataset the response rates drops below original SA approach if filtering is too restrictive i.e. CA-E during offline learning. Similar results were observed for accuracy versus average workload per worker.

Table 8.4: Relative performance of assessment approaches in comparison to the CA assessment approach.

(a) Movies Dataset

Approach	Workload	Response Rate	Accuracy
RND	-100.00%	-66.67%	-60.76%
SA	-70.42%	-50.00%	-43.04%
TA	-29.58%	3.92%	3.80%
CA	0.00%	0.00%	0.00%
CA-P	-23.82%	0.00%	0.00%
CA-F	-42.51%	0.00%	0.00%
CA-G	-57.87%	-5.88%	-5.06%
CA-E	-68.63%	-56.86%	-51.90%

(b) Actors Dataset

Approach	Workload	Response Rate	Accuracy
RND	-100.00%	-60.71%	-62.50%
SA	-80.00%	-3.57%	-7.50%
TA	-20.00%	0.00%	0.00%
CA	0.00%	0.00%	0.00%
CA-P	-26.67%	0.00%	0.00%
CA-F	-36.27%	0.00%	0.00%
CA-G	-53.97%	0.00%	0.00%
CA-E	-62.86%	1.79%	0.00%

8.6 Discussion

In terms of reducing the workload of offline assessment, the results in this chapter demonstrate the effectiveness of filtering-based combined approach of self-assessment and task-assessment. It is expected that the results are generalizable to other knowledge-intensive tasks such as tagging bird types in images, as opposed to observation based tasks such as comparing images. Compared to the previous

works on task assignment for crowdsourcing, our approach is distinguished by its use of topics for assessment, representation, and exploitation of workers' expertise. This unified approach provides a common framework for representing expertise requirements of a task and expertise profiles of workers. Thus allowing effective task assignment using topic matching. This approach is suitable for task assignment in knowledge intensive crowdsourcing.

The self-assessment allows workers to indicate their preferences over tasks. In our study, we found that the sequential process of responding to tasks for task-assessment can be tedious for workers. Therefore, limiting the number of test tasks based on self-assessment of knowledge is an effective strategy. The assignment decisions based on the resulting expertise profiles have similar response rate and accuracy; although the cost of building the expertise profile is much lower. Other application domains, such as scientific data management, that require a high diversity of knowledge among workers and across tasks can benefit from our approach.

The choice of topics to be used for profiling and assignment affects the quality of assignment to some extent. Although the general patterns of accuracy and response rate were same for various levels of filtering, there was a sharp decline for Movies dataset with very restrictive filters. In the case of Movies dataset, the topics were broader than the topics for Actors dataset. In Movies dataset, a task was related to the missing attribute of a Film and a topic was the genre of the same Film. In Actors dataset, a task was related to missing attribute of an Actor and the topic was the same Actor entity. Some workers felt confident about their excellent level of knowledge for some film genres but were unable to response to specific questions about films from those genres. This study used a relatively small number of topics; therefore, it would be interesting to study the scalability of the proposed

approach for a large number of topics. Possible strategies to handle large a number of topics include; using topic hierarchies, applying clustering techniques to group topics, or using the distribution of tasks to rank important topics.

8.7 Chapter Summary

This chapter introduced the cold-start problem for expertise based spatial crowd-sourcing. This problem formulates the explore-exploit trade-off and contextual learning requirements of adaptive task assignment using offline learning. The expertise requirements of tasks and expertise of workers were defined in terms of domain topics. The contextual learning is enabled through offline learning to estimate worker specific expertise coefficients on domain topics associated with tasks. This chapter studied the effects of three warm-start approaches, namely self-assessment; task-based assessment; and combined assessment, on the performance of an adaptive task assignment algorithm. It was observed that estimation of worker specific expertise coefficients through self-assessment is better than random estimation. Furthermore, the workload of offline learning with test tasks is reduced by filtering tasks according to self-assessment of expertise, without sacrificing the performance of the algorithm.

Chapter 9

Conclusion

9.1 Thesis Summary

Spatial crowdsourcing is becoming a popular paradigm for solving problems in the physical world. Its applications range from disaster management to city maintenance. Spatial crowdsourcing systems face three major research challenges: heterogeneity, dynamism, and uncertainty. Heterogeneity manifests itself in the form of different types of tasks and the variability of worker reliability. Dynamism stems from different times for task creation and expiry, as well as the intermittent availability of workers. Uncertainty results from the possibility that workers may not complete tasks assigned to them. Design and evaluation of task assignment algorithms that address these challenges are the focus of this thesis.

This thesis proposes a novel approach for adaptive task assignment in spatial crowdsourcing that addresses the major limitations of state-of-the-art dynamic task assignment approaches in spatial crowdsourcing. It formulates the adaptive

assignment problem as a combination of online learning and combinatorial optimization. It allows a broad variety of objective functions to optimize the utility of spatial crowdsourcing. The presented approach does not require prior knowledge about worker reliabilities to make intelligent assignment decisions. Instead, it enables heuristics-based online learning for estimation of worker reliabilities. It considers the spatial and non-spatial context for both optimization and learning in the assignment process. The generality of proposed approach is established by instantiating the adaptive assignment problem in a number of spatial crowdsourcing scenarios, as listed below:

- Chapter 4: Maximum reliability assignment scenario
- Chapter 5: Minimum cost maximum reliability assignment scenario
- Chapter 6: Maximum reliability assignment with context scenario
- Chapter 7: Maximum reliability semi-assignment scenario
- Chapter 8: Cold-start scenario

This thesis presents appropriate algorithms for each instantiated scenario. Experimental evaluation establishes the effectiveness of proposed algorithms on both synthetic and real-world datasets. The results show that the proposed algorithms achieve high average reliability, high task completion rate, low average travel costs, and low assessment workload. The rest of this chapter is structured as follows. Section 9.2 concludes the thesis. Section 9.3 summarizes the key contributes of this thesis towards spatial crowdsourcing research. Section 9.4 highlights the limitations of this work. Section 9.5 suggests areas that are worthy of future study.

9.2 Thesis Conclusion

This section analyzes the hypothesis statement and draws conclusions from the experimental evaluation.

Hypothesis Statement: *By formalizing the adaptive task assignment according to the combinatorial bandits framework and enabling contextual learning of reliability and expertise of workers according to spatial and non-spatial contextual information, effective assignment algorithms can be designed for multi-criteria optimization of spatial crowdsourcing.*

This thesis introduced the adaptive assignment problem in spatial crowdsourcing that considers dynamic task assignment under the observed knowledge assumption. It formalizes the trade-off between learning and optimization for enabling intelligent assignment decisions. Learning allows an approximation of unknown worker reliability. Optimization allows the best choice of assignments given an objective function. The thesis presents four scenarios to establish the applicability of adaptive assignment problem in spatial crowdsourcing. Each scenario considers different forms of constraints and contextual information. Adaptive task assignment algorithms are proposed for each scenario to address the research requirements:

- **Proposed algorithms effectively address the explore-exploit trade-off:** For each scenario, an adaptive task assignment algorithm is proposed and evaluated using real-world datasets. The results suggest that the proposed algorithms are effective in approximating the worker reliability while optimizing spatial crowdsourcing. The DRR-UCB algorithm uses an interval estimation heuristic to address the explore-exploit trade-off

(Chapter 5). It achieved 90% relative reliability when compared to the deterministic algorithm with known reliabilities. The DynTS algorithm uses a probability matching heuristic to address the explore-exploit trade-off (Chapter 7). It achieved 95% reliability on a synthetic dataset.

- **Proposed algorithms address the requirement of multi-criteria optimization:** Two scenarios required the design of assignment algorithms for multi-criteria optimization. The first scenario required maximization of worker reliabilities and minimization of travel costs. For this scenario, the DRR algorithm based on a distance reliability ratio approach was proposed. The DRR algorithm was compared with the state-of-the-art close distance priority algorithm. The results show that the DRR algorithm achieves higher task completion rate and lower average travel costs. Furthermore, the computational complexity of DRR is polynomial in the number of tasks and the number of workers. The second scenario required multiple workers assigned to each task for the sake of improving the task completion rate. For this scenario, the proposed DynTS algorithm employs a location diversity heuristic. The DynTS algorithm first calculates the number of workers required for each task using the entropy of the task location. Then, it chooses the best set of workers for each task while aiming to maximize the cumulative reliability. The DynTS algorithm is evaluated against algorithms with a fixed number of workers per task. The results indicate a high task completion rate with more than 10% relative reduction in assignments.
- **Proposed algorithms address the requirement of contextual learning:** Two scenarios required the design of assignment algorithms based on contextual learning. The first scenario necessitated an online learning heuristic

based on contextual variables. For this scenario, the proposed SpatialUCB algorithm uses an individualized linear model for online learning. The SpatialUCB is compared with the context-free algorithms using a real-world dataset. The results confirm that the SpatialUCB algorithm achieves 30%-50% higher task completion rate. The second scenario required an offline learning approach based on topics associated with tasks. The proposed approach combines self-rating with test tasks to estimate topic-specific worker expertise. Empirical evaluation compares the proposed approach with the baseline approach using real workers. The results confirm that the combined approach requires 18%-25% less exploration when compared to baseline. However, both approaches achieve a similar utility during exploitation.

In general, all the instantiation of the adaptive assignment problem involve scenarios with spatial tasks. All instantiations address the spatial context requirement of adaptive task assignment in spatial crowdsourcing.

9.3 Core Contributions

This thesis makes following core contributions:

- Introducing the adaptive assignment problem as a novel framework for adaptive task assignment in spatial crowdsourcing. The framework enables intelligent assignment decisions when worker reliabilities are unknown.
- Novel instantiations of the adaptive assignment problem under varying optimization objectives. Each instance of the adaptive assignment problem

also models a variation of contextual information available at the time of assignment.

This thesis also introduces the *contextual learning* and *multi-criteria optimization* along with the adaptive assignment problem. In this regards, the spatial context of tasks and workers are exploited to further improve the assignment process. As summarized by following contributions:

- A combinatorial fractional programming approach bi-objective optimization in spatial crowdsourcing.
- A contextualized learning approach for dynamic estimation of worker reliabilities based on contextual variables.
- A location diversity heuristic for dynamic redundancy in spatial crowdsourcing.
- A combined assessment approach for estimation of worker expertise on topics associated with tasks.

9.4 Limitations

Some dimensions of the adaptive task assignment are outside the scope of this thesis. The following list describes some of the dimensions:

- **Delayed observation of assignment outcomes:** The learning heuristics assume an immediate observation of assignment outcome at the end of a round. The case of delayed observation when a worker might complete a task after a round has ended was not considered.

- **Assignment based on task quality:** The algorithms proposed in this thesis consider task completion as the variable of assignment outcome. The utility of a task might also depend on the quality or diversity of data submitted by workers. The algorithms proposed in the thesis can be extended to take task quality or diversity into consideration.
- **Theoretical analysis:** The empirical performance of the proposed algorithms can be supported with a theoretical analysis. Such an analysis defines the upper or lower performance bounds of the proposed algorithms. The theoretical analysis includes competitive analysis for optimization and regret analysis for learning.
- **Incentive mechanisms:** The work presented in this thesis also does not consider the issues related to incentives in spatial crowdsourcing. Such issues play a central role in paid crowdsourcing systems, which require detailed investigation of the relationship between incentives and utility. Nonetheless, the task assignment approach presented is complementary to carefully designed incentive mechanisms.

9.5 Future Work

The contributions of this thesis focus on adaptive task assignment in spatial crowdsourcing. The detailed study of the problem revealed further research questions that were not addressed in this thesis. Some interesting areas for future research include the following questions:

- **Adversarial and highly dynamic worker pool:** The assignment algorithms proposed in this thesis make assumptions about the pool of workers.

First, the reliability of workers is stochastic and independent of each other as well as between assignments. This means that the assignment success probability for a worker is stable behavior over time. A natural direction of future research is to consider the adversarial reliability of workers. Under this assumption, workers adapt their reliability based on previously assigned tasks. For instance, a worker could become fatigued if assigned a significant number of tasks over time, or a worker might stay for a long time in a fixed location doing nearby tasks repeatedly. Second, the pool of available workers is assumed to be relatively stable over time. Few workers enter or leave the pool over many rounds of assignment. The effects of the temporal changes in the worker pool on the performance of algorithms require further investigation.

- **Consideration of additional features and constraints:** The thesis constrained the number of tasks to one per worker during a round. A natural extension is to consider many tasks per worker given capacity constraints. Assignments in the form of task chains can help workers perform tasks in one session which leads to optimal usage of workers time. It should be noted that the dynamic assignment with task chains is an NP-Hard problem. Consideration of additional constraints such as real-time completion, amount of physical effort, monetary budgets, and maximum delays, further increase the complexity of the problem.
- **Theoretical analysis of fully dynamic assignment:** The thesis contributes an experimental evaluation for fully-dynamic assignment algorithms in spatial crowdsourcing. But, the theoretical analysis of such algorithms is still an open research problem. Establishing theoretical bounds

on the performance of dynamic assignment algorithms is a significant contribution. Such analysis for both probabilistic and observed knowledge assumptions is needed. The algorithms proposed in the thesis in general recompute the complete assignment set during each round irrespective of assignment set in previous rounds. An interesting area for further research is to study the maintenance of an optimal assignment set over multiple rounds with minimum changes. In this case, the assignment algorithm focuses on selecting assignments based on changes to task and worker sets between rounds.

- **Real-world evaluation of algorithms:** In general, the evaluation methodology in this thesis follows an agent-based simulation approach for establishing the performance of the proposed algorithms. Evaluation using real workers and tasks would provide a better understanding of the proposed algorithms. However, running such experiments is still a very expensive and time-consuming process. Nonetheless, the relative performance of algorithm would depend on characteristics of worker population and duration of experiments. It is expected that the distribution of worker reliabilities would be heavy-tailed. It would be interesting to combine server assigned tasks algorithms with worker selected tasks mechanisms to investigate the best approaches for task assignment in spatial crowdsourcing.

CONCLUSION

Bibliography

- [1] Mark Weiser. The computer for the 21st century. *Scientific american*, 265(3): 94–104, 1991.
- [2] Adam Greenfield. *Everyware: The dawning age of ubiquitous computing*. New Riders, 2010.
- [3] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.
- [4] Edith Law and Luis von Ahn. Human computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(3):1–121, 2011.
- [5] Abraham Bernstein, Mark Klein, and Thomas W. Malone. Programming the global brain. *Communications of the ACM*, 55(5):41, 2012.
- [6] Franco Zambonelli. Pervasive urban crowdsourcing: Visions and challenges. In *Proceedings of the Pervasive Computing and Communications Workshops*, pages 578–583. IEEE, 2011.
- [7] David Alan Grier. *When Computers Were Human*. Princeton University Press, Princeton, NJ, USA, 2007. ISBN 9780691133829.
- [8] Joseph Carl Robnett Licklider. Man-computer symbiosis. *IRE Transactions on Human Factors in Electronics*, HFE-1(1):4–11, 1960.

- [9] Jeff Howe. The rise of crowdsourcing. *Wired Magazine*, 14(6):1–4, 2006.
- [10] James Surowiecki. *The wisdom of crowds*. Anchor, 2005.
- [11] Don Tapscott and Anthony D Williams. *Wikinomics: How mass collaboration changes everything*. Penguin, 2008.
- [12] Luis Von Ahn, Manuel Blum, Nicholas J Hopper, and John Langford. Captcha: Using hard ai problems for security. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 294–311. Springer, 2003.
- [13] Leyla Kazemi and Cyrus Shahabi. Geocrowd: enabling query answering with spatial crowdsourcing. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, pages 189–198. ACM, 2012.
- [14] Robert Munro. Crowdsourcing and the crisis-affected community. *Information Retrieval*, 16(2):210–266, 2013.
- [15] Annice E Kim, Alicea J Lieberman, and Daniel Dench. Crowdsourcing data collection of the retail tobacco environment: case study comparing data from crowdsourced workers to trained data collectors. *Tobacco control*, pages tobaccocontrol–2013, 2014.
- [16] Rannie Teodoro, Pinar Ozturk, Mor Naaman, Winter Mason, and Janne Lindqvist. The motivations and experiences of the on-demand mobile workforce. In *Proceedings of the 17th ACM Conference on Computer supported cooperative work & social computing*, pages 236–247. ACM, 2014.

- [17] Adam Sadilek, John Krumm, and Eric Horvitz. Crowdphysics: Planned and opportunistic crowdsourcing for physical tasks. In *International AAAI Conference on Web and Social Media*, 2013.
- [18] Taewoo Nam, Francisco Armando Aldama, Hamed Chourabi, Sehl Mellouli, Theresa A Pardo, J Ramon Gil-Garcia, Hans Jochen Scholl, Adegboyega Ojo, Elsa Estevez, and Lei Zheng. Smart cities and service integration. In *Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times*, pages 333–334. ACM, 2011.
- [19] Edward Curry, Schahram Dustdar, Quan Z Sheng, and Amit Sheth. Smart cities–enabling services and applications. *Journal of Internet Services and Applications*, 7(1):1, 2016.
- [20] Mohamed Musthag and Deepak Ganesan. Labor dynamics in a mobile micro-task market. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 641–650. ACM, 2013.
- [21] Aleksandrs Slivkins and Jennifer W Vaughan. Online decision making in crowdsourcing markets: Theoretical challenges. *ACM SIGecom Exchanges*, 12(2):4–23, 2013.
- [22] Aniket Kittur, Jeffrey V Nickerson, Michael Bernstein, Elizabeth Gerber, Aaron Shaw, John Zimmerman, Matt Lease, and John Horton. The future of crowd work. In *Proceedings of the ACM 2013 Conference on Computer Supported Cooperative Work*, pages 1301–1318. ACM, 2013.
- [23] Edith Law, Paul N Bennett, and Eric Horvitz. The effects of choice in routing relevance judgments. In *Proceedings of the 34th International ACM SIGIR*

- Conference on Research and Development in Information Retrieval*, pages 1127–1128. ACM, 2011.
- [24] Jacob Thebault-Spieker, Loren G Terveen, and Brent Hecht. Avoiding the south side and the suburbs: The geography of mobile crowdsourcing markets. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 265–275. ACM, 2015.
- [25] Hien To, Cyrus Shahabi, and Leyla Kazemi. A server-assigned spatial crowdsourcing framework. *ACM Transactions on Spatial Algorithms and Systems*, 2015.
- [26] Cen Chen, Shih-Fen Cheng, Hoong Chuin Lau, and Archan Misra. Towards city-scale mobile crowdsourcing: Task recommendations under trajectory uncertainties. In *International Joint Conferences on Artificial Intelligence*, 2015.
- [27] Anand Kulkarni, Philipp Gutheim, Prayag Narula, David Rolnitzky, Tapan Parikh, and Björn Hartmann. Mobileworks: Designing for quality in a managed crowdsourcing architecture. *IEEE Internet Computing*, 16(5):28–35, 2012.
- [28] James Albrecht. Search theory: the 2010 nobel memorial prize in economic sciences. *The Scandinavian Journal of Economics*, 113(2):237–259, 2011.
- [29] Dale T Mortensen. Markets with search friction and the dmp model. *The American Economic Review*, pages 1073–1091, 2011.
- [30] Anand Kulkarni, Matthew Can, and Björn Hartmann. Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the ACM*

- 2012 *Conference on Computer Supported Cooperative Work*, pages 1003–1012. ACM, 2012.
- [31] Yohan Chon, Nicholas D Lane, Yunjong Kim, Feng Zhao, and Hojung Cha. Understanding the coverage and scalability of place-centric crowdsensing. In *Proceedings of the ACM International joint Conference on Pervasive and ubiquitous computing*, pages 3–12. ACM, 2013.
- [32] Umair Ul Hassan and Edward Curry. A Multi-armed Bandit Approach to Online Spatial Task Assignment. In *Proceedings of the 11th IEEE International Conference on Ubiquitous Intelligence and Computing*, pages 212–219. IEEE, 2014.
- [33] Umair Ul Hassan and Edward Curry. Flag-Verify-Fix: Adaptive Spatial Crowdsourcing leveraging Location-based Social Networks. In *Proceedings of the 23rd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 79:1–79:4, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3967-4.
- [34] Umair Ul Hassan and Edward Curry. Efficient Task Assignment for Spatial Crowdsourcing: A Combinatorial Fractional Optimization Approach with Semi-bandit Learning. *Expert Systems with Applications*, 58:36–56, 2016.
- [35] David R Karger, Sewoong Oh, and Devavrat Shah. Budget-optimal crowdsourcing using low-rank matrix approximations. In *49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 284–291. IEEE, 2011.
- [36] Dingxiong Deng, Cyrus Shahabi, and Ugur Demiryurek. Maximizing the number of worker’s self-selected tasks in spatial crowdsourcing.

- In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 314–323. ACM, 2013.
- [37] Hien To, Gabriel Ghinita, and Cyrus Shahabi. A framework for protecting worker location privacy in spatial crowdsourcing. *Proceedings of the VLDB Endowment*, 7(10), 2014.
- [38] Peng Cheng, Xiang Lian, Zhao Chen, Lei Chen, Jinsong Han, and Jizhong Zhao. Reliable diversity-based spatial crowdsourcing by moving workers. *Proceedings of the VLDB Endowment*, 8, 2015.
- [39] David Geiger and Martin Schader. Personalized task recommendation in crowdsourcing information systems—current state of the art. *Decision Support Systems*, 65:3–16, 2014.
- [40] Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 1998.
- [41] Daqing Zhang, Leye Wang, Haoyi Xiong, and Bin Guo. 4w1h in mobile crowd sensing. *IEEE Communications Magazine*, 52(8):42–48, 2014.
- [42] Chris Van Pelt and Alex Sorokin. Designing a scalable crowdsourcing platform. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 765–766. ACM, 2012.
- [43] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems—Volume 1*, pages 467–474. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

- [44] Rhonda Righter. A resource allocation problem in a random environment. *Operations Research*, 37(2):329–338, 1989.
- [45] Xuanming Su and Stefanos A Zenios. Patient choice in kidney allocation: A sequential stochastic assignment model. *Operations research*, 53(3):443–455, 2005.
- [46] S Bubeck, N Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.
- [47] Long Tran-Thanh, Sebastian Stein, Alex Rogers, and Nicholas R Jennings. Efficient crowdsourcing of unknown experts using bounded multi-armed bandits. *Artificial Intelligence*, 2014.
- [48] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 207–216. IEEE, 2013.
- [49] Alexey Tarasov, Sarah Jane Delany, and Brian Mac Namee. Dynamic estimation of worker reliability in crowdsourcing for regression tasks: Making it work. *Expert Systems with Applications*, 41(14):6190–6210, 2014.
- [50] Ittai Abraham, Omar Alonso, Vasilis Kandyas, and Aleksandrs Slivkins. Adaptive crowdsourcing algorithms for the bandit survey problem. In *Conference on Learning Theory*, pages 882–910, 2013.
- [51] Chien-Ju Ho and Jennifer Wortman Vaughan. Online task assignment in crowdsourcing markets. In *AAAI Conference on Artificial Intelligence*, 2012.

- [52] Chien-Ju Ho, Shahin Jabbari, and Jennifer W Vaughan. Adaptive task assignment for crowdsourced classification. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 534–542, 2013.
- [53] Cen Chen, Shih-Fen Cheng, Aldy Gunawan, Archan Misra, Koustuv Dasgupta, and Deepthi Chander. Traccs: Trajectory-aware coordinated urban crowd-sourcing. In *Proceedings of the 2nd AAAI Conference on Human Computation and Crowdsourcing*, 2014.
- [54] David R Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In *Advances in Neural Information Processing Systems*, pages 1953–1961, 2011.
- [55] Warren B Powell and Peter Frazier. Optimal learning. *Tutorials in Operations Research*, pages 213–246, 2008.
- [56] Warren B Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality*, volume 703. John Wiley & Sons, 2007.
- [57] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.
- [58] Yohan Chon, Nicholas D Lane, Yunjong Kim, Feng Zhao, and Hojung Cha. A large-scale study of mobile crowdsourcing with smartphones for urban sensing applications. In *Proc. of ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp’13), Zurich, Switzerland*, 2013.
- [59] Umair Ul Hassan, Sean O’Riain, and Edward Curry. Effects of Expertise

- Assessment on the Quality of Task Routing in Human Computation. In *Proceedings of the 2nd International Workshop on Social Media for Crowdsourcing and Human Computation*. BCS, 2013.
- [60] Umair Ul Hassan and Edward Curry. A Capability Requirements Approach for Predicting Worker Performance in Crowdsourcing. In *Proceedings of the 11th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 429–437. IEEE, 2013.
- [61] Aranyak Mehta and Debmalya Panigrahi. Online matching with stochastic rewards. In *IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 728–737. IEEE, 2012.
- [62] Aranyak Mehta, Bo Waggoner, and Morteza Zadimoghaddam. Online stochastic matching with unequal probabilities. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1388–1404. SIAM, 2015.
- [63] Joannes Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In *Machine Learning: ECML 2005*, pages 437–448. Springer, 2005.
- [64] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.
- [65] Lihong Li, Wei Chu, John Langford, and Xuanhui Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proceedings of the fourth ACM International Conference on Web search and Data mining*, pages 297–306. ACM, 2011.

- [66] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pages 661–670. ACM, 2010.
- [67] Umair Ul Hassan, Murilo Bassora, Ali H Vahid, Sean O’Riain, and Edward Curry. A Collaborative Approach for Metadata Management for Internet of Things: Linking Micro Tasks with Physical Objects. In *Proceedings of the 9th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 593–598. IEEE, 2013.
- [68] Umair Ul Hassan, Sean O’Riain, and Edward Curry. SLUA: Towards Semantic Linking of Users with Actions in Crowdsourcing. In *Proceedings of the International Workshop on Crowdsourcing the Semantic Web*, pages 32–44, 2013.
- [69] Umair Ul Hassan, Sean O’Riain, and Edward Curry. Towards Expertise Modelling for Routing Data Cleaning Tasks within a Community of Knowledge Workers. In *Proceedings of the 17th International Conference on Information Quality*, 2012.
- [70] Umair Ul Hassan, Sean O’Riain, and Edward Curry. Leveraging Matching Dependencies for Guided User Feedback in Linked Data Applications. In *Proceedings of the 9th International Workshop on Information Integration on the Web*, pages 1–6. ACM Press, 2012.
- [71] Anhai Doan, Raghu Ramakrishnan, and Alon Y Halevy. Crowdsourcing systems on the world-wide web. *Communications of the ACM*, 54(4):86–96, 2011.

- [72] José María Cavanillas, Edward Curry, and Wolfgang Wahlster, editors. *New Horizons for a Data-Driven Economy - A Roadmap for Usage and Exploitation of Big Data in Europe*. Springer, 2016. ISBN 978-3-319-21568-6.
- [73] André Freitas and Edward Curry. Big data curation. In *New Horizons for a Data-Driven Economy*, pages 87–118. Springer, 2016.
- [74] Alan M Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.
- [75] Dafna Shahaf and Eyal Amir. Towards a theory of ai completeness. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, pages 150–155, 2007.
- [76] Roman V Yampolskiy. Turing test as a defining feature of ai-completeness. In *Artificial Intelligence, Evolutionary Computing and Metaheuristics*, pages 3–17. Springer, 2013.
- [77] Luis Von Ahn, Manuel Blum, and John Langford. Telling humans and computers apart automatically. *Communications of the ACM*, 47(2):56–60, 2004.
- [78] Luis Von Ahn and Laura Dabbish. Designing games with a purpose. *Communications of the ACM*, 51(8):58–67, 2008.
- [79] Rion Snow, Brendan O’Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics, 2008.

- [80] Jeffrey Goldman, Katie Shilton, Jeff Burke, Deborah Estrin, Mark Hansen, Nithya Ramanathan, Sasank Reddy, Vids Samanta, Mani Srivastava, and Ruth West. Participatory sensing: A citizen-powered approach to illuminating the patterns that shape our world. *Foresight & Governance Project, White Paper*, pages 1–15, 2009.
- [81] David M Aanensen, Derek M Huntley, Edward J Feil, Fada'a al Own, and Brian G Spratt. Epicollect: linking smartphones to web applications for epidemiology, ecology and community data collection. *PloS one*, 4(9):e6968, 2009.
- [82] David N Crowley, Edward Curry, and John G Breslin. Closing the loop: From citizen sensing to citizen actuation. In *IEEE 7th International Conference on Digital Ecosystems and Technologies (DEST)*, pages 108–113. IEEE, 2013.
- [83] Souleiman Hasan, Richard Medland, Marcus Foth, and Edward Curry. Curbing resource consumption using team-based feedback. In *International Conference on Persuasive Technology*, pages 75–86. Springer, 2013.
- [84] Tobias Franke, Paul Lukowicz, Martin Wirz, and Eve Mitleton-Kelly. Participatory sensing and crowd management in public spaces. In *Proceeding of the 11th annual International Conference on Mobile systems, applications, and services*, pages 485–486. ACM, 2013.
- [85] Viktoriya Degeler and Edward Curry. Human-assisted rule satisfaction in partially observable environments. In *Proceedings of the 11th IEEE International Conference on Ubiquitous Intelligence and Computing*, pages 171–178. IEEE, 2014.

- [86] Hung Dang, Tuan Nguyen, and Hien To. Maximum complex task assignment: Towards tasks correlation in spatial crowdsourcing. In *Proceedings of the International Conference on Information Integration and Web-based Applications & Services*, page 77. ACM, 2013.
- [87] Nicholas D Lane, Yohan Chon, Lin Zhou, Yongzhe Zhang, Fan Li, Dongwon Kim, Guanzhong Ding, Feng Zhao, and Hojung Cha. Piggyback crowdsensing (pcs): energy efficient crowdsourcing of mobile sensor data by exploiting smartphone app opportunities. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 7. ACM, 2013.
- [88] Nicholas D Lane, Shane B Eisenman, Mirco Musolesi, Emiliano Miluzzo, and Andrew T Campbell. Urban sensing systems: opportunistic or participatory? In *Proceedings of the 9th workshop on Mobile Computing Systems and applications*, pages 11–16. ACM, 2008.
- [89] Minh Shin, Cory Cornelius, Dan Peebles, Apu Kapadia, David Kotz, and Nikos Triandopoulos. Anonymsense: A system for anonymous opportunistic sensing. *Pervasive and Mobile Computing*, 7(1):16–30, 2011.
- [90] Andrew T Campbell, Shane B Eisenman, Nicholas D Lane, Emiliano Miluzzo, Ronald A Peterson, Hong Lu, Xiao Zheng, Mirco Musolesi, Kristóf Fodor, and Gahng-Seop Ahn. The rise of people-centric sensing. *IEEE Internet Computing*, 12(4):12–21, 2008.
- [91] Joel Ross. Pervasive human computing. In *Handbook of Human Computation*, pages 333–345. Springer, 2013.
- [92] Xiaoyong Liu, W Bruce Croft, and Matthew Koll. Finding experts in community-based question-answering services. In *Proceedings of the 14th*

- ACM International Conference on Information and Knowledge Management*, pages 315–316. ACM, 2005.
- [93] Dan Cosley, Dan Frankowski, Loren Terveen, and John Riedl. Suggestbot: using intelligent task routing to help people find work in wikipedia. In *Proceedings of the 12th International Conference on Intelligent User Interfaces*, pages 32–41. ACM, 2007.
- [94] Jinwen Guo, Shengliang Xu, Shenghua Bao, and Yong Yu. Tapping on the potential of q&a community by recommending answer providers. In *Proceedings of the 17th ACM Conference on Information and knowledge management*, pages 921–930. ACM, 2008.
- [95] Dawei Hu, Shenhua Gu, Shitong Wang, Liu Wenyin, and Enhong Chen. Question recommendation for user-interactive question answering systems. In *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication*, pages 39–44. ACM, 2008.
- [96] Mingcheng Qu, Guang Qiu, Xiaofei He, Cheng Zhang, Hao Wu, Jiajun Bu, and Chun Chen. Probabilistic question recommendation for question answering communities. In *Proceedings of the 18th International Conference on World Wide Web*, pages 1229–1230. ACM, 2009.
- [97] Yanhong Zhou, Gao Cong, Bin Cui, Christian S Jensen, and Junjie Yao. Routing questions to the right users in online communities. In *IEEE 25th International Conference on Data Engineering (ICDE'09)*, pages 700–711. IEEE, 2009.
- [98] Y Connie Yuan, Dan Cosley, Howards T Welser, Ling Xia, and Geri Gay. The diffusion of a task recommendation system to facilitate contributions

- to an online community. *Journal of Computer-Mediated Communication*, 15 (1):32–59, 2009.
- [99] Hani Jamjoom, Huiming Qu, Melissa J Bucu, Milton Hernandez, Debanjan Saha, and Mahmoud Naghshineh. Crowdsourcing and service delivery. *IBM Journal of Research and Development*, 53(6):12–1, 2009.
- [100] Baichuan Li and Irwin King. Routing questions to appropriate answerers in community question answering services. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1585–1588. ACM, 2010.
- [101] Wei-Chen Kao, Duen-Ren Liu, and Shiu-Wen Wang. Expert finding in question-answering websites: a novel hybrid approach. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 867–871. ACM, 2010.
- [102] Mingrong Liu, Yicen Liu, and Qing Yang. Predicting best answerers for new questions in community question answering. In *Web-Age Information Management*, pages 127–138. Springer, 2010.
- [103] Carlos Castro-Herrera. A hybrid recommender system for finding relevant users in open source forums. In *Third International Workshop on Managing Requirements Knowledge (MARK)*, pages 41–50. IEEE, 2010.
- [104] Damon Horowitz and Sepandar D Kamvar. The anatomy of a large-scale social search engine. In *Proceedings of the 19th International Conference on World Wide Web*, pages 431–440. ACM, 2010.
- [105] Peter Welinder, Steve Branson, Pietro Perona, and Serge J Belongie. The

- multidimensional wisdom of crowds. In *Advances in Neural Information Processing Systems*, pages 2424–2432, 2010.
- [106] Gideon Dror, Yehuda Koren, Yoelle Maarek, and Idan Szpektor. I want to answer; who has a question?: Yahoo! answers recommender system. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1109–1117. ACM, 2011.
- [107] Hengshu Zhu, Enhong Chen, and Huanhuan Cao. Finding experts in tag based knowledge sharing communities. In *Knowledge Science, Engineering and Management*, pages 183–195. Springer, 2011.
- [108] Vamshi Ambati, Stephan Vogel, and Jaime G Carbonell. Towards task recommendation in micro-task markets. In *Human Computation*, pages 1–4, 2011.
- [109] Fatemeh Riahi, Zainab Zolaktaf, Mahdi Shafiei, and Evangelos Milios. Finding expert users in community question answering. In *Proceedings of the 21st International Conference Companion on World Wide Web*, pages 791–798. ACM, 2012.
- [110] Tom Chao Zhou, Michael R Lyu, and Irwin King. A classification-based approach to question routing in community question answering. In *Proceedings of the 21st International Conference Companion on World Wide Web*, pages 783–790. ACM, 2012.
- [111] Zhenlei Yan and Jie Zhou. A new approach to answerer recommendation in community question answering services. In *Advances in Information Retrieval*, pages 121–132. Springer, 2012.

- [112] Gang Liu and Tianyong Hao. User-based question recommendation for question answering system. *International Journal of Information and Education Technology*, 2(3):243, 2012.
- [113] Haoqi Zhang, Eric Horvitz, Yiling Chen, and David C Parkes. Task routing for prediction tasks. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 889–896. International Foundation for Autonomous Agents and Multiagent Systems, 2012.
- [114] Benjamin Satzger, Harald Psailer, Daniel Schall, and Schahram Dustdar. Auction-based crowdsourcing supporting skill management. *Information Systems*, 38(4):547–560, 2013.
- [115] Gagan Goel, Afshin Nikzad, and Adish Singla. Allocating tasks to workers with matching constraints: truthful mechanisms for crowdsourcing markets. In *Proceedings of the Companion publication of the 23rd International Conference on World Wide Web Companion*, pages 279–280. International World Wide Web Conferences Steering Committee, 2014.
- [116] Christopher H Lin, Ece Kamar, and Eric Horvitz. Signals in the silence: Models of implicit feedback in a recommendation system for crowdsourcing. In *AAAI*, pages 908–915, 2014.
- [117] Hyun Joon Jung. Quality assurance in crowdsourcing via matrix factorization based task routing. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, pages 3–8. International World Wide Web Conferences Steering Committee, 2014.

- [118] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. Taskrec: A task recommendation framework in crowdsourcing systems. *Neural Processing Letters*, 41(2):223–238, 2015.
- [119] Leyla Kazemi. *Enabling query answering in a trustworthy privacy-aware spatial crowdsourcing*. PhD thesis, University of Southern California, 2012.
- [120] Long Tran-Thanh, Matteo Venanzi, Alex Rogers, and Nicholas R Jennings. Efficient budget allocation with accuracy guarantees for crowdsourcing classification tasks. In *Proceedings of the 2013 International Conference on Autonomous agents and multi-agent systems*, pages 901–908. International Foundation for Autonomous Agents and Multiagent Systems, 2013.
- [121] Ioannis Boutsis and Vana Kalogeraki. Crowdsourcing under real-time constraints. In *IEEE 27th International Symposium on Parallel & Distributed Processing*, pages 753–764. IEEE, 2013.
- [122] Xi Chen, Qihang Lin, and Dengyong Zhou. Optimistic knowledge gradient policy for optimal budget allocation in crowdsourcing. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 64–72, 2013.
- [123] Djellel Eddine Difallah, Gianluca Demartini, and Philippe Cudré-Mauroux. Pick-a-crowd: Tell me what you like, and i’ll tell you what to do. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 367–374. International World Wide Web Conferences Steering Committee, 2013.
- [124] Leyla Kazemi, Cyrus Shahabi, and Lei Chen. Geotrucrowd: trustworthy query answering with spatial crowdsourcing. In *Proceedings of the 21st ACM*

- SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 314–323. ACM, 2013.
- [125] Ioannis Boutsis and Vana Kalogeraki. On task assignment for real-time reliable crowdsourcing. In *IEEE 34th International Conference on Distributed Computing Systems (ICDCS)*, pages 1–10. IEEE, 2014.
- [126] Long Tran-Thanh, Trung Dong Huynh, Avi Rosenfeld, Sarvapali D Ramchurn, and Nicholas R Jennings. Budgetfix: budget limited crowdsourcing for interdependent task allocation with quality guarantees. In *Proceedings of the 2014 International Conference on Autonomous agents and multi-agent systems*, pages 477–484. International Foundation for Autonomous Agents and Multiagent Systems, 2014.
- [127] Shibo He, Dong-Hoon Shin, Junshan Zhang, and Jiming Chen. Toward optimal allocation of location dependent tasks in crowdsensing. In *Proceedings of the IEEE INFOCOM Conference*, pages 745–753. IEEE, 2014.
- [128] Daqing Zhang, Haoyi Xiong, Leye Wang, and Guanling Chen. Crowdrecriuter: Selecting participants for piggyback crowdsensing under probabilistic coverage constraint. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 703–714. ACM, 2014.
- [129] Edwin Simpson and Stephen Roberts. Bayesian methods for intelligent task assignment in crowdsourcing systems. In *Decision Making: Uncertainty, Imperfection, Deliberation and Scalability*, pages 1–32. Springer, 2015.
- [130] Ju Fan, Guoliang Li, Beng Chin Ooi, Kian-lee Tan, and Jianhua Feng. icrowd: An adaptive crowdsourcing framework. In *Proceedings of the 2015 ACM*

- SIGMOD International Conference on Management of Data*, pages 1015–1030. ACM, 2015.
- [131] Senjuti Basu Roy, Ioanna Lykourantzou, Saravanan Thirumuruganathan, Sihem Amer-Yahia, and Gautam Das. Task assignment optimization in knowledge-intensive crowdsourcing. *The VLDB Journal*, 24(4):467–491, 2015.
- [132] Leye Wang, Daqing Zhang, Animesh Pathak, Chao Chen, Haoyi Xiong, Dingqi Yang, and Yasha Wang. Ccs-ta: quality-guaranteed online task allocation in compressive crowdsensing. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 683–694. ACM, 2015.
- [133] Haoyi Xiong, Daqing Zhang, Guanling Chen, Leye Wang, and Vincent Gauthier. Crowdtasker: Maximizing coverage quality in piggyback crowdsensing under budget constraint. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 55–62. IEEE, 2015.
- [134] Hong Yao, Changkai Zhang, Chao Liu, Qingzhong Liang, Xuesong Yan, and Chengyu Hu. On participant selection for minimum cost participatory urban sensing with guaranteed quality of information. In *Collaborative Computing: Networking, Applications, and Worksharing*, pages 183–194. Springer, 2015.
- [135] Mingjun Xiao, Jie Wu, Liusheng Huang, Yunsheng Wang, and Cong Liu. Multi-task assignment for crowdsensing in mobile social networks. In *Proceedings of the IEEE INFOCOM Conference*. IEEE, 2015.

- [136] Man Hon Cheung, Richard Southwell, Fen Hou, and Jianwei Huang. Distributed time-sensitive task selection in mobile crowdsensing. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 157–166. ACM, 2015.
- [137] Xin Miao, Kebin Liu, Lei Chen, and Yunhao Liu. Quality-aware online task assignment in mobile crowdsourcing. In *IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pages 127–135. IEEE, 2015.
- [138] Chongyu Zhou, Chen-Khong Tham, and Mehul Motani. Qoata: Qoi-aware task allocation scheme for mobile crowdsensing under limited budget. In *IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pages 1–6. IEEE, 2015.
- [139] Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, 2001.
- [140] David W Pentico. Assignment problems: A golden anniversary survey. *European Journal of Operational Research*, 176(2):774–793, 2007.
- [141] Rainer E Burkard, Mauro Dell’Amico, and Silvano Martello. *Assignment Problems, Revised Reprint*. Siam, 2009.
- [142] D F Votaw and Alex Orden. The personnel assignment problem. In Alex Orden and Leon Goldstein, editors, *Symposium on Linear Inequalities and Programmng*, pages 155–163. Scientific Computation of Optimum Programs: Project SCOOP, US Air Force, 1952.
- [143] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

- [144] Richard M Karp, Umesh V Vazirani, and Vijay V Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing*, pages 352–358. ACM, 1990.
- [145] Jon Feldman, Aranyak Mehta, Vahab Mirrokni, and S Muthukrishnan. Online stochastic matching: Beating $1-1/e$. In *IEEE 50th Annual Symposium on Foundations of Computer Science*, pages 117–126. IEEE, 2009.
- [146] Yajun Wang and Sam Chiu-wai Wong. Two-sided online bipartite matching and vertex cover: Beating the greedy algorithm. In *Automata, Languages, and Programming*, pages 1070–1081. Springer, 2015.
- [147] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *Journal of the ACM (JACM)*, 54(5):22, 2007.
- [148] Gagan Aggarwal, Gagan Goel, Chinmay Karande, and Aranyak Mehta. Online vertex-weighted bipartite matching and single-bid budgeted allocations. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 1253–1264. SIAM, 2011.
- [149] Bala Kalyanasundaram and Kirk Pruhs. Online weighted matching. *Journal of Algorithms*, 14(3):478–488, 1993.
- [150] Samir Khuller, Stephen G Mitchell, and Vijay V Vazirani. On-line algorithms for weighted bipartite matching and stable marriages. *Theoretical Computer Science*, 127(2):255–267, 1994.
- [151] Bernhard Haeupler, Vahab S Mirrokni, and Morteza Zadimoghaddam. Online stochastic weighted matching: Improved approximation algorithms. In *Internet and Network Economics*, pages 170–181. Springer, 2011.

- [152] Michael Z Spivey and Warren B Powell. The dynamic assignment problem. *Transportation Science*, 38(4):399–419, 2004.
- [153] Saeed Alaei, MohammadTaghi Hajiaghayi, and Vahid Liaghat. The online stochastic generalized assignment problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 11–25. Springer, 2013.
- [154] Victor Pillac, Michel Gendreau, Christelle Guéret, and Andrés L Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11, 2013.
- [155] Aranyak Mehta. Online matching and ad allocation. *Foundations and Trends® in Theoretical Computer Science*, 8(4):265–368, 2013.
- [156] Ofer Dekel, Claudio Gentile, and Karthik Sridharan. Selective sampling and active learning from single and multiple teachers. *The Journal of Machine Learning Research*, 13(1):2655–2697, 2012.
- [157] Les E Atlas, David A Cohn, and Richard E Ladner. Training connectionist networks with queries and selective sampling. In *Advances in Neural Information Processing Systems*, pages 566–573, 1990.
- [158] Lester R Ford and Delbert R Fulkerson. Maximal flow through a network. *Canadian journal of Mathematics*, 8(3):399–404, 1956.
- [159] Aaron Bernstein and Cliff Stein. Fully dynamic matching in bipartite graphs. In *Automata, Languages, and Programming*, pages 167–179. Springer, 2015.
- [160] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit:

- General framework and applications. In *Proceedings of the 30th International Conference on Machine Learning*, pages 151–159, 2013.
- [161] Tian Lin, Bruno Abrahao, Robert Kleinberg, John Lui, and Wei Chen. Combinatorial partial monitoring game with linear feedback and its applications. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 901–909, 2014.
- [162] Tyler Lu, Dávid Pál, and Martin Pál. Contextual multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 485–492, 2010.
- [163] Wei Chu, Lihong Li, Lev Reyzin, and Robert E Schapire. Contextual bandits with linear payoff functions. In *International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.
- [164] Lihong Li, Wei Chu, John Langford, Taesup Moon, and Xuanhui Wang. An unbiased offline evaluation of contextual bandit algorithms with generalized linear models. In *JMLR Workshop and Conference Proceedings*, volume 26, pages 19–36. JMLR, 2012.
- [165] Dandan Zhang, Guangming Xie, Junzhi Yu, and Long Wang. Adaptive task assignment for multiple mobile robots via swarm intelligence approach. *Robotics and Autonomous Systems*, 55(7):572–588, 2007.
- [166] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

- [167] Gerhard Weiss. *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT press, 1999.
- [168] Jacques Ferber. *Multi-agent systems: an introduction to distributed artificial intelligence*, volume 1. Addison-Wesley Reading, 1999.
- [169] Eric Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7280–7287, 2002.
- [170] Andrei Borshchev and Alexei Filippov. From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools. In *Proceedings of the 22nd International Conference of the system dynamics society*, volume 22, 2004.
- [171] Andrew T Crooks and Sarah Wise. Gis and agent-based models for humanitarian assistance. *Computers, Environment and Urban Systems*, 41: 100–111, 2013.
- [172] Nina Schwarz and Andreas Ernst. Agent-based modeling of the diffusion of environmental innovations - an empirical approach. *Technological forecasting and social change*, 76(4):497–511, 2009.
- [173] Guangyu Zou, Alvaro Gil, and Marina Tharayil. An agent-based model for crowdsourcing systems. In *Proceedings of the 2014 Winter Simulation Conference*, pages 407–418. IEEE Press, 2014.
- [174] Zhiqiang Yu, Dejing Zhang, and Dong Yang. Participant selection for offline event marketing leveraging location-based social networks. *IEEE*

- Transactions on Systems, Man, and Cybernetics: Systems*, 45(6):853 – 864, 2015.
- [175] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1): 129–142, 2015.
- [176] Han Yu, Chunyan Miao, Zhiqi Shen, and Cyril Leung. Quality and budget aware task allocation for spatial crowdsourcing. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems*, pages 1689–1690. International Foundation for Autonomous Agents and Multiagent Systems, 2015.
- [177] Yu Li, Man Lung Yiu, and Wenjian Xu. Oriented online route recommendation for spatial crowdsourcing task workers. In *Advances in Spatial and Temporal Databases*, pages 137–156. Springer, 2015.
- [178] John Joseph Horton and Lydia B Chilton. The labor economics of paid crowdsourcing. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, pages 209–218. ACM, 2010.
- [179] Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. *Network flows: theory, algorithms, and applications*. Prentice-Hall, Inc., 1993.
- [180] Roy Jonker and Anton Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340, 1987.
- [181] Andrew Makhorin. Glpk (gnu linear programming kit), 2008.

- [182] Werner Dinkelbach. On nonlinear fractional programming. *Management Science*, 13(7):492–498, 1967.
- [183] Abraham Charnes and William W Cooper. Programming with linear fractional functionals. *Naval Research logistics quarterly*, 9(3-4):181–186, 1962.
- [184] Nimrod Megiddo. Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4(4):414–424, 1979.
- [185] David G Luenberger. *Introduction to linear and nonlinear programming*, volume 28. Addison-Wesley Reading, MA, 1973.
- [186] Tomasz Radzik. Newton’s method for fractional combinatorial optimization. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 659–669. IEEE, 1992.
- [187] Daniel Schall. *Service-oriented crowdsourcing: architecture, protocols and algorithms*. Springer Science & Business Media, 2012.
- [188] Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L Chevrier, Kristin A Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.
- [189] William E Hart, Carl Laird, Jean-Paul Watson, and David L Woodruff. *Pyomo—optimization modeling in python*, volume 67. Springer Science & Business Media, 2012.

- [190] Jean-Yves Audibert, Sébastien Bubeck, and Gábor Lugosi. Regret in online combinatorial optimization. *Mathematics of Operations Research*, 39(1):31–45, 2013.
- [191] Bernhard Meindl and Matthias Templ. Analysis of commercial and free and open source solvers for linear optimization problems. *Eurostat and Statistics Netherlands within the project ESSnet on common tools and harmonised methodology for SDC in the ESS*, 2012.
- [192] Erik B Bajalinov. *Linear-fractional programming: theory, methods, applications and software*, volume 84. Springer, 2003.
- [193] Steven L Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.
- [194] Lijing Qin, Shouyuan Chen, and Xiaoyan Zhu. Contextual combinatorial bandit and its application on diversified online recommendation. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 461–469, 2014.
- [195] Ryoma Kawajiri, Masamichi Shimosaka, and Hisashi Kahima. Steered crowdsensing: incentive design towards quality-oriented place-centric crowdsensing. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 691–701. ACM, 2014.
- [196] Eran Toch, Justin Cranshaw, Paul Hanks Drielsma, Janice Y Tsai, Patrick Gage Kelley, James Springfield, Lorrie Cranor, Jason Hong, and Norman Sadeh. Empirical models of privacy in location sharing. In *Proceedings of the 12th ACM International Conference on Ubiquitous computing*, pages 129–138. ACM, 2010.

- [197] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. In *Advances in Neural Information Processing Systems*, pages 2249–2257, 2011.
- [198] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, pages 285–294, 1933.
- [199] J Kennington and Z Wang. A shortest augmenting path algorithm for the semi-assignment problem. *Operations Research*, 40(1):178–187, 1992.
- [200] A Volgenant. Linear and semi-assignment problems: a core oriented approach. *Computers & operations research*, 23(10):917–932, 1996.
- [201] Dingqi Yang, Daqing Zhang, Vincent. W. Zheng, and Zhiyong Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1): 129–142, 2015. ISSN 2168-2216.
- [202] Stephanie Rosenthal, Anind K Dey, and Manuela Veloso. Using decision-theoretic experience sampling to build personalized mobile phone interruption models. In *Pervasive Computing*, pages 170–187. Springer, 2011.
- [203] Qiang Liu, Alex T Ihler, and Mark Steyvers. Scoring workers in crowdsourcing: How many control questions are enough? In *Advances in Neural Information Processing Systems*, pages 1914–1922, 2013.
- [204] Rensis Likert. A technique for the measurement of attitudes. *Archives of psychology*, 1932.

- [205] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. Dbpedia-a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, 2009.