Analysis of Online Optimization Problems in Navigation and Search on Networks

by

Davood Shiri

A Dissertation Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Doctor of Philosophy

in

Industrial Engineering and Operations Management

KOÇ
UNIVERSITY

January, 2019

Analysis of Online Optimization Problems in Navigation and Search on Networks

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a doctoral dissertation by

Davood Shiri

and have found that it is complete and satisfactory in all respects,

and that any and all revisions required by the final

examining committee have been made.

Committee Members:

Assoc. Prof. Sibel Salman

Prof. Emre Alper Yıldırım

Prof. Bahar Yetiş Kara

Asst. Prof. Dilek Günneç Danış

Asst. Prof. Eda Yücel

Date: ______________________________

ABSTRACT

In this thesis, we study online optimization problems that are related to navigation and search on networks. In online problems information is revealed incrementally, and decisions must be made before all information is available. We design and analyze strategies for several online problems with applications in disaster response, search-and-rescue, security, and defense. We prove worst-case competitive ratios to analyze the performance of the proposed strategies. We first study the online k-Canadian Traveler Problem (k-CTP) on O-D edge-disjoint graphs. An optimal randomized strategy was given in the literature. We prove that the given strategy cannot be implemented in some cases and modify it such that it is optimal and can be implemented in all cases. We consider the online multi-agent k-CTP. We derive improved lower bounds on the competitive ratio of deterministic strategies for the cases with limited and complete communication. We introduce two deterministic strategies and show that one of them is optimal in both cases with complete and limited communication on O-D edge-disjoint graphs. We provide lower bounds on the competitive ratio of randomized strategies for the cases without communication, with limited communication and with complete communication. We introduce a randomized online strategy which is optimal for both cases with limited and complete communication on O-D edge-disjoint graphs. We also consider the online Minimum Latency Problem with edge uncertainty. We present an optimal deterministic strategy. Moreover, we present a lower bound on the expected competitive ratio of randomized strategies. Finally, we investigate the online Discrete Search Problem with traveling and search costs on undirected graphs. We propose tight competitiveness lower bounds together with optimal deterministic and randomized strategies.

ÖZETÇE

Bu tezde, ağ yapıları üstünde navigasyon ve arama ile ilgili çeşitli çevrimiçi eniyileme problemleri üzerinde çalışılmıştır. Çevrimiçi problemlerde bilgiler adım adım açıklanır ve tüm bilgiler mevcut olmadan önce kararlar alınmalıdır. Tez kapsamında, afete müdahale, arama kurtarma, güvenlik ve savunma alanlarında uygulamaları olan birkaç çevrimiçi eniyileme problemi için eniyi stratejiler tasarlanıp bunların performansları teorik olarak analiz edilmiştir. Önerilen stratejilerin performanslarını analiz etmek için en kötü durumda, bilginin baştan elde olduğu (çevrimdışı) durumdaki en iyi çözüme göre, rekabetçi oranlar belirlenmiştir. İlk olarak, çevrimiçi k-Kanadalı Gezgin Problemi (k-KGP), ayrıtları kesişmeyen yollara sahip çizgeler üzerinde incelenmiştir. Daha önce literatürde bu problem için bir eniyi rassal strateji verilmiştir. Bu çalışmada, bazı durumlarda bu stratejinin uygulanamaz olduğu gösterilerek, strateji her durumda uygulanabilir ve eniyi olacak şekilde değiştirilmiştir. Daha sonra çevrimiçi çok katılımcılı k-KGP ele alınmıştır. Bu problemin sınırlı ve sınırsız iletişimin olduğu iki durumuna bakılarak, literatürde verilen, deterministik stratejilerin rekabetçi oranına alt sınırı iyileştirilmiştir. Aynı iki durum, ayrıtları kesişmeyen yollara sahip çizgeler üzerinde incelenerek, iki deterministik strateji geliştirilmiştir. Bunlardan bir tanesinin eniyi olduğu ispatlanmıştır.

Problemin iletişimin olmadığı, sınırlı ve sınırsız iletişimin olduğu üç durumuna bakılarak, rassal stratejilerin rekabetçi oranına alt sınırlar geliştirilmiştir. Ayrıca iletişimli durumlar için rassal bir strateji geliştirilerek, bunun ayrıtları kesişmeyen yollara sahip çizgeler üzerinde eniyi olduğu ispatlanmıştır. Ayrıt belirsizliği olan çevrimiçi Minimum Gecikme Problemi de ele alınan bir başka problemdir. Bu problem için bir eniyi deterministik strateji geliştirilmiştir. Ayrıca, rassal stratejilerin beklenen rekabetçi oranına bir alt sınır bulunmuştur. Son olarak, çevrimiçi Ayrık

Arama Problemi, yönlendirilmemiş çizgelerdeki seyahat ve arama maliyetleriyle incelenmiştir. Eniyi deterministik ve rassal stratejiler bulunmuştur.

ACKNOWLEDGMENTS

First, I would like to thank my thesis advisor Professor Sibel Salman, for her kind understanding, support, guidance, and encouragement through my PhD. She taught me innumerable lessons in research as well as in life. I could not wish for a better advisor. My sincere thanks must also go to the members of my thesis advisory committee: Professors Emre Alper Yıldırım and Bahar Yetiş Kara. They generously gave their time to offer me valuable comments toward improving my work. I also need to thank Professors Dilek Günneç Danış and Eda Yücel for being in my thesis defense committee and for their feedback on my thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

- CTP : Canadian Traveler Problem
- CTOP : Canadian Tour Operator Problem
- M-RBS : Modified-Randomized Backtrack Strategy
- PLS : Path Labeling Strategy
- Modified-PLS : Modified-Path Labeling Strategy
- RBS : Randomized Backtrack Strategy
- TSP : Traveling Salesman Problem
- sTSP : Steiner Traveling Salesman Problem
- CSP : Covering Salesman Problem
- MLP : Minimum Latency Problem
- OMLP : online Minimum Latency Problem with edge uncertainty
- ODSP : online Discrete Search Problem with traveling and search costs on undirected graphs
- OSP : Optimal Search Problem

Chapter 1

INTRODUCTION

Conventionally, optimization techniques within the area of Operations Research have mainly focused on deterministic planning for various problems. However, deterministic planning may lead to poor solutions when actuality differs from expectation. For example, deterministic routing plans may lead to infeasible solutions after a natural disaster due to unexpected failure of road network links. Furthermore, information about real-world problems is rarely completely known a priori. For instance, travel times associated with road network links are generally revealed over time in an *online* manner without advance knowledge. Waiting for all necessary information is costly, if not impossible, for many applications. Hence, for such problems, it is essential to develop approaches that make decisions online, as information is revealed.

Online optimization is a field of optimization theory which finds place in Operations Research, Computer Science and Economics. It deals with optimization problems with incomplete information on their inputs. Problems in which incomplete information is revealed online while a solution strategy is implemented are called *online problems*. Solution strategies for online problems are called *online strategies*.

This thesis focuses on designing and analyzing online strategies for various online optimization problems within the field of Network Optimization, namely the online k-Canadian Traveler Problem, the online multi-agent k-Canadian Traveler Problem, the online Minimum Latency Problem with edge uncertainty, and the online Discrete Search Problem on undirected graphs.

1.1 Online and offline strategies

A solution strategy to an online problem is called an online strategy. Online strategies process their inputs piece-by-piece, without having the entire input available from the beginning. Online strategies are divided into two categories as *deterministic* and *randomized*. In a deterministic online strategy, actions of the decision maker do not depend on probabilistic outcomes. That is, given a particular input, a deterministic online strategy will always produce the same output. In a randomized strategy, actions of the decision maker are taken according to some probability distribution in the sense that given a particular input, a randomized online strategy may produce different outputs.

The key concept in analyzing an online strategy is to compare a solution produced by the online strategy with the best possible solution under complete information, which is called the *offline optimum* solution. An *offline strategy* is to solve the same problem as an online strategy, except that all information about the problem inputs is revealed to an offline strategy from the beginning. An *optimal offline strategy* is the optimal strategy in presence of complete input information which produces the offline optimum solution.

1.2 Competitive analysis

The traditional approach for analyzing online strategies falls within the framework of *distributional (or average-case)* complexity, whereby one hypothesizes a distribution on input and studies the expected total cost. During the past decades the interest in this subject has been renewed largely as a result of the approach of *competitive analysis*, where the quality of an online strategy is measured by comparing its performance to that of an optimal offline strategy. Hence, competitive analysis falls within the framework of the *worst-case* complexity.

We note the reader that a more general study of online strategies is a much more ambitious topic, and competitive analysis is only one aspect of decision making in the absence of complete input information. Any conceptual model has limitations as well as benefits. The disadvantage of the traditional distributional complexity is that the prior distribution is rarely known precisely and often the distributional assumptions are unrealistically crude to allow for mathematical tractability. On the other hand, competitive analysis has the disadvantage of being too pessimistic, assuming a *malicious adversary* that chooses the worst input by which to measure a strategy's performance. (This is the limitation in any worst-case analysis). We should not expect competitive analysis to be uniformly worthwhile over all possible application areas. However, it is becoming apparent that in some application areas such as network routing it has practical relevance [21]. The focus of this thesis is on competitive analysis of online strategies for various online optimization problems defined on networks.

1.2.1 Notion of the competitive ratio for deterministic strategies

For minimization problems, a deterministic online strategy is called c-competitive ($c \geq 1$), if c is the smallest number such that for any instance of the problem, the cost of the solution given by the deterministic online strategy is at most c times the cost of an optimal offline solution for the instance:

$$Cost_{online}(I) \leq c(Cost_{offline}(I)),\ \forall\ instances\ I.$$

Equivalently, the competitive ratio equals to

$$c = \sup_I \frac{Cost_{online}(I)}{Cost_{offline}(I)}.$$

A deterministic online strategy is said to be *optimal* if no other deterministic strategy has a strictly smaller competitive ratio [21].

Similarly, for maximization problems, a deterministic online strategy is called c-competitive ($c \leq 1$), if c is the largest number such that for any instance of the problem, the cost of the solution given by the deterministic online strategy is at least

c times the cost of an optimal offline solution for the instance:

$$Cost_{online}(I) \geq c(Cost_{offline}(I)),\ \forall\ instances\ I.$$

Equivalently, the competitive ratio is equal to

$$c = \inf_{I} \frac{Cost_{online}(I)}{Cost_{offline}(I)}.$$

A deterministic online strategy is said to be optimal if no other deterministic strategy has a strictly larger competitive ratio.

1.2.2 Notion of the expected competitive ratio for randomized strategies

For minimization problems, a randomized online strategy is called c-competitive ($c \geq 1$), if c is the smallest number such that for any instance of the problem, the expected cost of the solution given by the randomized online strategy is at most c times the cost of an optimal offline solution for the instance:

$$E[Cost_{online}(I)] \leq c(Cost_{offline}(I)),\ \forall\ instances\ I.$$

Equivalently, the expected competitive ratio equals to

$$c = \sup_{I} \frac{E[Cost_{online}(I)]}{Cost_{offline}(I)}.$$

A randomized online strategy is said to be *optimal* if no other randomized strategy has a strictly smaller expected competitive ratio.

Similarly, for maximization problems, a randomized online strategy is called c-competitive ($c \leq 1$), if c is the largest number such that for any instance of the problem, the expected cost of the solution given by the randomized online strategy is at least c times the cost of an optimal offline solution for the instance:

$$E[Cost_{online}(I)] \geq c(Cost_{offline}(I)),\ \forall\ instances\ I.$$

Equivalently, the expected competitive ratio is equal to

$$c = \inf_{I} \frac{E[Cost_{online}(I)]}{Cost_{offline}(I)}.$$

A randomized online strategy is said to be optimal if no other randomized strategy has a strictly larger competitive ratio.

1.3 Problem definitions

In this thesis we address four different online problems, as defined in the following.

1.3.1 Online k-Canadian Traveler Problem

The online Canadian Traveler Problem (CTP) is a navigation problem under incomplete information. A traveling agent receives an undirected graph $G = (V, E)$ with a given source node O and a destination node D, together with non-negative edge costs as input. The agent is located at O initially. There are some blocked edges in the graph, but these edges are not known to the agent. The agent discovers the status of an edge when he reaches an end-node of the edge. The objective is to provide an online strategy such that the agent finds a feasible path, i.e. one without blocked edges from O to D with minimum total cost of the edges taken by the agent. When an upper bound k ($k \geq 1$) on the number of blocked edges is given as input, the problem is called the k-CTP.

1.3.2 Online multi-agent k-Canadian Traveler Problem

The multi-agent k-CTP is an online optimization problem that generalizes the k-CTP by the existence of multiple agents. In the multi-agent k-CTP, there are L agents in the graph who are initially located at O. The objective of the agents is to provide an online strategy such that at least one of them finds a feasible path, from O to D with minimum total cost of the edges taken by the agent that finds a feasible path first. Two versions of the multi-agent k-CTP have been introduced in the literature, with complete and limited communication. When the communication is limited, some agents can both send and receive information and some of the agents can only receive information. The agents who are able to both send and receive information are called RS-type agents and the agents who are only able to receive information are called R-type agents. In presence of complete communication, all of the agents can send and receive information, i.e. all of the agents are RS-type.

1.3.3 Online minimum latency problem with edge uncertainty

In the Minimum Latency Problem (MLP), an undirected simple connected graph $G = (V, E)$ is given to an agent, where $V = \{v_0, v_1, v_2, ..., v_n\}$ and $v_0 \in V$ is a root node. Non-negative edge distances are also given. The agent should start from v_0 and complete a tour visiting all the nodes. The latency of v_i is denoted by l_i, which represents the distance traveled before first visiting v_i. Naturally, l_0 is zero. The objective of the agent is to find a tour on G, starting from v_0, that minimizes $\sum_{i=1}^{n} l(i)$. In the *online Minimum Latency Problem with edge uncertainty* (OMLP), k edges of G are blocked, and the agent learns that an edge $e \in E$ is blocked, only if she reaches at one of the end-nodes of e. It is assumed that the graph remains connected if the blocked edges are removed from it. The objective of the problem is to provide an online strategy such that the agent finds a feasible tour, i.e. one without blocked edges, starting from v_0 which minimizes $\sum_{i=1}^{n} l(i)$.

1.3.4 Online discrete search problem with traveling and search costs on undirected graphs

In the *online Discrete Search Problem on undirected graphs* (ODSP), an undirected connected graph $G = (V, E)$ is given, where $V = \{v_0, v_1, v_2, ..., v_n\}$, each node v_i ($i \in \{1, 2, ..., n\}$) of the graph is associated with a given non-negative search cost s_i. A non-negative edge cost d_e is also given for each edge $e \in E$. A static hider is at one of the nodes v_{i^*} ($i^* \in \{1, 2, ..., n\}$) which is not known to the searcher. The hider is not found unless the searcher arrives at v_{i^*} and incurs the search cost of v_{i^*}. Starting from

v_0, the searcher wants to devise an online strategy to locate the hider with minimum total cost.

1.4 Literature review

In this section, we review related literature on the problems defined in the previous section.

1.4.1 Online optimization and competitive analysis

Online strategies have been implicitly and explicitly studied in the context of scheduling, optimization, data structures, and other computational topics. The roots of competitive analysis can be found in classical combinatorial optimization theory and in the analysis of data structures [21]. The first systematic study of online strategies is presented by Sleator and Tarjan [55], who suggest comparing an online strategy with an optimal offline strategy. This type of analysis was later called competitive analysis in [36]. For more details and information on online optimization and competitive analysis, see the survey paper of Albers [3] and the books of Borodin and El-Yaniv [21], Fiat and Woeginger [29], and Komm [38].

1.4.2 Online k-Canadian Traveler Problem

The CTP is defined first in [48]. Papadimitriou and Yannakakis [48] proved that devising an online strategy with a bounded competitive ratio is PSPACE-complete for the CTP.

Bar-Noy and Schieber [14] considered several variations of the CTP. They introduced the k-CTP, where an upper bound k on the number of blocked edges is given as input. They showed that for arbitrary k, the problem of designing an online strategy that guarantees the minimum travel cost is PSPACE-complete. They also considered the Recoverable k-CTP, in which each blocked edge is associated with a recovery cost to re-open. They considered the worst-case criterion for the Recoverable k-CTP and provided a strategy with minimum worst-case cost under the assumption that the recovery costs of the edges is less than or equal to their original costs.

Nikolova and Karger [46] studied a stochastic version of the CTP, where the edges in the input graph have uncertain costs which are associated with known probability distributions. The objective of the problem is to find an optimal policy that minimizes total expected cost. They applied a mix of techniques from algorithm analysis and the theory of Markov Decision Processes to provide efficient exact strategies for directed acyclic graphs and undirected node-disjoint O-D paths. Fried et al. [31] showed that the stochastic CTP is PSPACE-complete. They initially proved PSPACE-hardness for the dependent version of the stochastic CTP, and extended their proof to the independent case. They also examined the complexity of the more general remote-sensing CTP, and showed that it is NP-hard even for disjoint-path graphs. Aksakalli et al. [2] investigated the stochastic CTP. They introduced an optimal strategy for the problem based on a Markov decision process formulation, which is a new improvement on AO* search that takes advantage of the special problem structure in CTP. Sahin and Aksakalli [49] studied the stochastic CTP, where the agent is given prior blockage probabilities associated with each edge, and the objective is to devise a strategy that minimizes the expected traversal cost between two given nodes. They compared penalty-based and rollout-based algorithmic frameworks via computational experiments involving Delaunay and grid graphs using one specific penalty-based strategy and four rollout-based strategies. Their results indicated that the penalty-based strategy executes several orders of magnitude faster than rollout-based ones while also providing better policies.

Westphal [64] considered the k-CTP from the competitive ratio perspective. By analyzing an instance of graphs that consist of only node-disjoint O-D paths, he showed the lower bounds of $2k+1$ and $k+1$ on the competitive ratio of deterministic and randomized strategies, respectively. He also presented an optimal deterministic strategy which is called the *backtrack* strategy and showed that its competitive ratio matches the lower bound of $2k+1$. Xu et al. [66] also considered the k-CTP and presented two online strategies, the *greedy* and the *comparison* strategy and proved the competitive ratios of $2^{k+1} - 1$ and $2k + 1$, respectively for these strategies.

Su and Xu [58] studied online strategies for the recoverable k-CTP. They presented the waiting strategy and the greedy strategy. They proved tight competitive ratios of $1+\alpha$ and $(1+\alpha^k)$, respectively for the two strategies, where α denotes the maximum ratio of recovery time to normal travel time among all edges in the graph. Su et al. [59] presented an optimal online strategy, i.e. a comparison strategy for recoverable k-CTP and proved its competitive ratio in special graph. Liao and Huang [42] considered a generalization of the k-CTP, where each edge of the input graph is associated with two different travel costs. They considered graphs with only node-disjoint O-D paths to derive lower bounds on the competitive ratio of deterministic and randomized strategies for this problem. They also proved that their proposed lower bound on the competitive ratio of deterministic strategies is tight by introducing a deterministic strategy whose competitive ratio meets their proposed lower bound.

Bender and Westphal [4] presented a randomized online strategy for the k-CTP which meets the lower bound of $k + 1$ in special cases. This randomized strategy can be regarded as the randomized version of the backtrack strategy. Demaine et al. [27] suggested a randomized online strategy for the k-CTP which achieves the competitive ratio of $(1 + \frac{\sqrt{2}}{2})k + 1$ in pseudo-polynomial time.

Related problems to the CTP

There are some recent works on similar online problems in the literature. Liao and Huang [41], investigated a variation of the Traveling Salesman Problem that involves finding a shortest tour, under the same uncertainty as that of the CTP. They called this online routing problem as Covering Canadian Traveler Problem (CCTP). They studied the problem from the competitive analysis perspective and presented an efficient touring strategy within an $o\sqrt{k}$ - competitive ratio, where the number of blockages is at most k. They also demonstrated the tightness of their competitive analysis.

Zhang et al. [71] considered the online Steiner Traveling Salesman Problem (sTSP), in which the traveler needs to visit multiple destination vertices. The objective of their problem is to find a minimum weight tour that traverses every destination vertex at least once. As in the k-CTP, the traveler could encounter at most k blocked edges. They provided a lower bound on the competitive ratio and presented an optimal strategy for the problem. While the optimal strategy does not have polynomial running time, they presented another online polynomial time near-optimal strategy for the problem. Zhang et al. [72] formulated the sTSP with online advanced edge blockages to address an application in package delivery, where the driver (salesman) receives road blockage messages when he is at a certain distance to the respective blocked edges. Such road blockages are referred to as advanced information. With these online advanced road blockages, the driver wishes to deliver all the packages to their respective customers and returns back to the service depot through a shortest route. During the entire delivery process, there will be at most k road blockages, and they are non-recoverable. Zhang et al. [72] proved lower bounds on the competitive ratio of deterministic strategies for this problem. They present a polynomial time online strategy with a competitive ratio very close to this lower bound. Similar results for a variation, in which the driver does not need to return to the service depot, are also provided.

Buttner and Krumke [24] studied the Canadian Tour Operator Problem (CTOP) which is an online variant of Prize-collecting Traveling Salesman Problem with online blocked edges related to graph exploration. The goal consists of minimizing the sum of the travel costs and the refunds. They analyzed the problem on a simple (weighted) path and prove tight bounds on the competitiveness of deterministic strategies. They also considered the effect of resource augmentation, where the online strategy either pays a discounted cost for traversing edges or for the penalties. Zhang and Xu [73] proposed the online Covering Salesman Problem (CSP) in which the salesman will encounter at most k blocked edges during the traversal. They suggested a lower bound on the competitive ratio of deterministic strategies and introduced a deterministic strategy which is near-optimal in a special case.

1.4.3 Online multi-agent k-Canadian Traveler Problem

The k-CTP with multiple agents is first considered by Zhang et al. [74]. They analyzed the problem in two scenarios, with complete and limited communication. They proposed lower bounds of $2\lfloor \frac{k-1}{L_1} \rfloor + 1$ and $2\lfloor \frac{k}{L} \rfloor + 1$ on the competitive ratio of the deterministic strategies for the cases with limited and complete communication, respectively. Note that in the proposed lower bounds L denotes the total number of agents and L_1 denotes the number of RS-type agents. They also proposed an optimal deterministic strategy when there are two agents in the graph.

Xu and Zhang [67] focused on a real-time rescue routing problem from a source node to an emergency spot in presence of online blocked edges. They analyzed the problem with the objective to make all the rescuers arrive at the emergency spot with minimum total cost. They studied the problem in two scenarios: (1) without communication and (2) with complete communication. They investigated both of the scenarios on the grid networks and general networks, respectively. They showed that the consideration of both the grid network and the rescuers' communication can significantly improve the rescue efficiency. Bnaya et al. [20] considered a stochastic version of the CTP, where some of the edges are blocked with a known probability. They generalized CTP to a repeated task version where a number of agents need to travel to the same goal, minimizing their combined expected travel cost. They provided optimal strategies for the special case of disjoint path graphs.

1.4.4 Online minimum latency problem with edge uncertainty

The MLP is a well-studied problem in combinatorial optimization. This problem is also known as the *deliveryman problem* [1] or the *traveling repairman problem* [30]. The MLP is an NP-hard problem [50] and it is APX-hard, implying the non-existence of a polynomial-time approximation scheme (PTAS) unless P=NP [54]. Several exact algorithms have been proposed for the MLP (see [44], [65], [45], [7] and [6]). Approximation algorithms for the MLP have been extensively investigated (see [19], [32], [9] and [25]), and the best approximation ratio achieved to date is 3.59

which is presented in [25].

The OMLP has been recently studied in [70], where a lower bound of $2k + 1$ on the competitive ratio of deterministic strategies is provided. Two heuristic online strategies are also suggested in [70] which are called the *GoodTreeTraversal* strategy and the *Detour* strategy. The *GoodTreeTraversal* strategy produces near optimal solutions when the number of blockages is large enough. The *Detour* strategy has no theoretical guarantee on its performance and runs in polynomial time.

Related problems to the OMLP

The MLP is closely related to the well-known Traveling Salesman Problem (TSP), of which the input is the same but the objective is to minimize the total length of the tour visiting all nodes. In the online traveling salesman problem requests for visits to cities (points in a metric space) arrive online while the salesman is traveling. The salesman moves at no more than unit speed and starts and ends his work at a designated origin. The objective is to find a routing for the salesman which finishes as early as possible. Ausiello et al. [13] studied the problem of efficiently serving a sequence of requests presented in an online fashion located at points of a metric space. They considered two versions of the problem. In the first one the server is not required to return to the departure point after all presented requests have been served. In the second one returning to the departure point is required. They provided competitiveness lower bounds and efficient deterministic online strategies for these cases. Blom et al. [18] considered the online TSP when restricted to the non-negative part of the real line. They showed that a very natural strategy is $\frac{3}{2}$-competitive which matches their suggested lower bound. The main contribution of their paper is the presentation of a *fair adversary*, as an alternative to the omnipotent adversary used in competitive analysis for online routing problems. They presented an efficient deterministic online strategy against a fair adversary.

Ausiello et al. [12] investigated an online variant of the Quota TSP which is a generalization of the TSP. In the Quota TSP, the objective is to reach a given quota of sales minimizing the amount of time. Ausiello et al. [12] addressed the online version of the problem, where requests are given over time. They presented strategies for various metric spaces, and analyze their performance in the usual framework of competitive analysis. Auseillo et al. [10] considered two online versions of the asymmetric traveling salesman problem with triangle inequality. For the homing version, in which the salesman is required to return in the city where it started from, they presented a tight lower bound on the competitive ratio of deterministic strategies together with an optimal deterministic online strategy. For the nomadic version, the online analogue of the shortest asymmetric Hamiltonian path problem, they showed that the competitive ratio of any online strategy depends on the amount of asymmetry of the space in which the salesman moves. Ausiello et al. [11] studied the online version of the Prize-Collecting TSP, a generalization of the TSP, where each city (node) has a given weight and penalty, and the goal is to collect a given quota of the weights of the cities while minimizing the length of the tour plus the penalties of the cities not in the tour. In the online version, cities are disclosed over time. They derived a lower bound of 2 on the competitive ratio of deterministic strategies and introduced a $\frac{7}{3}$-competitive deterministic online strategy.

Jaillet and Lu [33] analyzed the online TSP with service flexibility. They assumed that there is a penalty for not serving a request. Requests for visit of points in the metric space are revealed over time to a server, initially at a given origin, who must decide in an online fashion which requests to serve to minimize the time to serve all accepted requests plus the sum of the penalties associated with the rejected requests. They investigated the problem on non-negative real line, real line, and general metric space and proposed optimal deterministic online strategies for special cases. Jaillet and Lu [34] considered online versions of the TSP on metric spaces for which requests to visit points are not mandatory. Associated with each request is a penalty (if rejected). Requests are revealed over time (at their release dates) to a server who must decide which requests to accept and serve in order to minimize a linear combination of the time to serve all accepted requests and the total penalties of all rejected requests. In the basic online version of the problem, a request can be accepted any time after its release date. In the real-time online version, a request must be accepted or rejected at the time of its release date. They also provided optimal deterministic online strategies for both of the versions on general metric space. For the real-time version, they also introduced an optimal deterministic online strategy for the special case of non-negative real line.

Wen et al. [62] modelled the customers waiting psychology and service preparation time into the online TSP with the objective to serve as many requests as possible. In their paper, each request has a disclosure time before accepting service at its release time, and a deadline, which is no bigger than its release time plus the travel time from origin to its position. They presented lower bounds for the competitive ratios, online strategies, and quantify the influence of advanced information on competitive ratios. Wen et al. [63] analyzed a version of the online TSP with deadlines and service flexibility, where the salesman can choose whether to serve or not when a new request arrives. By rejecting the request or missing its deadline, penalties will be generated. The goal is to minimize servers costs (travel makespan plus the penalties of missed requests). They showed that no deterministic or randomized online strategies can achieve constant competitive ratio for the problem on general metric space.

1.4.5 Online discrete search problem with traveling and search costs on undirected graphs

Work on search theory began in the U.S. Navy's Antisubmarine Warfare Operations Research Group (WORG) in 1942 in response to the German submarine threat in the Atlantic [57]. Bernard Koopman joined WORG in 1943 and was the first person to provide the basic probabilistic foundation for search problems. Koopman [39] studied the optimal allocation of a fixed amount of search effort to detect a static hider. He defined the elements of the basic problem of optimal search: a prior distribution on hider location, a function relating search effort and detection probability, a constrained amount of search effort, and the optimization criterion of maximizing the probability of detection subject to a constraint on the amount of effort. This problem is now called the *Optimal Search Problem* (OSP). Significant progress was made on the OSP with a static hider between 1946 and 1965, which is summarized in [56]. An excellent review on the theoretical achievements on search theory related to both versions with static and moving hider is presented in [57].

Search problems on graphs

Search problems on graphs were introduced by Rufus Isaacs in the final chapter of his classic 1965 book, *Differential Games* [4]. An extensive amount of research has been conducted on search problems in which a static hider selects a point on a given graph, and a moving searcher targets to find her as quickly as possible by traversing the edges of the graph. The searcher starts from a point of the graph which is either decided by the searcher or fixed a priori. The literature on this type of search problems (those defined on graphs) is divided into two main streams. The first stream assumes that the hider is permitted to position herself at any point of the graph, and the searcher locates the hider when he arrives at the point in which the hider is positioned, i.e. the first stream corresponds to continuous search problems. For problems belonging to the first stream, see [4], [8], [26] and [35].

The second stream assumes that the hider hides at a node of the graph and before the hider can be found, the searcher conducts a search at the node in which the hider is positioned, i.e. the second stream corresponds to discrete search problems. In this case, operating a search at a node incurs a cost to the searcher, which can vary from node to node. Hence, the searcher not only bears traveling costs but also search costs. When the searcher reaches a node, he does not have to search it but can pass through it and return at a later time to conduct a search, if he has not found the hider in the meantime. There are two decision rules to tackle search problems of the second stream in the literature: 1) *Bayes*, in which the searcher minimizes the expected cost with respect to a prior distribution, and 2) *minimax*, in which the problem is modeled as a zero-sum game between the searcher and the hider. In the Bayes approach, the probability of overlooking the hider in case the searcher probes the same node in which the hider is positioned is also taken into account.

Loessner and Wegener [43] studied a version of search problems of the second stream where the starting node is decided by the searcher from Bayes perspective. They obtained necessary and sufficient conditions for the existence of optimal strategies. Wegener [61] analyzed the complexity of the same problem as the one considered in [43] and proved that the problem is NP-hard. Kikuta [37] studied a version of the search problems belonging to the second stream on finite cyclic graphs from the minimax point of view, where the starting node of the searcher is given. He modeled the problem as a two-person zero-sum game and solved it for a special case. He also proposed properties of optimal strategies for both the searcher and the hider. Baston and Kikuta [15] investigated a version of search problems of the second stream from the minimax perspective, where the starting node is decided by the searcher. They modeled the problem as a two-person zero-sum game and provided an upper bound for the value of the game. In addition, they proved a lower bound on the value of the game when the edge costs are uniform. Baston and Kikuta [15] also provided results on star and line graphs. In another article, Baston and Kikuta [16] analyzed a version of search problems of the second stream from the minimax perspective on directed, not necessarily strongly connected, graphs, where the starting node is decided by the searcher. For more on search theory and related problems, the reader is referred to [5], where zero-sum search games between the searcher and the hider under different scenarios are considered.

1.5 Thesis contributions

Below we summarize our contributions chapter by chapter.

1.5.1 Chapter 2 (Online k-Canadian Traveler Problem)

In this chapter, we reconsider the randomized online strategy that is presented for the k-CTP on graphs where all O-D paths are node-disjoint in [4]. This strategy can be regarded as the randomized version of the backtrack strategy according to Bender and Westphal. Hence we call it the *randomized backtrack strategy* (RBS).

We show that a particular property regarding the costs of the O-D paths must hold to implement the RBS. We formally specify this property. Next we show that this property does not necessarily hold when the costs of the O-D paths in the input graph are arbitrary and $k > 2$. That is we prove that the RBS is not applicable on graphs that consist of only node-disjoint O-D paths, where the costs of the O-D paths are arbitrary and $k > 2$. Moreover, we modify the RBS and introduce an optimal strategy which is applicable on graphs where all O-D paths are node-disjoint and the costs of the O-D paths are arbitrary.

These results have been published in [53].

1.5.2 Chapter 3 (Online multi-agent k-Canadian Traveler Problem)

Analysis of deterministic online strategies

In the first section of Chapter 3, we study deterministic online strategies for the online multi-agent k-CTP. We focus on the case where communication among the agents is limited. We define three levels of agents' intelligence. We introduce two simple deterministic online strategies and use them when the agents benefit from higher levels of intelligence. By this way, we provide updated lower bound on the competitive ratio of deterministic online strategies for the case with limited communication on general graphs. We also show that one of our strategies is optimal in both cases with complete and limited communication in the special case of *edge-disjoint* graphs. Note that in edge-disjoint graphs, there exists no path with common edges with any other path. We need to mention that analyzing edge-disjoint graphs is a standard restriction in the context of k-CTP and its variants. Finally, we argue that increasing the number of R-type agents can improve the competitive ratio of deterministic strategies for the online multi-agent k-CTP with limited communication. These results have been published in [51].

Analysis of randomized online strategies

In the second section of Chapter 3, we focus on randomized online strategies for the multi-agent k-CTP. We analyze the problem in three cases: 1) without communication, 2) with limited communication and 3) with complete communication. We derive lower bounds on the competitive ratio of the randomized strategies for all of these cases. For the case without communication, we introduce a simple randomized strategy and prove its competitive ratio on a special case. By this way, we prove that increasing the number of agents can improve the competitive ratio of the randomized strategies for the multi-agent k-CTP.

For the cases with limited and complete communication, we introduce an optimal randomized strategy for both cases on O-D edge-disjoint graphs. Here we note that most optimal strategies in the literature are confined to O-D edge-disjoint graphs. Because our optimal strategy achieves a better expected competitive ratio in comparison to the optimal deterministic strategy on O-D edge-disjoint graphs, we conclude that randomization can improve the expected competitive performance of the online strategies for the k-CTP in presence of multiple agents and communication. We also prove that the competitive ratio of the optimal randomized strategy does not improve on O-D edge disjoint graphs, when the case with complete communication is compared to the case with limited communication.

These results have been published in [52].

Analysis of online strategies on graphs having common edges on the O-D paths

In the third section of Chapter 3, we provide an improved lower bound on the competitive ratio of deterministic strategies by analyzing graphs in which the O-D paths have common edges. By this way we show that no deterministic strategy achieves the lower bound of $2\lfloor \frac{k}{L} \rfloor + 1$ given in [74].

1.5.3 Chapter 4 (Online minimum latency problem with edge uncertainty)

A lower bound of $2k+1$ has been derived for the competitive ratio of deterministic online strategies for OMLP in [70]. However, a deterministic online strategy which meets the lower bound of $2k+1$ is not provided. In this chapter, we prove that the lower bound of $2k+1$ is tight by introducing an optimal deterministic online strategy whose competitive ratio matches the lower bound. Furthermore, we prove that no randomized online strategy can achieve an expected competitive ratio better than $k+1$ for OMLP.

1.5.4 Chapter 5 (Online discrete search problem with traveling and search costs on undirected graphs)

Several past studies have conducted competitive analysis on the variants of online continuous search problems which are defined on graphs, see [8], [26] and [35]. However, to the best of our knowledge, online discrete search problems have not been studied from the competitive analysis point of view. In this chapter, we investigate an online discrete search problem which we call the ODSP from the competitive analysis perspective for the first time. The ODSP finds applications in diverse areas such as security, defense, and search-and-rescue. We provide policies that are optimal with respect to the worst-case scenarios for such applications. We derive a tight lower bound on the competitive ratio of deterministic strategies and propose an optimal deterministic strategy. We also provide a tight lower bound on the expected competitive ratio of randomized strategies and prove its tightness by introducing an optimal randomized strategy. In this way, we show that randomized strategies can achieve a better competitive ratio in comparison to deterministic strategies for the ODSP in the expected sense.

1.6 Significance of the study

In this thesis, we investigate several online optimization problems in which some part of input is incomplete and information is revealed incrementally while implementing a solution strategy. In situations when complete information is not a priori available but becomes gradually available while making decisions according to the actions taken, online optimization approach addresses the underlying problems better, in comparison to optimization approaches where complete input information is assumed. For these problems, it is essential to develop strategies that make effective decisions online, as information is revealed.

The objective of this thesis is to design and analyze online solution strategies for a selection of online optimization problems within the field of Network Optimization. We study four online optimization problems which are defined in the context of navigation and search on networks; namely, the online k-Canadian Traveler Problem, the online multi-agent k-Canadian Traveler Problem, the online Minimum Latency Problem with edge uncertainty, and the online Discrete Search Problem with traveling and search costs on undirected graphs. These problems find applications in various areas such as disaster response, search-and-rescue, security, and defense. In such application areas, for instance, in a relief operation during disaster response (which can be regarded as an application of all of the aforementioned problems), the first responder(s) would start the operation without waiting for complete information to save from time. In such operations, having solution strategies with a good worst-case performance is vital since human life is at stake. This necessity motivates us to investigate the aforementioned problems from the competitive analysis perspective, which is a standard worst-case measure in comparison to the offline optimum, over all possible instances of the problem, to evaluate the performance of online solution strategies. We obtain novel results including proven optimal policies and characterization of worst-case scenarios. The policies guide the decision makers and the worst-case scenarios are useful to identify how an adversary would behave so that necessary precautions can be taken.

Chapter 2

ONLINE K-CANADIAN TRAVELER PROBLEM

2.1 Introduction

The online Canadian Traveler Problem (CTP) is a navigation problem under incomplete information. A traveling agent receives an undirected graph $G = (V, E)$ with a given source node O and a destination node D, together with non-negative edge costs as input. The agent is located at O initially. There are some blocked edges in the graph, but these edges are not known to the agent. The agent discovers the status of an edge when he reaches an end-node of the edge. The objective is to provide an online strategy such that the agent finds a feasible path, i.e. one without blocked edges from O to D with minimum total cost of the edges taken by the agent. When an upper bound k ($k \geq 1$) on the number of blocked edges is given as input, the problem is called the k-CTP.

To evaluate the performance of online strategies, the notion of *competitive ratio* has been introduced by Sleator and Tarjan [55] and adopted by many researchers. For a deterministic strategy, the competitive ratio is the maximum ratio of the cost of the online strategy to the cost of the offline strategy over all instances of the problem. For a randomized strategy, the competitive ratio is the maximum ratio of the expected cost of the online strategy to the cost of the offline strategy over all instances of the problem. In the offline k-CTP, the blocked edges are removed from the graph. Hence, it reduces to a shortest path problem.

2.1.1 Our Contributions

In this study we reconsider the randomized online strategy that is presented for the k-CTP on graphs where all O-D paths are node-disjoint in [4]. This strategy can be regarded as the randomized version of the backtrack strategy according to Bender and Westphal. Hence we call it the *randomized backtrack strategy*, in short the RBS.

We show that a particular property regarding the costs of the O-D paths must hold to implement the RBS. We formally specify this property. Next we show that this property does not necessarily hold when the costs of the O-D paths in the input graph are arbitrary and $k > 2$. That is we prove that the RBS is not applicable on graphs that consist of only node-disjoint O-D paths, where the costs of the O-D paths are arbitrary and $k > 2$. Moreover, we modify the RBS and introduce an optimal strategy which is applicable on graphs where all O-D paths are node-disjoint and the costs of the O-D paths are arbitrary.

2.2 Preliminaries

We assume that the input graph $G = (V, E)$ contains only node-disjoint O-D paths. We denote the number of node-disjoint O-D paths in the graph by n. We denote the cost of an O-D path P_i $(i = 1, 2, ..., n)$ by c_i, where c_i is the sum of the costs of the edges on P_i. We assume that the graph remains connected if all of the blocked edges are removed from it. Note that $n \geq 2$, since $k \geq 1$. We define $t = min\{k + 1, n\}$.

Before we explain our results, we need to explain the RBS. We first present the following definition that is taken from [4].

Definition 2.2.1. The paths $P_1, P_2, ..., P_n$ with costs $c_1 \leq c_2 \leq ... \leq c_n$ have the *similar costs property* if for all $i = 1, 2, ..., n$ it holds that

$$c_i \leq \frac{2}{n} \sum_{j=1}^{n} c_j.$$

Suppose that the O-D paths $P_1, P_2, ..., P_n$ with costs $c_1, c_2, ..., c_n$ satisfy the similar costs property. In this case regarding the random selection of one O-D path among $P_1, P_2, ..., P_n$, the RBS constructs the probability distribution $\Omega_n = (p_1, p_2, ..., p_n)$ as follows [4]. $\Omega_n = \lambda^* p^{'}$, where Ω_n and $p^{'}$ are n-vectors, $\lambda^* = \sum_{i=1}^{n} \frac{1}{p_i^{'}} \in [0, 1]$ and

$$p_i^{'} = \frac{(2-n)c_i + \sum_{j=1 j\neq i}^{n} 2c_j}{n^2 c_i},$$

for $i = 1, ..., n$.

Note that $\lambda^* \in [0, 1]$ when the costs of the O-D paths $P_1, ..., P_n$ are arbitrary [4], i.e. $P_1, ..., P_n$ should not necessarily satisfy the similar costs property to ensure that $\lambda^* \in [0, 1]$. We also note that it is assumed that $P_1, P_2, ..., P_n$ fulfill the similar costs property to ensure the non-negativity of $p_1, ..., p_n$ [4]. That is, the RBS only uses Ω_n when $P_1, P_2, ..., P_n$ have the similar costs property, i.e. when $p_1, p_2, ..., p_n$ are non-negative. Bender and Westphal [4] proved the following lemma regarding the probability distribution Ω_n.

Lemma 2.2.1. *Suppose that the O-D paths $P_1, P_2, ..., P_n$ with costs $c_1 \leq c_2 \leq ... \leq c_n$ satisfy the similar costs property. In this case the probability distribution $\Omega_n = (p_1, p_2, ..., p_n)$, belongs to the polyhedron Q_n which is defined as*

$$Q_n = \{p \in \mathbb{R}_+^n : (2-n)p_i + \sum_{j=1 j\neq i}^{n} 2\frac{c_j}{c_i} p_j \leq 1 \;\; \forall i = 1, 2, ..., n, \sum_{i=1}^{n} p_i = 1\}.$$

Now we can describe the RBS.

2.2.1 Description of the RBS

Recall that $t = min\{k+1, n\}$. Note that $t \geq 2$, since $k \geq 1$ and $n \geq 2$. The RBS partitions the t shortest O-D paths in the graph into classes of O-D paths as follows. Initially an empty class is opened. The strategy sorts the t shortest O-D paths in the graph in non-decreasing order of their costs. O-D paths are added to the open class as long as possible such that the similar costs property holds. If adding an additional O-D path violates the property, the currently open class is closed and a new class is opened. Partitioning the O-D paths into classes continues until the tth shortest O-D path in the graph is assigned to an open class. The classes are maximal with respect to the similar costs property, i.e. they cannot be extended without violating the property. Moreover such a classification of the O-D paths always exists and it is unique up to permutations of the O-D paths that have the same costs. For a graph that contains only node-disjoint O-D paths, let $Class_1, Class_2, ..., Class_L$ be the unique classification that is constructed as described above, i.e. $Class_1, Class_2, ..., Class_L$ are the constructed classes and $L \geq 1$. The classes are sorted in ascending order of costs of their O-D paths, i.e. the costs of the O-D paths in $Class_1$ are less than the costs of the O-D paths in the other classes. Let n_l denote the number of O-D paths in $Class_l$ for $l = 1, ..., L$. Now we can explain the rest of the strategy.

The classes are processed in ascending order, i.e. $Class_1$ is processed first, as follows. We call the class that is being processed by the agent the *current* class, i.e. $Class_1$ is the current class at the beginning. Initially, the agent constructs the probability distribution $\Omega_{n_1} = (p_1, ..., p_{n_1})$ and takes the O-D path P^* ($P^* \in Class_1$), according to it. We note that $p_1, ..., p_{n_1}$ are non-negative since the n_1 O-D paths in $Class_1$ satisfy the similar costs property. If P^* is not blocked, the agent arrives at D and the RBS ends. Otherwise, the agent backtracks to O and removes P^* from $Class_1$. However the O-D paths in $Class_1 - \{P^*\}$ do not necessarily obey the similar costs property. Here we specify an implicit assumption that is used in the RBS.

Definition 2.2.2. Suppose that the t shortest O-D paths in the graph are partitioned into classes $Class_1, Class_2, ..., Class_L$ by the RBS. let $P_1^l, P_2^l, ..., P_{n_l}^l$ be the O-D paths in $Class_l$ for $l = 1, ..., L$. The O-D paths in $Class_l$ have the *strong similar costs property* if and only if the O-D paths in any non-empty subset of $\{P_1^l, P_2^l ..., P_{n_l}^l\}$ satisfy the similar costs property.

Assuming that the O-D paths in $Class_1$ fulfill the strong similar costs property, the agent constructs the probability distribution $\Omega_{n_1-1} = (p_1, ..., p_{n_1-1})$ and takes an O-D path $P^{*'}$ among the remaining O-D paths in $Class_1$ according to Ω_{n_1-1}. If $P^{*'}$ is not blocked, the agent arrives at D and the strategy ends; otherwise, the agent backtracks to O and removes $P^{*'}$ from $Class_1$. The procedure is repeated until the agent arrives at D or all of the O-D paths in $Class_1$ are taken. In the latter case, the agent processes the next class, i.e. $Class_2$. The same procedure is then repeated until the agent arrives at D.

2.3 Consideration of the RBS on graphs having only node-disjoint O-D paths

We consider the implementation of the RBS on graphs that contain n node-disjoint O-D paths. We show that there are cases in which the RBS cannot be implemented.

Definition 2.3.1. We call the set of n ($n \geq 4$) O-D paths $P_1, P_2, ..., P_n$ with costs $c_1, c_2, ..., c_n$, the *adversary set of n O-D paths* (A^n); if $c_1, c_2, ..., c_n$ satisfy the following conditions. 1) $c_1 = c_2 = 1$ and 2) $c_i = 2(i-1)$ for $3 \leq i \leq n$.

Theorem 2.3.1. *When $k \geq 3$ in the k-CTP, the RBS cannot be implemented on A^{k+1}.*

Proof. Consider A^{k+1} that contains the O-D paths $P_1, P_2, ..., P_k, P_{k+1}$ with costs $1, 1, ..., 2(k-1), 2k$ for $k \geq 3$. Note that $P_1, P_2, ..., P_{k+1}$ have the similar costs property according to Definitions 5.5.1 and 2.3.1. The RBS initially partitions the O-D paths into a single class $Class_1$, i.e. $Class_1$ contains $P_1, P_2, ..., P_{k+1}$. Note that only one class is constructed since $P_1, P_2, ..., P_{k+1}$ fulfill the similar costs property. Then the agent takes one of the O-D paths in $Class_1$ according to the probability distribution $\Omega_{k+1} = (p_1, p_2, ..., p_{k+1})$, i.e. $p_i = \lambda^*(\frac{(1-k)c_i + \sum_{j=1 j \neq i}^{k+1} 2c_j}{(k+1)^2 c_i})$ for $i = 1, 2, ..., k+1$. Note that $p_1, p_2, ..., p_{k+1}$ are non-negative since $P_1, P_2, ..., P_{k+1}$ satisfy the similar costs property. Suppose that P_k is chosen (this case happens with probability p_k). If P_k is not blocked, the agent arrives at D and the strategy ends; otherwise, the agent finds a blocked edge and backtracks to O. Suppose that the latter case happens. At this stage of the strategy the O-D path with cost $2(k-1)$ is excluded from $Class_1$ and the other k O-D paths remain in $Class_1$. Let $P_1^{'}, P_2^{'}, ..., P_{k-1}^{'}, P_k^{'}$ with costs $c_1^{'}, c_2^{'}, ..., c_{k-1}^{'}, c_k^{'}$ denote the remaining O-D paths in $Class_1$, i.e. $c_1^{'} = c_2^{'} = 1$, $c_i^{'} = 2(i-1)$ for $3 \leq i \leq k-1$ and $c_k^{'} = 2k$. Observe that $P_1^{'}, P_2^{'}, ..., P_{k-1}^{'}, P_k^{'}$ do not fulfill the similar costs property by Definition 5.5.1, i.e. $c_k^{'} > \frac{2}{k}\sum_{j=1}^{k} c_j^{'}$. Here, the agent selects one of the k remaining

O-D paths in $Class_1$ according to the probability distribution $\Omega_k = (p_1^{'}, p_2^{'}, ..., p_k^{'})$, i.e. $p_i^{'} = \lambda^*(\frac{(2-k)c_i^{'}+\sum_{j=1 j\neq i}^{k} 2c_j^{'}}{k^2 c_i^{'}})$ for $i = 1, 2, ..., k$. Now suppose $P_k^{'}$ is selected and the corresponding probability is

$$p_k^{'} = \lambda^*(\frac{(2-k)c_k^{'} + \sum_{j=1}^{k-1} 2c_j^{'}}{k^2 c_k^{'}}) = \lambda^*(\frac{(2-k)2k + 2 + 2 + 2\sum_{j=3}^{k-1} 2(j-1)}{2k^3}),$$

which is equal to

$$\lambda^*(\frac{(2-k)2k + 2\sum_{j=1}^{k-2} 2j}{2k^3}) = \lambda^*(\frac{(2-k)2k + 2(k-2)(k-1)}{2k^3}).$$

Note that $\lambda^* \in [0, 1]$ according to [4]. It is straightforward to show that the right hand side is decreasing in k, i.e. $p_k^{'}$ is decreasing in k. Also observe that $p_k^{'}$ is negative when $k = 3$. Hence $p_3^{'}$ is negative for $k \geq 3$. This contradicts with the rationale of the RBS, since the RBS assumes that $p_1^{'}, p_2^{'}, ..., p_k^{'}$ are non-negative. The theorem follows. □

As an illustrative example for the proof of Theorem 2.3.1, let $k = 3$. Consider A^4 that contains the O-D paths P_1, P_2, P_3, P_4 with costs $c_1 = 1, c_2 = 1, c_3 = 4, c_4 = 6$, respectively. Observe that P_1, P_2, P_3, P_4 satisfy the similar costs property by Definition 5.5.1. The RBS initially partitions the O-D paths into a single class $Class_1$. Then, the agent takes one of the O-D paths in $Class_1$ according to the probability distribution $\Omega_4 = (\frac{15}{32}, \frac{15}{32}, \frac{2}{32}, 0)$. Suppose that P_3 is chosen (this happens with probability $\frac{2}{32}$). If P_3 is not blocked the agent arrives at D and the strategy ends; otherwise, the agent finds a blocked edge and backtracks to O. Suppose that the latter case happens. P_3 is excluded from $Class_1$ and P_1, P_2, P_4 remain in $Class_1$. Observe that P_1, P_2, P_4 do not fulfill the similar costs property by Definition 5.5.1, i.e. $c_4 = 6 > \frac{2}{3}(c_1 + c_2 + c_4) = \frac{16}{3}$. The RBS fails to construct a probability distribution $\Omega_3 \in Q_3$ for the selection of one O-D path among P_1, P_2, P_4.

Remark 2.3.1. We note that the implementation of the RBS will fail in any instance of the input graph in which the O-D paths in at least one of the constructed classes do not have the strong similar costs property, based on the similar reason that is discussed in the proof of Theorem 2.3.1. In other words, the adversary sets of O-D paths do not cover all of the instances in which the implementation of the strategy fails.

The RBS can be implemented in all of the instances of the input graph in which the O-D paths in the constructed classes have the strong similar costs property. Below we show that such instances exist in special cases.

Theorem 2.3.2. *Suppose that $P_1, P_2, ..., P_t$ with costs $c_1 \leq c_2 \leq ... \leq c_t$ are the t shortest O-D paths in the input graph, where $t = min\{k+1, n\}$. When $c_t \leq 2c_1$, the RBS is implementable.*

Proof. Let $P = \{P_1, P_2, ..., P_t\}$ denote the set of t shortest O-D paths in the input graph. Observe that O-D paths in P satisfy the similar costs property by Definition 5.5.1, i.e. the RBS creates a single class. Let $P' = \{P'_1, P'_2, ..., P'_{n_{p'}}\}$ (with costs $c'_1 \leq ... \leq c'_{n_{p'}}$) be an arbitrary and non-empty subset of P, where $n_{p'}$ is the number of O-D paths in P'. We need to show that

$$c'_i \leq \frac{2}{n_{p'}} \sum_{j=1}^{n_{p'}} c'_j,$$

for $i = 1, 2, ..., n_{p'}$. Note that the costs of the O-D paths in P' are at least c_1. Hence the right-hand side is at least $2c_1$. Thus the O-D paths in the constructed class have the strong similar costs property and the RBS can be implemented. The theorem follows. □

Below, we also specify the cases in which the RBS is applicable when the costs of the O-D paths in the input graph are arbitrary.

Theorem 2.3.3. *The RBS is implementable and achieves the competitive ratio of at most $k+1$, when $k \leq 2$.*

Proof. Note that when $k = 1$, there are at least two O-D paths in the graph since we assume that the graph remains connected if the blocked edge is removed from it.

The proof is shown for $k = 1$ in [4]. For $k = 2$ and $n = 2$, the RBS achieves the competitive ratio of two based on the same proof that is presented in [4], for $k = 1$. (For $t = 2$, the RBS achieves the competitive ratio of two).

For $k = 2$ and $n \geq 3$, the RBS considers three shortest O-D paths in the input graph, i.e. P_1, P_2, P_3 with costs $c_1 \leq c_2 \leq c_3$. Two cases might happen when the RBS is implemented.

- Two classes $Class_1$ and $Class_2$ are constructed, i.e. $Class_1 = \{P_1, P_2\}$ and $Class_2 = \{P_3\}$. In this case the O-D paths in $Class_l$ ($l \in \{1, 2\}$) fulfill the similar costs property according to the RBS. Moreover, any other non-empty subset of $Class_l$ ($l \in \{1, 2\}$) contains at most one O-D path that satisfies the similar costs property by Definition 5.5.1.

- Only $Class_1 = \{P_1, P_2, P_3\}$ is constructed. In this case P_1, P_2, P_3 fulfill the similar costs property according to the RBS. Moreover any other non-empty subset of $Class_1$ contains at most two O-D paths that satisfy the similar costs property by Definition 5.5.1.

Hence the O-D paths in the same class have the strong similar costs property, when $k = 2$ and $n \geq 3$. Thus the RBS meets the competitive ratio of $k + 1$ when $k \leq 2$. □

For $k \geq 3$, the RBS fails to be implemented on A^{k+1} (Definition 2.3.1) according to Theorem 2.3.1 and fails in any instance of the input graph in which the O-D paths in at least one of the constructed classes do not have the strong similar costs property according to Remark 2.3.1.

2.4 Optimal randomized strategy on graphs having only node-disjoint O-D paths

In this section we present a modification of the RBS which is implementable on graphs where all O-D paths are node-disjoint. We call this strategy the *modified randomized* *backtrack strategy*, in short the M-RBS. Before we explain the M-RBS, we present the following strategy.

2.4.1 A strategy for partitioning the O-D paths into classes

We present a strategy for dividing a set of O-D paths into classes such that the O-D paths in the same class satisfy the similar costs property. We call this strategy the *partitioning strategy*. The partitioning strategy decomposes the O-D paths into classes by going through them in non-decreasing order of their costs. The O-D paths are added to the currently open class as long as they obey the similar costs property. The classes are maximal in the sense that they cannot be extended without violating the similar costs property. We note that the partitioning strategy is used in both of the RBS and the M-RBS.

The partitioning strategy:

- Initialization. Take a set of n O-D paths together with their costs as input. Label the O-D paths from P_1 to P_n such that $c_1 \leq c_2 \leq ... \leq c_n$. Let i and l be counter variables and set their initial values to one.

- Step 1. Create an empty class and call it $Class_l$, then go to Step 2.

- Step 2. If P_i fulfills the similar costs property with the O-D paths in $Class_l$, then add P_i to $Class_l$, set $i = i + 1$ and go to Step 3. Otherwise, close $Class_l$, i.e. no more O-D paths are added to $Class_l$. Set $l = l + 1$ and go to Step 1.

- Step 3. If $i > n$, close $Class_l$ and then stop, i.e. the n O-D paths are decomposed into $L = l$ new classes. Otherwise, go to Step 2.

Note that sorting the O-D paths with respect to their costs can be done in $O(nlogn)$. Also note that partitioning the sorted O-D paths into classes can be done in $O(n)$. Thus, the running time of the partitioning strategy is $O(nlogn)$. Below, we describe how it is possible to apply the partitioning strategy to make the RBS implementable on graphs where all O-D paths are node-disjoint.

2.4.2 Modification of the RBS

As described before, the RBS is not applicable when the remaining O-D paths in the current class do not satisfy the similar costs property. To solve this problem, we present a modification of the RBS which we call the M-RBS. In the M-RBS, when the remaining O-D paths in the current class do not fulfill the similar costs property, the M-RBS decomposes them into new classes by applying the partitioning strategy. Then the M-RBS processes the class among the newly constructed classes that contains the shortest O-D path in the graph. Note that in this case the M-RBS is implementable since the partitioning strategy generates the classes such that the O-D paths in the same class obey the similar costs property. Now we can formally describe the M-RBS.

Modified randomized backtrack strategy (M-RBS):

- Initialization. Take a graph $G = (V, E)$ that contains n node-disjoint O-D paths $P_1, P_2, ..., P_n$ together with k as input. Let $c_1, c_2, ..., c_n$ represent the costs of $P_1, P_2, ..., P_n$. Compute $c_1, c_2, ..., c_n$. Define S as an empty set.

- Step 1. Partition the t ($t = min\{k + 1, n\}$) shortest O-D paths into classes by applying the partitioning strategy. Add the constructed classes to S and go to Step 2.

- Step 2. Let L denote the number of classes in S. Label the L classes in S from 1 to L arbitrarily, i.e. $Class_1, Class_2, ..., Class_L$. Let SP_l ($l = 1, ..., L$) be the cost of the shortest O-D path in $Class_l$. Identify $Class_{l^*}$ ($l^* \in \{1, 2, ..., L\}$) as the current class such that $SP_{l^*} = \min\{SP_l\}$ for $l = 1, 2, ..., L$. Then go to Step 3.

- Step 3. Let w denote the number of O-D paths in the current class ($Class_{l^*}$).

Take an O-D path P^* among the w O-D paths in the current class according to the probability distribution Ω_w. If the agent has arrived at D, stop; otherwise, backtrack to O and remove P^* from the current class. If the current class is empty, remove it from S and go to Step 2. Otherwise, go to Step 4.

- Step 4. If the remaining O-D paths in the current class satisfy the similar costs property, go to Step 3. Otherwise, partition the remaining O-D paths in the current class into new classes by applying the partitioning strategy. Add the newly generated classes to S. Also remove the current class ($Class_{l^*}$) from S. Then go to Step 2.

Since G contains only node-disjoint O-D paths, computing the costs of the O-D paths can be done in $O(|E|)$. Hence the running time of the initialization of the M-RBS is $O(|E|)$. Note that Step 1 is implemented only once and Step 4 is implemented at most $t \leq n$ times. Also recall that the running time of the partitioning strategy for partitioning n O-D paths into the classes is $O(nlogn)$. Thus the running times of Steps 1 and 4 are at most $O(nlogn)$ and $O(n^2logn)$, respectively. Identifying the current class in Step 2 can be done in $O(t)$ ($t < n$) and Step 2 is implemented at most t times. Therefore the running time of Step 2 is at most $O(n^2)$. Constructing the probability distribution Ω_w ($w \leq n$) can be done in $O(w)$ in Step 3 of the M-RBS. Since Step 3 is implemented at most $t \leq n$ times, the running time of Step 3 is at most $O(n^2)$. Here we note that $n \leq |E|$. Thus the running time of the M-RBS can be bounded from above by $O(|E|^2log|E|)$.

Remark 2.4.1. Note that the classes that are constructed in Step 1 are exactly the same as the classes constructed in the RBS. Also note that when the O-D paths in the current class have the strong similar costs property and the M-RBS enters Step 4, the remaining O-D paths in the current class always satisfy the similar costs property. That is, the generated classes are removed from S only if they become empty, i.e. the remaining O-D paths in them never decompose into new classes in Step 4. Thus the constructed classes in Step 1 are processed in non-decreasing order of costs of their O-D paths, exactly the same way as they are processed in the RBS. Hence, we conclude that the RBS and the M-RBS are equivalent when the O-D paths that are added to the same class in Step 1 possess the strong similar costs property.

Below we prove that the M-RBS is optimal when the t shortest O-D paths in the graph satisfy the similar costs property.

Lemma 2.4.1. *Suppose that the M-RBS is implemented on the input graph G, such that the t ($t = min\{k+1, n\}$) shortest O-D paths $P_1, P_2, ..., P_t$ with costs $c_1 \leq c_2 \leq ... \leq c_t$ in G fulfill the similar costs property and at least one of them is traversable. In this case, the M-RBS achieves the competitive ratio of $t \leq k+1$.*

Proof. We present the proof by induction on t.

- **Base Case**. Note that $t \geq 2$, since $k \geq 1$ and $n \geq 2$. When $t = 2$, the M-RBS constructs a single class that contains two O-D paths. Thus the number of O-D paths in the current class never exceeds two. Also note that any one or two O-D paths with arbitrary cost(s) satisfy the similar costs property according to Definition 5.5.1. Hence the O-D paths in the constructed class have the strong similar costs property. Thus RBS and M-RBS are equivalent according to Remark 2.4.1. Since the RBS is $2-$competitive for $t = 2$ according to the proof of Theorem 2.3.3, the base case follows.

- **Induction**. Suppose that the claim is correct for $x \leq t-1$. Since the t shortest O-D paths satisfy the similar costs property, the M-RBS initially constructs a single class that contains t O-D paths. Then the agent takes one of the t O-D paths $P_1, P_2, ..., P_t$ in the class with probability distribution $\Omega_t = (p_1, p_2, ..., p_t)$. Let B and T denote the set of blocked O-D paths and the set of traversable O-D paths in the class, respectively. Also let P_{i^*} ($i^* \in \{1, 2, ..., t\}$) with cost c_{i^*} denote the offline optimum and p_{i^*} be the probability that P_{i^*} is taken. The competitive ratio can be shown as

$$\sum_{i \in T} \frac{p_i c_i}{c_{i^*}} + \sum_{j \in B} p_j \frac{2c_j + C^{t-1}}{c_{i^*}}.$$

If the taken O-D path is traversable, i.e. the taken path is P_i ($i \in T$), then the agent arrives at D (the M-RBS stops) by incurring a cost of c_i; otherwise, it means that a path P_j ($j \in B$) is selected and the agent backtracks to O by incurring a cost of $2c_j$. Note that C^{t-1} denotes the cost of the strategy from the time that the agent arrives back to O until the end of the strategy in expression above. If the selected path P_j ($j \in B$) is blocked the agent removes P_j from the class and backtracks to O, then two cases might happen.

- Case 1. The remaining O-D paths in the class fulfill the similar costs property. In this case C^{t-1} is at most $(t-1)c_{i^*}$ by the induction assumption for $x = t-1$.
- Case 2. The remaining O-D paths in the class do not fulfill the similar costs property. In this case the remaining O-D paths in the class are decomposed into new classes by applying the partitioning strategy. Suppose that the classes $Class_1, Class_2, ..., Class_L$ are the constructed classes that are sorted in non-decreasing order of the costs of their O-D paths, i.e. the costs of the O-D paths in $Class_1$ are less than the costs of the O-D paths in the other classes. Let n_l ($l = 1, 2, ..., L$) be the number of O-D paths in the lth class, i.e. $\sum_{l=1}^{L} n_l = t-1$. Let c_w^l denote the cost of the wth ($w = 1, 2, ..., n_l$) O-D path in the lth ($l = 1, 2, ..., L$) class, if the O-D paths in $Class_l$ are sorted in non-decreasing order of their costs. Now suppose that P_{i^*} ($i = 1, 2, ..., t$) is contained in some class, i.e. $Class_{l^*}$ ($l^* \in \{1, 2, ..., L\}$). The agent first takes all of the O-D paths in the classes $Class_1, Class_2, ..., Class_{(l^*-1)}$ and incurs the costs of $\sum_{l=1}^{l^*-1} 2 \sum_{w=1}^{n_l} c_w^l$. This is because the agent always processes the class that contains the shortest O-D path in the graph according to Step 2 of the M-RBS. Then the agent processes $Class_{l^*}$. Here, we use an observation which is also applied in [4].

Observation 4.1. Note that when the agent finds a blocked edge on an

O-D path, he removes the O-D path from the current class. Hence, the agent may find at most $n_{l^*} - 1 \leq t - 2$ blocked edges on the O-D paths in $Class_{l^*}$ since $P_{i^*} \in Class_{l^*}$. Because the n_{l^*} O-D paths in $Class_{l^*}$ obey the similar costs property, finding P_{i^*} among the O-D paths in $Class_{l^*}$ within the competitive ratio of n_{l^*} is equivalent to the induction assumption for $x = n_{l^*}$.

Hence C^{t-1} can be written as

$$\sum_{l=1}^{l^*-1} 2 \sum_{w=1}^{n_l} c_w^l + n_{l^*} * c_{i^*}.$$

Since the O-D paths in $Class_{l^*}$ do not fulfill the similar costs property with the O-D paths in $Class_l$ ($l = 1, 2, ..., l^* - 1$), we have $(n_l - 1)c_1^{l^*} > 2\sum_{w=1}^{n_l} c_w^l$ for $l = 1, 2, ..., l^* - 1$ according to Definition 5.5.1. Also note that $c_1^{l^*} \leq c_{i^*}$. Hence we can bound C^{t-1} from above by

$$\sum_{l=1}^{l^*-1} (n_l - 1) * c_{i^*} + n_{l^*} * c_{i^*},$$

which is equivalent to

$$(-l^* + 1 + \sum_{l=1}^{l^*} n_l)c_{i^*} \leq (\sum_{l=1}^{l^*} n_l)c_{i^*} \leq (t-1)c_{i^*}.$$

We just showed that C^{t-1} is at most $(t-1)c_{i^*}$. Now we can present the rest of our proof by replacing C^{t-1} with $(t-1)c_{i^*}$ in the competitive ratio to bound it from above by

$$\sum_{i \in T} \frac{p_i c_i}{c_{i^*}} + \sum_{j \in B} p_j (\frac{2c_j}{c_{i^*}} + t - 1) \leq p_{i^*} + \sum_{j=1 j \neq i^*}^{t} p_j (\frac{2c_j}{c_{i^*}} + t - 1).$$

We need to show that the right hand side is at most t for all instances, i.e.

$i^* = 1, 2, ..., t$. This is equivalent to

$$(2-t)p_i + \sum_{j=1 j\neq i}^{t} 2\frac{c_j}{c_i}p_j \leq 1,$$

for all $i^* = 1, 2, ..., t$. Note that the agent selects the O-D paths based on the probability distribution $\Omega_t = (p_1, p_2, ..., p_t)$ such that $\Omega_t \in Q_t$ according to Lemma 5.5.1. Hence the inequality above is valid and the lemma follows.

□

Next we apply Lemma 2.4.1 to prove the optimality of the M-RBS on graphs where all O-D paths are node-disjoint.

Theorem 2.4.1. *The competitive ratio of the M-RBS is at most $k+1$ on graphs where all O-D paths are node-disjoint.*

Proof. Note that the M-RBS uses the partitioning strategy in Step 1 and decomposes the t shortest O-D paths of the input graph into classes at the beginning. Suppose that the classes $Class_1, Class_2, ..., Class_L$ are constructed in Step 1, such that $Class_l$ $(l = 1, 2, ..., L)$ contains n_l O-D paths. Let c_w^l denote the cost of the wth O-D path in $Class_l$ $(l = 1, 2, ..., L)$ if the O-D paths in $Class_l$ are sorted in non-decreasing order of their costs. Note that the n_l O-D paths in $Class_l$ $(l = 1, 2, ..., L)$ obey the similar costs property and $\sum_{l=1}^{L} n_l = t$. Also suppose that the classes are sorted such that $c_1^1 \leq c_1^2 \leq ... \leq c_1^L$. Let P_{i^*} with cost c_{i^*} be the offline optimum.

The M-RBS processes the classes in ascending order of costs of their O-D paths, i.e. $Class_1$ is processed first. Suppose that the offline optimum (P_{i^*}) belongs to some class $Class_{l^*}$ $(l^* \in \{1, 2, ..., L\})$. In this case the agent first takes the O-D paths in the classes $Class_1, ..., Class_{l^*-1}$ and incurs the costs of $\sum_{l=1}^{l^*-1} 2\sum_{w=1}^{n_l} c_w^l$. This is because the agent always processes the class that contains the shortest O-D path in the graph according to Step 2 of the M-RBS. Then the agent processes $Class_{l^*}$ and finds the offline optimum by incurring a cost of at most $n_{l^*} * c_{i^*}$ according to Observation 4.1.

Table 2.1: Summary of the results in Chapter 2

Problem	Result	Case	Network Type	Publication Status
k-CTP	Optimal strategy	Randomized	Node-disjoint O-D paths	Published in [53]

Hence the competitive ratio can be written as

$$\sum_{l=1}^{l^*-1} 2 \sum_{w=1}^{n_l} c_w^l + n_{l^*} * c_{i^*}.$$

Similar to the proof of Lemma 2.4.1, $2\sum_{w=1}^{n_l} c_w^l \leq (n_l - 1)c_{i^*}$. The competitive ratio is bounded above by

$$\sum_{l=1}^{l^*-1} (n_l - 1) * c_{i^*} + n_{l^*} * c_{i^*},$$

which is equivalent to

$$(-l^* + 1 + \sum_{l=1}^{l^*} n_l)c_{i^*} \leq (\sum_{l=1}^{l^*} n_l)c_{i^*} \leq tc_{i^*}.$$

The theorem follows since $t \leq k+1$ and the cost of the offline optimum is c_{i^*}. □

2.5 Concluding remarks

We reconsidered the implementation of the RBS on graphs which contain n node-disjoint O-D paths. We showed that to implement the strategy, a certain property (strong similar costs property) regarding the costs of the O-D paths in the input graph must hold. That is we proved that the RBS is not applicable in some cases when $k > 2$. We showed that the RBS is applicable when the cost of the $(min\{k+1, n\})$th shortest O-D path is at most twice of the shortest path in the input graph. Furthermore we modified the RBS to obtain an optimal strategy which is applicable on graphs having only node-disjoint O-D paths.

Table 2.1 summarizes the results of Chapter 2.

Chapter 3

ONLINE MULTI-AGENT K-CANADIAN TRAVELER PROBLEM

In this chapter, we investigate deterministic and randomized strategies for the multi-agent k-CTP. In the first two sections of this chapter, we consider the problem on O-D edge-disjoint graphs. In the third section, we analyze the problem on graphs having common edges on the O-D paths.

3.1 Analysis of deterministic online strategies

3.1.1 Introduction

The online multi-agent O-D k-Canadian Traveler Problem is a navigation problem, where traveling agents receive a graph with a given source node O and a destination node D together with edge costs as input. Initially all of the agents are located at O. At most k edges are blocked in the graph but these edges are not known to the agents (travelers). An agent discovers the status of an edge when he/she reaches an end node of the edge. The objective is to provide an online strategy such that at least one of the agents finds a feasible path, i.e. one without blocked edges, from its initial location O to the given destination D with minimum total cost of the edges taken by the agent that finds a feasible path first. This is called the *route* of that agent. The problem is an online optimization problem that generalizes the k-Canadian Traveler Problem (k-CTP) by the existence of multiple agents. To measure the performance of online strategies, *competitive ratio* has been introduced by Sleator and Tarjan [55]. The competitive ratio is the maximum ratio of the cost of the online strategy to the cost of the offline strategy over all instances of the problem. In our problem, in the offline problem the blocked edges are removed from the graph. Hence, it reduces to a shortest path problem.

Two versions of the online multi-agent k-CTP have been introduced in the literature, with complete and limited communication. Zhang et al. [74] studied the problem in presence of two levels of communication. In their article, P_1 denotes the problem where complete communication is available among the agents and P_2 is the problem where limited communication is possible such that some of the agents can only receive information. In their problem definition, RS-type agents can send and receive information, while R-type agents can only receive information.

3.1.2 Our Contributions

We focus on the case where communication among the agents is limited. By specifying the communication protocols among the agents, we define three levels of agents' intelligence. We introduce two simple deterministic online strategies and use them when the agents benefit from higher levels of intelligence. By this way, we provide updated lower bounds on the competitive ratio of deterministic online strategies for P_2. We also show that one of our strategies is optimal in both cases with complete and limited communication in the special case where the input graph has only O-D edge-disjoint paths between the given O-D pair. Formally we define an O-D edge-disjoint graph as an undirected graph G with a given source node O and a destination node D, such that any two distinct O-D paths in G are edge-disjoint, that is, they do not have a common edge. We need to mention that analyzing O-D edge-disjoint graphs is a standard restriction in the context of k-CTP and its variants. Finally, contrary to what is claimed in [74], we show that there are instances with O-D edge-disjoint graphs in which the competitive ratio of deterministic strategies on P_2 improves by increasing the number of R-type agents.

3.1.3 The online multi-agent O-D k-CTP with limited communication

The online multi-agent O-D k-CTP with limited communication has been introduced by Zhang et al. [74]. Limited communication means that some of the agents can both send and receive information (RS-type), but the others can only receive information (R-type). Suppose that there are L agents ($L < k$) and L_1 of them are RS-type and the other $L - L_1$ agents are R-type. Zhang et al. [74] assume that the agents will take the best decision applying the available information. This problem is defined as $P_2(L, L_1)$ in [74].

Zhang et al. [74] argue that $P_2(L, L_1)$ can be regarded as P_1 with L_1 agents with complete communication and at most $k - 1$ blockages. Based on this argument, they extend their obtained results for P_1 to P_2. By converting P_2 to its equivalent problem in P_1, they provide a lower bound on the competitive ratio of deterministic online strategies for P_2 by analyzing O-D edge-disjoint graph. They also use this argument to prove the competitive ratio of two common strategies, i.e. Retrace-Alternating Strategy and Greedy Strategy for P_2 after proving their competitive ratios for P_1. Below, we present some of the main assumptions in [74]. Note that we apply these assumptions to obtain our updated results.

(1) The number of blockages, k is larger than the number of the total agents, L.

(2) All of the R-type agents are regarded as one traveler based on the fact that they might re-take the infeasible paths which are already taken by other R-type agents.

(3) All of the agents will take the best possible decision with their given information.

Before introducing the new levels of agent's intelligence, we need to formalize some definitions on O-D edge-disjoint graphs. We say that the problem is at its *initial stage* when all of the agents are located at O initially. When at least one of the agents reaches D, we say that the problem is at its *final stage*. The decision of which O-D path to select whenever the agent is located at O is defined as the *decision* of an agent. Note that since the graph is O-D edge-disjoint, the agents take their decisions only when they are located at O. Let us assume that an agent walks back to O when he/she discovers a blocked edge. This is called *backtracking*. A closed walk of an agent starting at O and including exactly one backtracking or a walk of the agent starting at O and ending at D without backtracking are defined as the *travel* of an agent. We note that when the problem is at its final stage, a backtracking agent may not reach O but this does not affect the objective function. The *travel schedule* of an agent is defined as the set of the travels which the agent takes consecutively, from the initial stage to the final stage of the problem. We can characterize the available information to all of the agents as the topology of the input graph together with the edge costs, both explored blockages and travel schedules of the RS-type agents. Now, let us define the communication protocols between the agents.

Communication Protocol 1 (CP1). The RS-type agents can share the blockage information, in the sense that while any of the RS-type agents explores a new blockage, he/she will transmit the blockage information to the other RS-type and R-type agents. Note that this protocol has been applied in [74].

Communication Protocol 2 (CP2). The RS-type agents can transmit their own travel schedules to the R-type and RS-type agents. This protocol has been used in [74] as well.

We also define a third communication protocol which is not used in [74], to obtain our improved results.

Communication Protocol 3 (CP3). The RS-type agents can schedule for the R-type agents and transmit the planned travel schedule of each R-type agent to them. Having CP3 means that the RS-type agents can make decisions for the

R-type agents. Note that this protocol is realistic because all of the agents know the graph structure and real-time blockage information of the RS-type agents.

Although Zhang et al. [74] analyzed the strategies on O-D edge-disjoint graphs to provide a lower bound to the problem, they ignored the advantages of determining the travel schedules of each agent when the problem is at its initial stage. They also missed the advantages of using CP3. To utilize the advantages of determining the travel schedules at the initial stage of the problem and CP3, we formalize the definition of agents' intelligence by defining three levels of agents' intelligence.

I. Intelligence Level 1 (IL1). When an agent with IL1 is located at O, he/she is allowed to decide only his/her next travel. The RS-type agents are not allowed to plan their complete travel schedules at the initial stage of the problem. Moreover, having CP3 is not allowed at this level. We call such agents *IL1 agents*. Zhang et al. [74] have considered P_2 with IL1 agents. For the rest of this section, we call P_2 with IL1 agents P_{21}.

II. Intelligence Level 2 (IL2). The agents with IL2 have all properties of the IL1 agents. In addition, they are allowed to determine their complete travel schedule at the initial stage of the problem. However, the RS-type agents cannot utilize CP3 at this level. We call such agents *IL2 agents*. For the rest of this section, we call P_2 with IL2 agents P_{22}.

III. Intelligence Level 3 (IL3). The agents with IL3 have all properties of the IL2 agents. In addition, they can utilize CP3. We call such agents *IL3 agents*. For the rest of this section, we call P_2 with IL3 agents P_{23}.

Note that all of the notations P_{21}, P_{22} and P_{23} refer to problem P_2. We use them to show that how the results on P_2 will change as we utilize the advantages of planning at the initial stage and CP3.

3.1.4 The online multi-agent O-D k-CTP with IL2 agents (P_{22})

In this section, we study P_{22} on O-D edge-disjoint graphs to show that the previous lower bound on P_2 should be updated if the agents benefit from IL2. Note that we consider P_{22} to show that even if we apply the same assumptions as are given in [74], the results on P_2 improves by utilizing IL2. To propose our results, we first define a new online strategy.

Path Labeling Strategy (PLS)

Consider an undirected O-D edge-disjoint graph with total number of N different paths ($N > k$) from O to D. The K shortest paths between two nodes in an O-D edge-disjoint graph can be found in polynomial time in K by one of the known K-shortest paths algorithms ([69], [28]). Note that in O-D edge-disjoint graphs K is at most the number of edges. Here, we also note that k and K denote different numbers and they are not necessarily related. Applying the K-shortest paths algorithm, RS-type agents sort the paths in non-decreasing order, when the problem is at its initial stage. Now, they communicate and the ith RS-type agent assigns $(((L_1+1)*j)+i)th$ ($j = 0, 1, 2, \ldots$) shortest paths to his/her travel schedule. Hence the $((L_1+1)*j)th$ ($j = 1, 2, 3, \ldots$) shortest paths remain unselected. If the length of n number of paths is equal, RS-type agents select $\lfloor L_1(\frac{n}{L_1+1})\rfloor$ or $\lfloor L_1(\frac{n}{L_1+1})\rfloor + 1$ of them with respect to the number of paths which are already selected. Then, RS-type agents share their travel schedules to the R-type agent. Since the R-type agent benefits from IL2, he/she will take the shortest of the unselected paths. At each iteration, the agents will reach the destination or will find a blocked edge and return to the origin node. Then, they take the shortest unvisited path on their travel schedules.

Lemma 3.1.1. *P_{22} (L, L_1), equals to P_1 with $L_1 + 1$ agents with complete communication and at most k blockages in O-D edge-disjoint graphs.*

Proof. We consider O-D edge-disjoint graphs in two cases.

Case 1. There exists no RS-type agent in the problem. In this case, the R-type agent will find a new blockage at each iteration or will reach the destination. This implies that the problem equals to P_1 with one agent and at most k blockages.

Case 2. There exists at least one RS-type agent. In this case, applying PLS, the agents will find $L_1 + 1$ blockages at each iteration or at least one of them will reach the destination. This implies that the problem equals to P_1 with $L_1 + 1$ agents with complete communication and at most k blockages.

□

Corollary 3.1.1. *Considering levels 2 and 3 of agents' intelligence, P_2 (L, L_1) does not necessarily equal to P_1 with L_1 agents with complete communication and at most $k - 1$ blockages for all type of the graphs.*

Proof. Since we proved in Lemma 3.1.1 that P_2 (L, L_1) with IL2 agents equals to P_1 with $L_1 + 1$ agents with complete communication and at most k blockages on O-D edge-disjoint graphs, the corollary follows. □

Theorem 3.1.1. *For $P_{22}(L, L_1)$, there is no deterministic online strategy with competitive ratio less than $2\lfloor \frac{k}{L_1+1} \rfloor + 1$.*

Proof. As in [74], we replace the parameters of our converted problem in the lower bound of problem P_1, i.e. $2\lfloor \frac{k}{L} \rfloor + 1$. Since the special graph which is analyzed in [74] to provide the lower bound of the problem is an O-D edge-disjoint graph, the proof follows. Note that since our lower bound on P_2 is either strictly smaller than or equal to the lower bound in [74], it is necessary to update the lower bound of the problem when the agents benefit from the higher levels of intelligence, i.e. IL2 and IL3. □

3.1.5 The online multi-agent O-D k-CTP with IL3 agents (P_{23})

In this section, we provide improved results on P_2 by analyzing it with IL3 agents on O-D edge-disjoint graphs. We assume that there exists at least one RS-type agent in the graph. Since the agents benefit from the highest level of intelligence, we also ignore the second assumption of Zhang et al. in this section. We apply the RS-type agents ability to plan the travel schedules of the R-type agents (CP3) to utilize the R-type agents in an efficient way. Note that having CP3 is a reasonable assumption in most applications of the online multi-agent k-CTP. For example, in disaster relief the R-type agents can receive their travel plans from the RS-type agents in order to avoid retaking the selected paths by other agents. In the rest of this section, we analyze P_{23} by applying a modification of PLS that we call the modified PLS.

Modified Path Labeling Strategy (modified PLS)

Consider an undirected O-D edge-disjoint graph with total number of N different paths ($N > k$) from O to D. When the problem is at its initial stage the RS-type agent applies the K-shortest paths algorithm and labels the paths in non-decreasing order where path 1 is the shortest. If the length of some of the paths is equal, the RS-type agent selects their order arbitrarily. Now, he/she assigns paths $(L * j) + i$ ($j = 0, 1, 2, \ldots$) to the travel schedule of the ith agent. Then, he/she informs the travel schedule of each agent to them. At each iteration of this strategy, the L agents will take the shortest paths among the unvisited paths in their travel schedules. They will reach the destination or will find a blocked edge and return to the origin node. Then, they take the shortest unvisited path on their travel schedules. The cost of each iteration includes the cost of sending the agents through the assigned path plus the cost of returning to the origin node, if they face a blocked edge.

Lemma 3.1.2. *P_{23} (L, L_1) equals to P_1 with L agents with complete communication and at most k blockages in O-D edge-disjoint graphs.*

Proof. We consider O-D edge-disjoint graphs where there exists at least one RS-type agent in the graph. In this case, applying the modified PLS, the agents will find L blockages at each iteration or at least one of them will reach the destination. This implies that the problem equals to P_1 with L agents with complete communication and at most k blockages.

□

Theorem 3.1.2. *For $P_{23}(L, L_1)$, there is no deterministic online strategy with competitive ratio less than $2\lfloor \frac{k}{L} \rfloor + 1$.*

Proof. As in [74], we replace the parameters of our converted problem in the lower bound of problem P_1, i.e. $2\lfloor \frac{k}{L} \rfloor + 1$. Since the special graph which is analyzed in [74] to provide the lower bound of the problem is an O-D edge-disjoint graph, the proof follows. Note that since our lower bound on P_2 is either strictly smaller than or equal to the lower bound in [74], it is necessary to update the lower bound of the problem when the agents benefit from the highest level of intelligence, i.e. IL3. □

3.1.6 Optimal results for P_{23} on O-D edge-disjoint graphs

Proposition 3.1.1. *The competitive ratio of modified PLS in O-D edge-disjoint graphs is $(2\lfloor k/L \rfloor) + 1$, for both of the problems P_1 and P_{23}.*

Proof. Let C_O denote the cost of the shortest path after removing blocked edges. Since the RS-type agent has ordered the paths in non-decreasing order of cost, the cost of each iteration will be at most $2{*}C_O$. When an agent returns to the origin node, he/she will take the minimum cost unvisited path on his/her travel schedule. It takes at most $\lfloor \frac{k}{L} \rfloor + 1$ iterations to identify all of the blocked edges in the graph. Since the agents have to come back to the origin node after visiting a blocked edge in at most $\lfloor \frac{k}{L} \rfloor$ of the iterations, the cost of identifying blocked edges and returning to the origin node is at most $2\lfloor \frac{k}{L} \rfloor {*} C_O$. It will take a cost of C_O to find the remaining blockages and reach the destination. Thus, the total cost of the strategy is at most $(2\lfloor \frac{k}{L} \rfloor + 1) * C_O$. Since the strategy does not use complete communication, the competitive ratio is also valid for P_1. □

The obtained competitive ratio of modified PLS in O-D edge-disjoint graphs is optimal since it meets the lower bound offered in [74] for P_1.

Corollary 3.1.2. *Modified PLS is optimal for both problems P_1 and P_{23} in O-D edge-disjoint graphs.*

Proof. There is no deterministic online strategy with competitive ratio less than $2\lfloor \frac{k}{L} \rfloor + 1$ for P_1 in O-D edge-disjoint graphs. Note that, the lower bound of P_{23} is less than or equal to the lower bound of P_1. Since we have introduced a strategy for both P_1 and P_{23} which meets the lower bound of P_1, we conclude that the lower bound of P_{23} is equal to the lower bound of P_1. Hence, the strategy meets the lower bounds of both P_1 and P_{23}. □

Corollary 3.1.3. *Enabling all of the agents to communicate does not improve the competitive ratio in O-D edge-disjoint graphs in problem P_{23}.*

Proof. Since we have introduced a strategy (modified PLS) that does not need complete communication which meets the lower bound of P_1, the corollary follows. □

Corollary 3.1.4. *There are instances with O-D edge-disjoint graphs in which the competitive ratio of deterministic strategies on P_{23} improves by increasing the number of R-type agents.*

3.1.7 Concluding remarks

We analyzed the online multi-agent O-D k-Canadian Traveler Problem. We provided updated results including the lower bounds on the competitive ratio of deterministic strategies of the problem for the case where the communication is limited. We argued that it is vital to consider and utilize the higher levels of agents' intelligence in online problems by defining three levels of agents' intelligence. We introduced an online strategy in O-D edge-disjoint graphs which is optimal in both cases with complete and limited communication when the travel schedules are shared at the initial stage of the problem. We showed that enabling all of the agents to communicate does not improve the competitive ratio in O-D edge-disjoint graphs. Furthermore, we showed that there are instances with O-D edge-disjoint graphs in which the competitive ratio of deterministic strategies on P_2 improves by increasing the number of R-type agents.

3.2 Analysis of randomized online strategies

3.2.1 Introduction

We study the multi-agent k-Canadian Traveler Problem (k-CTP) to investigate the competitive performance of the randomized online strategies under different levels of communication between the traveling agents. The multi-agent k-CTP is a generalization of the single-agent k-CTP, for which randomized strategies have been investigated in the literature [64], [4]. The k-CTP originated from the Canadian Traveler Problem (CTP), which is an online navigation problem under incomplete information. In CTP, a traveling agent receives a graph with a given source node O and a destination node D, together with non-negative edge costs as input. The agent is located at O initially. There are some blocked edges in the graph, but these edges are not known to the agent. The agent discovers the status of an edge when he reaches an end-node of the edge. The objective of the agent is to provide an online strategy such that the agent finds a feasible path, i.e. one without blocked edges from O to D with minimum total cost of the edges taken by the agent. When an upper bound k on the number of blocked edges is given as input, the problem is called the k-CTP.

The multi-agent k-CTP is an online optimization problem that generalizes the k-CTP by the existence of multiple agents. In the multi-agent k-CTP, there are L agents in the graph who are initially located at O. The objective of the agents is to provide an online strategy such that at least one of them finds a feasible path, from O to D with minimum total cost of the edges taken by the agent that finds a feasible path first. Two versions of the multi-agent k-CTP have been introduced in the literature, with complete and limited communication [74]. When the communication is limited, some agents can both send and receive information and some of the agents can only receive information. The agents who are able to both send and receive information are called RS-type agents and the agents who are only able to receive information are called R-type agents. In presence of complete communication all of the agents can send and receive information, i.e. all of the agents are RS-type.

To evaluate the performance of online strategies, competitive ratio has been introduced by Sleator and Tarjan [55]. For a deterministic strategy, the competitive ratio is the maximum ratio of the cost of the online strategy to the cost of the offline strategy over all instances of the problem. For a randomized strategy, the competitive ratio is the maximum ratio of the expected cost of the online strategy to the cost of the offline strategy over all instances of the problem. In the offline k-CTP, the blocked edges are removed from the graph. Hence, it reduces to a cheapest (shortest) path problem.

3.2.2 Our Contributions

So far, only deterministic strategies have been studied for the multi-agent k-CTP. The competitiveness lower bounds have been provided for both of the cases with limited and complete communication. An optimal deterministic strategy has been introduced for both cases on O-D edge-disjoint graphs. However, the research on randomized strategies on the k-CTP is restricted to the single-agent version of the problem, where an optimal randomized strategy is introduced for only O-D edge-disjoint graphs. Note that an O-D edge-disjoint graph is an undirected graph G with a given source node O and a destination node D, such that any two distinct O-D paths in G are edge-disjoint; that is, they do not have a common edge. Here we note that the problem on O-D edge-disjoint graphs entails real-life applications. Because of the overlap among different paths on a general network, it is difficult to design a good strategy including multiple paths without overlaps for all the travelers [74]. However, most urban city traffic networks are highly connected and there usually exist several paths without overlap from O to D which are not much costlier than the cheapest O-D path in the graph [74].

In this section, we focus on randomized strategies for the multi-agent k-CTP. We analyze the problem in three cases: 1) without communication, 2) with limited communication and 3) with complete communication. We derive lower bounds on the competitive ratio of the randomized strategies for all of these cases. For the case without communication, we introduce a simple randomized strategy and prove its competitive ratio on a special case. By this way, we prove that increasing the number of agents can improve the competitive ratio of the randomized strategies for the multi-agent k-CTP.

For the cases with limited and complete communication, we introduce an optimal randomized strategy for both cases on O-D edge-disjoint graphs. Here we note that most optimal strategies in the literature are confined to O-D edge-disjoint graphs. Because our optimal strategy achieves a better expected competitive ratio in comparison to the optimal deterministic strategy on O-D edge-disjoint graphs, we conclude that randomization can improve the expected competitive performance of the online strategies for the k-CTP in presence of multiple agents and communication. We also prove that the competitive ratio of the optimal randomized strategy does not improve on O-D edge disjoint graphs, when the case with complete communication is compared to the case with limited communication.

3.2.3 Preliminaries

As standard assumptions for the multi-agent k-CTP, we assume: 1) the graph remains connected if any set of k edges are removed from the graph, and 2) the number of agents L is less than or equal to the number of blocked edges k. Before we explain our results, we need to present the property below that is defined by Bender and Westphal [4] to design a randomized strategy for the k-CTP.

Definition 3.2.1. The paths $P_1, P_2, ..., P_{k+1}$ with costs $c_1 \leq c_2 \leq ... \leq c_{k+1}$ have the *similar costs property* if for all $i = 1, 2, ..., k+1$ it holds that

$$c_i \leq \frac{2}{k+1} \sum_{j=1}^{k+1} c_j.$$

We also use the following two lemmas in our results which were proven in [4].

Lemma 3.2.1. *Suppose that the O-D paths $P_1, P_2, ..., P_{k+1}$ with costs $c_1 \leq c_2 \leq ... \leq c_{k+1}$ satisfy the similar costs property. In this case the probability distribution $\Omega_{k+1} = \lambda^* p'$ belongs to the polyhedron Q_k, where Ω_{k+1} and p' are $(k+1)$-vectors, $\lambda^* = \sum_{i=1}^{k+1} \frac{1}{p'_i} \in [0,1]$,*

$$p'_i = \frac{(1-k)c_i + \sum_{j=1 j\neq i}^{k+1} 2c_j}{(k+1)^2 c_i} \qquad \forall i = 1, 2, ..., k+1$$

and

$$Q_k = \{p \in \mathbb{R}_+^{k+1} : (1-k)p_i + \sum_{j=1 j\neq i}^{k+1} 2\frac{c_j}{c_i} p_j \leq 1 \;\; \forall i = 1, 2, ..., k+1, \sum_{i=1}^{k+1} p_i = 1\}.$$

Lemma 3.2.2. *Let P_1 and P_2 with costs $c_1 \leq c_2$ be the two cheapest paths in the graph. The strategy that chooses P_1 with probability $\frac{c_2^2}{c_1^2+c_2^2}$ and P_1 with probability $\frac{c_1^2}{c_1^2+c_2^2}$ is 2-competitive.*

3.2.4 Optimal randomized online strategy for the single-agent k-CTP

In this section we use Definition 5.5.1 and Lemma 5.5.1 to present a new randomized strategy which is optimal when L is one. We call this strategy S_1. Note that we apply S_1 later on to design our strategies for the multi-agent case in different levels of communication.

A new optimal strategy for the single-agent case (S_1)

- Initialization. Take an O-D edge-disjoint graph and k as input. Define S as the selection list and let $S = \emptyset$, initially. Let i be a counter variable and set $i = 0$, at the beginning. For any arbitrary set of n ($n \geq 3$) O-D paths, let the probability distribution $\Omega_n = (p_1, p_2, ..., p_n) \in Q_{n-1}$ be the probability distribution that is defined in Lemma 5.5.1. For any arbitrary set of two O-D paths P_1 and P_2 with costs $c_1 \leq c_2$, let $\Omega_2 = (\frac{c_2^2}{c_1^2+c_2^2}, \frac{c_1^2}{c_1^2+c_2^2}) \in Q_1$.

- Step 1. Remove all of the O-D paths in S to make it empty. Add the O-D paths to S by going through them in non-decreasing order of their costs until one of the following two conditions happen: 1) adding the O-D paths violates the similar costs property, or 2) the number of O-D paths in S exceeds $k+1-i$. Go to step 2.

- Step 2. Let n_S denote the number of O-D paths in S. Take one of the O-D paths in S according to the probability distribution Ω_{n_S}. If the agent has arrived at D, stop. Otherwise, a new blockage is identified and the agent backtracks to O. Remove the taken O-D path from the graph and set $i = i+1$. Then, go to Step 1.

To prove the optimality of S_1, we need to present the following three lemmas, where we consider the O-D paths $P_1, P_2, ..., P_{k+1}$ with costs $c_1 \leq c_2 \leq ... \leq c_{k+1}$ and the probability distribution $\Omega_{k+1} = (p_1, p_2, ..., p_{k+1}) \in Q_k$ (Lemma 5.5.1).

Lemma 3.2.3. *It holds that* $p_{k+1} \leq p_k \leq ... \leq p_2 \leq p_1$.

Proof. We show that for any arbitrary v and w ($v, w \in \{1, 2, ..., k+1\}$) such that $v < w$, $p_v \geq p_w$. We have

$$p_v = \lambda^* \frac{(1-k)c_v + \sum_{j=1 j\neq v}^{k+1} 2c_j}{(k+1)^2 c_v}$$

and

$$p_w = \lambda^* \frac{(1-k)c_w + \sum_{j=1 j\neq w}^{k+1} 2c_j}{(k+1)^2 c_w}.$$

Note that eliminating λ^* from the definitions of p_v and p_w has no effect on the comparison of their values since it is common in both p_v and p_w. Hence, to compare p_v and p_w we can consider the quantities

$$p_v^{'} = \frac{1-k}{(k+1)^2} + \frac{\sum_{j=1 j\neq v}^{k+1} 2c_j}{(k+1)^2 c_v}$$

and

$$p_w^{'} = \frac{1-k}{(k+1)^2} + \frac{\sum_{j=1 j\neq w}^{k+1} 2c_j}{(k+1)^2 c_w}.$$

Again we can eliminate $\frac{1-k}{(k+1)^2}$ from $p_v^{'}$ and $p_w^{'}$ since it is common in both of them. Then, we can consider the quantities

$$p_v^{''} = \frac{\sum_{j=1 j\neq v}^{k+1} 2c_j}{(k+1)^2 c_v}$$

and

$$p_w^{''} = \frac{\sum_{j=1 j\neq w}^{k+1} 2c_j}{(k+1)^2 c_w},$$

to compare the values of p_v and p_w. Note that $c_v \leq c_w$, thus the numerator of $p_v^{''}$ is greater than or equal to the numerator of $p_w^{''}$ and the denominator of $p_v^{''}$ is smaller than or equal to the denominator of $p_w^{''}$. We get $p_w^{''} \leq p_v^{''}$, which implies that $p_w \leq p_v$. □

Lemma 3.2.4. *Consider the vector* $\Pi_{k+1} = (\pi_1, \pi_2, ..., \pi_{k+1})$ *such that* $\pi_i = \frac{1}{k+1}$ *for* $i = 1, 2, ..., k+1$. *It holds that* $\sum_{i=1}^{k+1} p_i c_i \leq \sum_{i=1}^{k+1} \pi_i c_i$.

Proof. Note that $\sum_{i=1}^{k+1} p_i = 1$ since $\Omega_{k+1} \in Q_k$ according to Lemma 5.5.1. Also note that $\sum_{i=1}^{k+1} \pi_i = \sum_{i=1}^{k+1} \frac{1}{k+1} = 1$. We need to show that

$$0 \leq \sum_{i=1}^{k+1} (\pi_i - p_i) c_i.$$

If $0 \leq \pi_i - p_i$ for all $i = 1, 2, ..., k+1$ the lemma follows. Suppose that $\pi_i - p_i \leq 0$ for $i = 1, 2, ..., j$ $(j < k+1)$ and $0 \leq \pi_i - p_i$ for $i = j+1, ..., k, k+1$. In this case it holds that

$$\sum_{i=1}^{j} (\pi_i - p_i) = -\sum_{i=j+1}^{k+1} (\pi_i - p_i),$$

since $\sum_{i=1}^{k+1} \pi_i - \sum_{i=1}^{k+1} p_i = 0$. Note that $\sum_{i=1}^{j} (\pi_i - p_i) c_i$ is not less than $c_j \sum_{i=1}^{j} (\pi_i - p_i)$ since we assumed $\pi_i - p_i \leq 0$ for $i = 1, 2, ..., j$, and we have $c_1 \leq c_2 \leq ... \leq c_j$. Also note that $\sum_{i=j+1}^{k+1} (\pi_i - p_i) c_i$ is not less than $c_{j+1} \sum_{i=j+1}^{k+1} (\pi_i - p_i)$ since we assumed $0 \leq \pi_i - p_i$ for $i = j+1, ..., k, k+1$, and we have $c_{j+1} \leq c_{j+2} \leq ... \leq c_{k+1}$. Hence

$\sum_{i=1}^{k+1}(\pi_i - p_i)c_i$ can be bounded from below by

$$c_j(\sum_{i=1}^{j} \pi_i - p_i) + c_{j+1}(\sum_{i=j+1}^{k+1} \pi_i - p_i).$$

Because $\sum_{i=1}^{j}(\pi_i - p_i) = -\sum_{i=j+1}^{k+1}(\pi_i - p_i)$, we can re-write the expression above as

$$(c_{j+1} - c_j)\sum_{i=j+1}^{k+1}(\pi_i - p_i).$$

Note that $\sum_{i=j+1}^{k+1}(\pi_i - p_i)$ is positive since we assumed that $0 \leq \pi_i - p_i$ for $i = j+1, ..., k, k+1$. Also we have $c_j \leq c_{j+1}$. Thus we obtain

$$0 \leq (c_{j+1} - c_j)\sum_{i=j+1}^{k+1}(\pi_i - p_i) \leq \sum_{i=1}^{k+1}(\pi_i - p_i)c_i.$$

$\square$

Lemma 3.2.5. *Consider an arbitrary O-D path P_{k+2} with cost $c_{k+1} < c_{k+2}$ such that P_{k+2} does not fulfill the similar costs property with $P_1, P_2, ..., P_{k+1}$. It holds that $\frac{2\sum_{i=1}^{k+1} p_i c_i}{c_{k+2}} \leq 1$.*

Proof. Since P_{k+2} does not satisfy the similar costs property with the set of O-D paths $P_1, P_2, ..., P_{k+1}$ it follows from Definition 5.5.1 that

$$\frac{2}{k+2}\sum_{i=1}^{k+2} c_i < c_{k+2}.$$

We first multiply both sides by $k+2$ and then eliminate $2c_{k+2}$ from both sides to obtain

$$\frac{2\sum_{i=1}^{k+1} c_i}{k+1} < \frac{2\sum_{i=1}^{k+1} c_i}{k} < c_{k+2}.$$

Note that the left-hand side is greater than or equal to $2\sum_{i=1}^{k+1} p_i c_i$ according to Lemma 5.5.2. Thus

$$2\sum_{i=1}^{k+1} p_i c_i < c_{k+2}$$

and the lemma follows. □

Now we can prove that S_1 is $k+1$ competitive and optimal on arbitrary O-D edge-disjoint graphs.

Theorem 3.2.1. *The competitive ratio of S_1 is $k+1$ on O-D edge-disjoint graphs.*

Proof. Our proof is by induction.

- **Base case.** Suppose that P_1 and P_2 with costs $c_1 \leq c_2$ are the two cheapest O-D paths in the graph. When $k=1$, the agent takes one of P_1 and P_2 with probability distribution $\Omega_2 = (\frac{c_2^2}{c_1^2+c_2^2}, \frac{c_1^2}{c_1^2+c_2^2})$. If the taken O-D path is blocked, the agent backtracks to O and takes the cheapest O-D path in the graph to arrive at D. Otherwise, the agent arrives at D. We need to show that S_1 achieves a competitive ratio of two in this case. This is exactly what Lemma 3.2.2 states.

- **Induction.** Let P_{i^*} ($i^* \in \{1,2,...,k+1\}$ with cost c_{i^*} be the offline optimum. At the first implementation of Step 1, the O-D paths are added to the selection list in non-decreasing order of their costs until adding the O-D paths to the selection list violates the similar costs property or the number of O-D paths in the selection list exceeds $k+1$. Let the O-D paths $P_1, P_2, ..., P_n$ with costs $c_1, c_2, ..., c_n$ be the O-D paths that are added to the selection list after the first implementation of Step 1. Then the strategy enters Step 2. We present the rest of our proof by considering two cases.

  - Case 1. P_{i^*} is not added to the selection list. In this case the agent takes one of the O-D paths in the selection list according to the probability distribution $\Omega_n = (p_1, p_2, ..., p_n)$. Then he arrives at the end-node of a blockage and backtracks to O. Hence, the expected cost of $2\sum_{i=1}^{n} p_i c_i$ is incurred. Note that $2\sum_{i=1}^{n} p_i c_i \leq 2\sum_{i=1}^{n} \frac{1}{n} c_i$ according to Lemma 5.5.2. Let C^{k-1} denote the cost of S_1 from the time when he arrives back at O until the end of the strategy. Note that C^{k-1} is at most kc_{i^*} by the induction assumption. Thus the expected cost of S_1 is at most $2\sum_{i=1}^{n} \frac{1}{n} c_i + kc_{i^*}$.

Since P_{i^*} is not in the selection list, it follows that P_{i^*} does not fulfill the similar costs property with the set of O-D paths in the selection list and $c_n < c_{i^*}$. Hence $2\sum_{i=1}^{n}\frac{1}{n}c_i < c_{i^*}$ according to Lemma 3.2.5. Thus, the expected cost of S_1 is at most $(k+1)c_{i^*}$. It follows that the competitive ratio is at most $k+1$ in this case.

– Case 2. P_{i^*} is added to the selection list. Let B and T denote the set of blocked and traversable O-D paths in the selection list, respectively. In this case the agent takes one of the O-D paths in the selection list according to the probability distribution $\Omega_n = (p_1, p_2, ..., p_n)$. If the taken O-D path belongs to B, he backtracks to O and discards the traversed O-D path from the graph; otherwise, the agent proceeds with the chosen O-D path to arrive at D, i.e. the strategy ends. Suppose that the taken O-D path belongs to B and let C^{k-1} denote the cost of S_1 from when the agent arrives back at O until the end of the strategy in this case. The competitive ratio can be written as

$$\sum_{i\in T}\frac{p_i c_i}{c_{i^*}} + \sum_{j\in B} p_j \frac{2c_j + C^{k-1}}{c_{i^*}}.$$

Note that C^{k-1} is at most kc_{i^*} by the induction assumption. Hence, the competitive ratio is bounded from above by

$$\sum_{i\in T}\frac{p_i c_i}{c_{i^*}} + \sum_{j\in B} p_j(\frac{2c_j}{c_{i^*}} + k) \leq p_{i^*} + \sum_{j=1 j\neq i^*}^{k+1} p_j(\frac{2c_j}{c_{i^*}} + k).$$

We claim that the right-hand side is at most $k+1$ for all $i^* = 1, 2, ..., k+1$, i.e.

$$(1-k)p_i + \sum_{j=1 j\neq i}^{k+1} p_j \frac{2c_j}{c_i} \leq 1$$

for all $i = 1, 2, ..., k+1$. Since the probability distribution $\Omega_{k+1} = (p_1, p_2, ..., p_{k+1})$ belongs to the polyhedron Q_k, the claim follows by the definition of Q_k in Lemma 5.5.1.

The theorem follows. □

3.2.5 Competitive analysis of the randomized online strategies on the multi-agent k-CTP without communication

In this section we analyze the randomized strategies on the multi-agent k-CTP in the absence of communication. First, we derive a lower bound to this problem. Next, we present a simple randomized strategy for when multiple agents are in the graph and communication is not possible. We call this strategy S_2 and prove its competitive ratio for a special case. By this way, we show that increasing the number of agents can improve the competitive ratio of the randomized strategies on the multi-agent k-CTP without communication.

The lower bound

Yao [68] showed that the expected cost of a randomized strategy on the worst-case input is no better than a worst-case random probability distribution of the deterministic strategy which performs the best for that distribution. This principle is known as *Yao's Principle* and was applied to prove the lower bound of the problem in the existence of only one agent in the graph in [64]. In this section, we use Yao's Principle to derive lower bounds for the multi-agent versions of the problem in different levels of communication. Below we present a lower bound on the competitive ratio of the randomized strategies for the multi-agent k-CTP without communication.

Theorem 3.2.2. *There is no randomized strategy with competitive ratio less than*

$$\sum_{j=1}^{k+1}(1-(\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1),$$

for the multi-agent k-CTP in absence of communication.

Proof. We consider the O-D edge-disjoint graph in Figure 3.1. Note that we assume that ϵ is sufficiently small such that we do not consider ϵ values in our analysis.

We select $i \in \{1, 2, ..., k+1\}$ uniformly at random, block all edges (V_j, D) with $j \neq i$ and let the edge (V_i, D) remain traversable. Thus $O - V_i - D$ becomes the only traversable O-D path whose cost is one, i.e. the cost of the offline optimum is one. To complete the proof, we need to show that the expected cost of an arbitrary deterministic strategy with respect to the distribution given on the inputs is not greater than $\sum_{j=1}^{k+1}(1 - (\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1)$. We organize the rest of our proof in two parts.

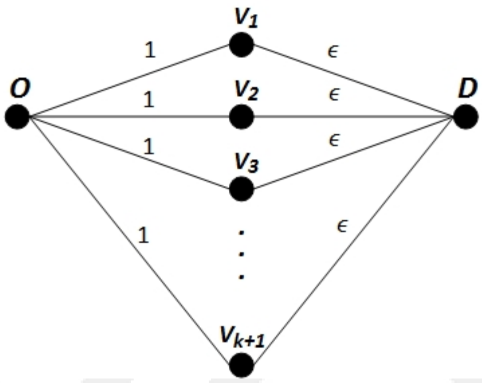

Figure 3.1: A special graph

- **The concept of iteration.** Note that there is no communication between agents. Hence the agents do not benefit from waiting at O in order to receive blockage information. Moreover, each arbitrary agent only knows the O-D paths that he has taken and has no information about the O-D paths that are taken by the other agents. Note that all of the agents are initially at O. Thus we can define the concept of iteration for an arbitrary deterministic strategy for the multi-agent k-CTP in absence of communication on the special graph in Figure 3.1 as follows. At the beginning of each iteration, each arbitrary agent takes an O-D path from the set of O-D paths that he has not taken before and proceeds with it to arrive at D or an end-node of a blockage. Note that the taken O-D paths in this case are not necessarily different. If one of the agents arrives at D, then the strategy ends and the cost of the iteration equals to one. If all of the agents arrive at an end-node of a blockage, all of the agents retrace back to O and the cost of the iteration equals to two. Also note that an arbitrary deterministic strategy ends within $k+1$ iterations in this case.

- **Computation of the competitive ratio.** Note that the probability that $O-V_i-D$ is chosen at iteration j^* ($j^*=1,2,...,k+1$) in an arbitrary deterministic strategy can be computed as

$$(1-(\frac{k-(j^*-1)}{k+1-(j^*-1)})^L)\prod_{j=1}^{j^*-1}(\frac{k-(j-1)}{k+1-(j-1)})^L,$$

which is equal to

$$(1-(\frac{k-(j^*-1)}{k+1-(j^*-1)})^L)(\frac{k-(j^*-2)}{k+1})^L$$

for $j^*=1,2,...,k+1$. Note that if $O-V_i-D$ is selected at the jth iteration, the strategy ends with cost of $2j-1$. Hence the expected cost can be computed as

$$\sum_{j=1}^{k+1}(1-(\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1).$$

This implies that the expected cost of an arbitrary deterministic strategy with respect to the distribution given on the inputs, does not exceed $\sum_{j=1}^{k+1}(1-(\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1)$. Since the cost of offline optimum is one, the theorem follows from Yao's Principle. □

Now we introduce our randomized strategy S_2 for the case without communication.

Randomized strategy for the multi-agent case without communication (S_2)

Take an O-D edge-disjoint graph, L and k as input. Let each agent apply S_1 on the input graph independently to the other agents.

Below we prove an upper bound on the competitive ratio of S_2 on arbitrary O-D edge-disjoint graphs.

Lemma 3.2.6. *S_2 is at most $k+1$ competitive on O-D edge-disjoint graphs.*

Proof. Let P^* be the offline optimum and c^* be its cost. Note that the expected cost of S_2 corresponds to the expected cost of the minimum of L independent strategies that their expected costs are at most $(k+1)c^*$. Hence the expected cost of S_2 is at most $(k+1)c^*$ and the lemma follows. □

By proposing the following two lemmas, we investigate whether increasing the number of agents can improve the competitive ratio of the randomized strategies for the multi-agent k-CTP when there is no communication between agents. First, we prove the competitive ratio of S_2 on the special graph in Figure 3.1.

Lemma 3.2.7. *S_2 is $\sum_{j=1}^{k+1}(1-(\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1)$ competitive on the special graph in Figure 3.1.*

Proof. Note that the agents do not benefit from waiting at O to receive any blockage information from the other agents, based on the same reason that is presented in the proof of Theorem 3.2.2. Because of the symmetry in the special graph in Figure 3.1, we can define the concept of iteration for S_2 on this special graph as follows. At the beginning of each iteration, each arbitrary agent takes an O-D path from the set of O-D paths that he has not taken before uniformly at random. Then, the agent proceeds with the taken O-D path to arrive at D or an end-node of a blockage. Note that the taken O-D paths in this case are not necessarily different. If one of the agents arrives at D, then the strategy ends and the cost of the iteration equals to one. If all of the agents arrive at an end-node of a blockage, all of the agents retrace back to O and the cost of the iteration equals to two. Also note that S_2 ends within $k+1$ iterations in this case. Similar to the proof of Theorem 3.2.2, we can compute the probability that $O-V_i-D$ is taken at iteration j^* as

$$(1-(\frac{k-(j^*-1)}{k+1-(j^*-1)})^L)(\frac{k-(j^*-2)}{k+1})^L$$

for $j^* = 1, 2, ..., k+1$. Note that if $O - V_i - D$ is traversed at the jth iteration, the strategy ends with cost of $2j - 1$. Thus the expected cost can be computed as

$$\sum_{j=1}^{k+1}(1 - (\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1).$$

□

As a consequence, we prove that increasing the number of agents can improve the competitive ratio of the randomized strategies, when there is no communication between the agents.

Lemma 3.2.8. *There are instances in which increasing the number of agents improves the competitive ratio of the randomized strategies in absence of communication.*

Proof. We just showed in Lemma 3.2.7 that the competitive ratio of S_2 on the special graph in Figure 3.1 is

$$CR = \sum_{j=1}^{k+1}(1 - (\frac{k-(j-1)}{k+1-(j-1)})^L)(\frac{k-(j-2)}{k+1})^L(2j-1)$$

in absence of communication. If we show that CR is decreasing in L, the lemma follows. We set

$$\alpha = (\frac{k-(j-1)}{k+1-(j-1)})$$

and

$$\beta = (\frac{k-(j-2)}{k+1}).$$

Note that $\alpha \in (0, 1)$ and $\beta \in (0, 1]$, thus $\alpha\beta < \beta$. We define

$$CR' = (1 - \alpha^L)\beta^L = \beta^L - (\alpha\beta)^L.$$

Note that CR' is an item of CR for a given j and if it is shown that CR' is decreasing in L, it follows that CR is also decreasing in L. Hence if it is shown that CR' is decreasing in L, the lemma follows. When we assume that L is continuous and compute the first derivative of $CR^{'}$ with respect to L, we obtain that $CR^{'}$ is decreasing in L for $L > 1$. This is because the first derivative of $CR^{'}$ with respect to L is positive for $L > 1$. This implies that $CR^{'}$ is decreasing in L for integer values of L that are greater than one. □

3.2.6 *Competitive analysis of the randomized online strategies on the multi-agent k-CTP with communication*

In this section we analyze randomized strategies on the multi-agent k-CTP in presence of limited and complete communication. For the case with limited communication, we assume that L_1 $(1 \leq L_1 < L)$ agents are RS-type and $(L - L_1)$ agents are R-type. We remind that the RS-type agents are able to both send and receive information while the R-type agents can only receive information. To specify what information can be transmitted between the agents, we assume that the agents benefit from intelligence level 3 (IL3) defined in [51]. The RS-type agents with IL3 have the following attributes. 1) They can transmit their own travel plans to the other agents, i.e. they can transmit the O-D paths that they plan to traverse to the other agents. 2) They can plan for the R-type agents, i.e. they can determine for the R-type agents which O-D paths to traverse. 3) They can share blockage information with the other agents when they learn a new blockage. We note that the detailed description of IL3 can be found in [51]. We suggest a lower bound for both of the cases with limited and complete communication. We also present an optimal strategy for both cases on O-D edge-disjoint graphs.

The lower bound

We again apply Yao's Principle to derive our lower bound for the case with complete communication.

Theorem 3.2.3. *There is no randomized strategy with competitive ratio less than* $\frac{L}{k+1}(\lfloor\frac{k}{L}\rfloor)^2 + \frac{k+1-L\lfloor\frac{k}{L}\rfloor}{k+1}(2\lfloor\frac{k}{L}\rfloor + 1)$, *for the multi-agent k-CTP in presence of complete communication.*

Proof. Similar to the proof of Theorem 3.2.2, we consider the O-D edge-disjoint graph in Figure 3.1. Note that we assume that ϵ is sufficiently small such that we do not consider ϵ values in our analysis. We select $i \in \{1, 2, ..., k+1\}$ uniformly at random, block all edges (V_j, D) with $j \neq i$ and let the edge (V_i, D) remain traversable. Thus $O - V_i - D$ becomes the only traversable O-D path whose cost is one, i.e. the cost of the offline optimum is one. To complete the proof, we need to show that the expected cost of an arbitrary deterministic strategy with respect to the distribution given on the inputs is not greater than $\frac{L}{k+1}(\lfloor \frac{k}{L} \rfloor)^2 + \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(2\lfloor \frac{k}{L} \rfloor + 1)$. We organize the rest of our proof in two parts.

- **The concept of iteration.** Note that the agents benefit from complete communication, i.e. all of the agents are RS-type. Also note that the graph in Figure 3.1 is O-D edge-disjoint and there is no benefit for the agents to take an O-D path which is already traversed. Thus in any deterministic strategy, when an arbitrary O-D path P is taken by one of the agents the other agents learn that P is taken and discard it from the graph. The agent who has taken P will proceed with it to arrive at D or at an end-node of a blockage, in the latter case he backtracks to O and discards P from the graph. In other words, the agents do not benefit from waiting at O in order to receive any blockage information and they immediately take a new O-D path when they retrace back to O. Note that all of the L agents are initially at O. Thus we can define the concept of iteration for an arbitrary deterministic strategy for the multi-agent k-CTP in presence of complete communication on the special graph in Figure 3.1 as follows. At the beginning of each iteration, L agents at O take L different O-D paths from the set of O-D paths that are not traversed. The iteration ends either when at least one of the agents arrive at D or all of the agents retrace back to O. The former case happens when the O-D path $O - V_i - D$ is selected by one of L agents, in this case the cost of the iteration equals to one and the strategy ends. The latter case occurs when all of the L O-D paths that are taken by L agents are blocked. In this case the cost of the iteration equals to two. Also note that any deterministic strategy ends within at most $\lfloor \frac{k}{L} \rfloor + 1$ iterations.

- **Computation of the competitive ratio.** Note that the probability that the O-D path $O - V_i - D$ is selected at iteration j^* in an arbitrary deterministic strategy can be computed as

$$\frac{(\prod_{j=1}^{j^*-1} \binom{k-(j-1)L}{L})\binom{1}{1}\binom{k-(j^*-1)L}{L-1}}{\prod_{j=1}^{j^*} \binom{k+1-(j-1)L}{L}}$$

for $j^* = 1, 2, ..., \lfloor \frac{k}{L} \rfloor + 1$. The value of the probability above is $\frac{L}{k+1}$ for $j^* = 1, 2, ..., \lfloor \frac{k}{L} \rfloor$ and equals to $\frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}$ for $j^* = \lfloor \frac{k}{L} \rfloor + 1$. If $O - V_i - D$ is selected at the jth iteration, the strategy ends with cost of $2j - 1$. Hence the expected cost of the strategy can be computed as

$$\frac{L}{k+1} \sum_{j=1}^{\lfloor \frac{k}{L} \rfloor} 2j - 1 + \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(2\lfloor \frac{k}{L} \rfloor + 1),$$

which equals to

$$\frac{L}{k+1}(\lfloor \frac{k}{L} \rfloor)^2 + \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(2\lfloor \frac{k}{L} \rfloor + 1).$$

We just showed that the expected cost of an arbitrary deterministic strategy with respect to the distribution given on the inputs does not exceed $\frac{L}{k+1}(\lfloor \frac{k}{L} \rfloor)^2 + \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(2\lfloor \frac{k}{L} \rfloor + 1)$. Since the cost of the offline optimum is one, the theorem follows from Yao's Principle.

□

Corollary 3.2.1. *There is no randomized online strategy with competitive ratio less than* $\frac{L}{k+1}(\lfloor \frac{k}{L} \rfloor)^2 + \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(2\lfloor \frac{k}{L} \rfloor + 1)$*, for multi-agent k-CTP in presence of limited communication.*

Proof. Note that the agents have lower level of communication in comparison to the case with complete communication. Hence the competitive ratio can not be better than the competitive ratio of the case with complete communication. □

The remark below shows that our proposed lower bound is bounded above by $\lfloor \frac{k}{L} \rfloor + 1$.

Remark 3.2.1. Note that $0 \leq k + 1 - L\lfloor \frac{k}{L} \rfloor \leq L$ and equals to L when $k + 1$ is a multiple of L. Thus the maximum value of the competitive ratio corresponds to the case when $k + 1$ is a multiple of L. Note that in this case $\lfloor \frac{k}{L} \rfloor + 1 = \frac{k+1}{L}$. Hence the competitive ratio can be bounded above by $\frac{L}{k+1} \sum_{i=1}^{\lfloor \frac{k}{L} \rfloor + 1} 2i - 1 = \frac{L}{k+1} \sum_{i=1}^{\frac{k+1}{L}} 2i - 1$, which equals to $\frac{L}{k+1}(\frac{k+1}{L})^2 = \frac{k+1}{L} = \lfloor \frac{k}{L} \rfloor + 1$. We just bounded the competitive ratio from above by the consideration of the case that maximizes the competitive ratio.

Now we introduce our optimal randomized strategy in presence of both limited and complete communication. We call our strategy S_3.

Optimal strategy for the cases with limited and complete communication (S_3)

Take an O-D edge-disjoint graph, L, L_1 and k as input. Select an arbitrary RS-type agent A. Let A determine $k+1$ cheapest O-D paths in the graph, i.e. $P_1, P_2, ..., P_{k+1}$ with costs $c_1 \leq c_2 \leq ... \leq c_{k+1}$. Then, A randomly classifies $P_1, P_2, ..., P_{k+1}$ into $k+1-L\lfloor \frac{k}{L} \rfloor$ groups which contain $\lfloor \frac{k}{L} \rfloor + 1$ O-D paths and $L - (k+1-L\lfloor \frac{k}{L} \rfloor)$ groups which have $\lfloor \frac{k}{L} \rfloor$ O-D paths such that: 1) P_i ($i \in \{1, 2, ..., k+1\}$) is added to one of the groups that have $\lfloor \frac{k}{L} \rfloor + 1$ O-D paths with probability $p_1 = \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(\lfloor \frac{k}{L} \rfloor + 1)$, and 2) P_i ($i \in \{1, 2, ..., k+1\}$) is added to one of the groups that have $\lfloor \frac{k}{L} \rfloor$ O-D paths with probability $p_2 = \frac{L-(k+1-L\lfloor \frac{k}{L} \rfloor)}{k+1}\lfloor \frac{k}{L} \rfloor$, i.e. $p_1 + p_2 = 1$. Note that L different groups are constructed. Next, A assigns exactly one agent to each group arbitrarily. In this case the problem is decomposed into L different sub-problems with only one agent. Apply S_1 on each sub-problem. If at least one of the agents has arrived at D, stop.

Note that S_1 and S_3 are equivalent when L is one. Below we prove the competitive ratio of S_3.

Theorem 3.2.4. *The competitive ratio of S_3 is at most* $\frac{(k+1-L\lfloor\frac{k}{L}\rfloor)}{k+1}(\lfloor\frac{k}{L}\rfloor+1)^2+\frac{(L-(k+1-L\lfloor\frac{k}{L}\rfloor))}{k+1}(\lfloor\frac{k}{L}\rfloor)^2$, *on O-D edge-disjoint graphs.*

Proof. Let P_{i^*} ($i^* \in \{1,2,...,k+1\}$) be the offline optimum and c^* be its cost. S_3 partitions the O-D paths into $k+1-L\lfloor\frac{k}{L}\rfloor$ groups that have $\lfloor\frac{k}{L}\rfloor+1$ O-D paths and $L-(k+1-L\lfloor\frac{k}{L}\rfloor)$ groups which contain $\lfloor\frac{k}{L}\rfloor$ O-D paths. Also, exactly one agent is assigned to each group and implements S_1 on that group. P_{i^*} belongs to one of the groups that have $\lfloor\frac{k}{L}\rfloor+1$ O-D paths with probability $\frac{(k+1-L\lfloor\frac{k}{L}\rfloor)(\lfloor\frac{k}{L}\rfloor+1)}{k+1}$ and belongs to one of the groups which contain $\lfloor\frac{k}{L}\rfloor$ O-D paths with probability $\frac{(L-(k+1-L\lfloor\frac{k}{L}\rfloor))\lfloor\frac{k}{L}\rfloor}{k+1}$. In the former case, there are at most $\lfloor\frac{k}{L}\rfloor$ blocked O-D paths in the group which contains P_{i^*}, hence S_3 ends with an expected cost of at most $(\lfloor\frac{k}{L}\rfloor+1)c^*$ by Theorem 5.5.1. In the latter case, there are at most $\lfloor\frac{k}{L}\rfloor-1$ blocked O-D paths in the group that contains P_{i^*}. Hence, S_3 ends with an expected cost of at most $(\lfloor\frac{k}{L}\rfloor)c^*$ by Theorem 5.5.1. Therefore, the expected cost of S_3 is at most

$$\frac{(k+1-L\lfloor\frac{k}{L}\rfloor)(\lfloor\frac{k}{L}\rfloor+1)}{k+1}((\lfloor\frac{k}{L}\rfloor+1)c^*)+\frac{(L-(k+1-L\lfloor\frac{k}{L}\rfloor))\lfloor\frac{k}{L}\rfloor}{k+1}((\lfloor\frac{k}{L}\rfloor)c^*).$$

The theorem follows since the cost of the offline optimum is c^*. □

Theorem 3.2.5. *S_3 meets the lower bound of* $\frac{L}{k+1}(\lfloor\frac{k}{L}\rfloor)^2+\frac{k+1-L\lfloor\frac{k}{L}\rfloor}{k+1}(2\lfloor\frac{k}{L}\rfloor+1)$ *on the competitive ratio of the randomized strategies for the multi-agent k-CTP in presence of both limited and complete communication.*

Proof. We need to show that the competitive ratio of S_3 and the lower bound are equal. Set $\alpha = k+1-L\lfloor\frac{k}{L}\rfloor$. In this case the competitive ratio of S_3 can be written as

$$\frac{L-\alpha}{k+1}\lfloor\frac{k}{L}\rfloor^2+\frac{\alpha}{k+1}(\lfloor\frac{k}{L}\rfloor+1)^2=\frac{L}{k+1}\lfloor\frac{k}{L}\rfloor^2+\frac{\alpha}{k+1}(2\lfloor\frac{k}{L}\rfloor+1),$$

and the lower bound can be written as

$$\frac{L}{k+1}\lfloor\frac{k}{L}\rfloor^2+\frac{\alpha}{k+1}(2\lfloor\frac{k}{L}\rfloor+1).$$

Since the presence of only one RS-type agent suffices to implement S_3, the theorem

Table 3.1: Comparison between the competitive ratio of the optimal deterministic strategy and the optimal randomized strategy on O-D edge-disjoint graphs in presence of limited and complete communication

	M-PLS	S_3
Case with limited communication	$2\lfloor \frac{k}{L} \rfloor + 1$	$\lfloor \frac{k}{L} \rfloor + 1$
Case with complete communication	$2\lfloor \frac{k}{L} \rfloor + 1$	$\lfloor \frac{k}{L} \rfloor + 1$

follows for both cases of limited and complete communication. □

Remark 3.2.2. Note that $\frac{L}{k+1}(\lfloor \frac{k}{L} \rfloor)^2 + \frac{k+1-L\lfloor \frac{k}{L} \rfloor}{k+1}(2\lfloor \frac{k}{L} \rfloor + 1) \leq \lfloor \frac{k}{L} \rfloor + 1$ according to Remark 3.2.1. It follows that S_3 is at most $\lfloor \frac{k}{L} \rfloor + 1$ competitive on O-D edge-disjoint graphs in both cases with limited and complete communication.

Table 3.1 presents a comparison between the competitive ratio of our optimal randomized strategy S_3 and the optimal deterministic strategy (M-PLS) that is presented in [51] for O-D edge-disjoint graphs. As we can observe, S_3 achieves a better expected competitive performance (almost twice better) in both cases of limited and complete communication in comparison to the M-PLS.

3.2.7 Concluding remarks

We studied randomized online strategies for the multi-agent k-CTP. We analyzed the problem in three different cases: 1) without communication, 2) with limited communication, and 3) with complete communication. We proved lower bounds on the competitive ratio of the randomized online strategies for these cases. We introduced an optimal randomized strategy for the cases with limited and complete communication on O-D edge-disjoint graphs which finds real-life applications. We showed that our optimal randomized strategy S_3 achieves a better expected competitive performance in comparison to the optimal deterministic strategy (M-PLS) that is given in the literature. We also showed that having complete communication does not improve the competitive ratio of the optimal randomized strategy on O-D edge-disjoint graphs in comparison to the case when communication is limited. Additionally, we showed that increasing the number of agents can improve the competitive ratio of the randomized strategies when there is no communication between agents. We also note that the problem of designing a randomized online strategy in the case without communication that meets the lower bound of the problem on O-D edge-disjoint graphs remains as an open research problem.

3.3 Analysis of online strategies on graphs having common edges on the O-D paths

3.3.1 Introduction

In recent years, several studies have focused on designing and analyzing online strategies for the multi-agent k-CTP in presence of complete communication (e.g. see [51] and [74]). We remind that in the multi-agent k-CTP, k denotes the number of blocked edges and L denotes the number of agents. A lower bound of $2\lfloor \frac{k}{L} \rfloor + 1$ on the competitive ratio of deterministic strategies has been derived in [74]. However, an optimal deterministic strategy which meets the given lower bound is not provided.

3.3.2 Our contributions

In this section, we prove a lower bound on the competitive ratio of deterministic online strategies for the multi-agent k-CTP with complete communication which is tighter than the lower bound of $2\lfloor \frac{k}{L} \rfloor + 1$ given in the literature. By this way, we show that no deterministic strategy matches the lower bound of $2\lfloor \frac{k}{L} \rfloor + 1$ on the competitive ratio of deterministic strategies for the multi-agent k-CTP in presence of complete communication.

3.3.3 The lower bound

A lower bound on the competitive ratio is usually derived by designing a set of specific instances on which no online strategy can perform well compared to an optimal offline strategy. For the multi-agent k-CTP with complete communication, the input graph instances that are considered for providing the current best lower bound ($2\lfloor\frac{k}{L}\rfloor + 1$) are such that the O-D paths do not contain common edges. However, common edges on the O-D paths may cause the agents that are assigned to different O-D paths to encounter the same blocked edge. We note that since all of the agents can communicate with each other, the competitive performance of deterministic strategies may reduce in cases in which the O-D paths on the input graph contain common edges in comparison to the cases where the O-D paths on the input graph are edge-disjoint. Based on this argument, we consider input graph instances where the O-D paths have common edges in order to provide an improved lower bound on the competitive ratio of deterministic strategies. We note that edge costs represent the edge traveling times in the analysis provided in this section. First, we need to present the following definition.

Definition 3.3.1. For an integer value of k, consider the graph topology that consists of three connected parts (e.g., shown in Figure 3.2 for $k = 2$). The graph consists of (1) a *left* part which is a full binary tree of height k, where the root node of the binary tree is the leftmost node (node O in Figure 3.2 for $k = 2$). (2) A *right* part which is a full binary tree of height k, where the root node of the binary tree is the rightmost node (node D in Figure 3.2 for $k = 2$). (3) A *bridge* part that consists of 2^k edges such that each one connects a leaf node of the left part to a leaf node of the right part. We call the edges in the bridge part the *bridge edges*. For an integer value of k, we call such a graph topology the *adversary topology of order k*.

Theorem 3.3.1. *No deterministic online strategy achieves a competitive ratio better than* $2\lfloor\frac{k}{\lambda}\rfloor + 1$, *where*

$$\lambda = max_{L^* \in \{0,1,2,\ldots,L\}}\{max\{-1, \lfloor log_2 L^*\rfloor\} + max\{-1, \lfloor log_2(L - L^*)\rfloor\} + 2\}.$$

Proof. We assume that the input graph has the adversary topology of order k (Definition 3.3.1), where O is the root node of the left tree and D is the root node of the right tree. We set the traveling time of the bridge edges equal to one and the traveling

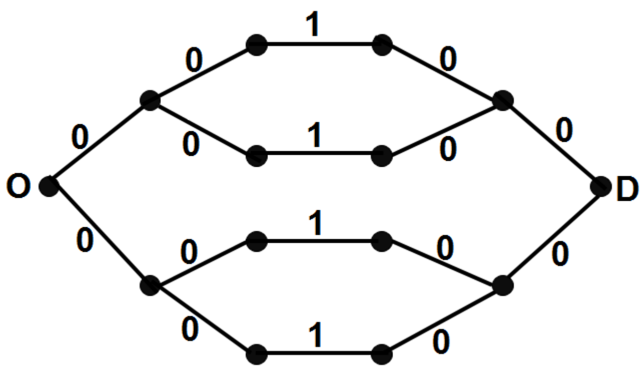

Figure 3.2: An adverse topology of order two

time of the other edges equal to zero. Figure 3.2 represents an adverse topology of order two with specified values. We assume that the blocked edges are located at the right tree. Note that all of the agents are at O at time zero. We organize the rest of our proof in three parts.

- **Representation of an arbitrary deterministic strategy** ALG^D. Suppose that an arbitrary deterministic strategy ALG^D ends in time interval $[t^*, t^*+1)$, where t^* is a non-negative integer number. Then, ALG^D can be represented by set of pairs $\{([i, i+1), L_i)\}$ for $i = 0, 1, 2, ..., t^*$, where $[i, i+1)$ represents a time interval and L_i ($L_i \in \{0, 1, 2, ..., L\}$) denotes the number of agents which depart from the left tree by taking a bridge edge in time interval $[i, i+1)$. We need to prove the following lemma which we apply in the rest of our proof.

  Lemma 3.3.1. *It holds that* $L_i + L_{i+1} \leq L$ $(i = 0, 1, 2, ..., t^*)$ *for* ALG^D *applied to an adversary topology of order* k.

  Proof. Note that L_i agents that leave the left tree in time interval $[i, i+1)$ $(i = 0, 1, 2, ..., t^*)$ arrive at the right tree in time interval $[i+1, i+2)$ since we assumed that the blocked edges are at the right tree, i.e. none of the bridge edges are blocked. Also note that L_i agents that leave the left tree in time interval $[i, i+1)$ do not arrive back to the left tree in time interval $[i+1, i+2)$

since traversing forward and traversing back a bridge edge takes at least two units of time. Hence $L_{i+1} \leq L - L_i$. □

- **Configuration of adversary blocked edges**. Note that the left tree consists of $k+1$ levels, where the root node of the left tree has level one. Similarly, the right tree consists of $k+1$ levels, where the root node of the right tree has level one. Also note that level j ($j \in \{1, 2, ..., k+1\}$) consists of 2^{j-1} nodes. We label the nodes arbitrarily such that v_{wj}^{left} represents the wth ($w = 1, 2, ..., 2^{j-1}$) node of the jth level on the left tree, and v_{wj}^{right} represents the wth ($w = 1, 2, ..., 2^{j-1}$) node of the jth level on the right tree. For ALG^D, we let x_{wj}^i denote the number of agents that visit v_{wj}^{left} within time interval $[i, i+1)$ for $i = 0, 1, 2, ..., t^*$. For ALG^D, we assume that the online adversary blocks the edges according to the following strategy which we call the *blocking strategy.*

  Blocking Strategy:

  - Initialization. Define i and j as counter variables and set $i = 0$ and $j = 2$ initially.
  - Step 1. If there exists at least one node such that $x_{wj}^i > 0$, go to Step 2. Otherwise, set $i = i + 1$ and go to the beginning of Step 1.
  - Step 2. Determine $w^* \in \{1, 2, ..., 2^{j-1}\}$ such that $x_{w^*j}^i = max_{w\in\{1,2,...,2^{j-1}\}}\{x_{wj}^i\}$. Block the edge which emanates from $v_{w^*j}^{right}$ and enters a node on the $(j-1)$th level of the right tree, and set $j = j + 1$. If $j > k$, stop; otherwise, go to Step 1.

- **Deriving the lower bound**. We need to prove the following lemma to derive our lower bound.

Lemma 3.3.2. *The agents can find at most λ blocked edges within time interval $(2i, 2i+2]$ for $i = 0, 1, 2, ..., 2\lfloor \frac{t^*}{2} \rfloor$ in ALG^D applied to adversary topology of order*

k with specified edge traveling time values, where

$$\lambda = max_{L^* \in \{0,1,2,...,L\}}\{max\{-1, \lfloor log_2 L^* \rfloor\} + max\{-1, \lfloor log_2 (L - L^*) \rfloor\} + 2\}.$$

Proof. Consider the L_{2i} agents that leave the left tree in time interval $[2i, 2i+1)$ for $i = 0, 1, 2, ..., 2\lfloor \frac{t^*}{2} \rfloor$ in ALG^D. These agents do not find a blocked edge before and after time interval $[2i + 1, 2i + 2)$ since all of them should traverse a bridge edge to arrive at the right tree and traversing a bridge edge takes at least one unit of time. Suppose that the blocking strategy generated $m < k$ blocked edges before investigating ALG^D in time interval $[2i, 2i + 1)$. We assume that the m blocked edges are known to the L_{2i} agents. If $L_{2i} = 0$, the blocking strategy generates no blocked edge when time interval $[2i, 2i + 1)$ is investigated. If $2i = t^*$, ALG^D ends before time $2i + 2$. Otherwise, the blocking strategy blocks an edge emanating from the $(m+u)$th level of the right tree for $u = 1, 2, ..., \lfloor log_2 L_{2i} \rfloor + 1$ such that all of the L_{2i} agents encounter a blocked edge on their assigned O-D paths in ALG^D (see Step 2 of the blocking strategy). Thus, the L_{2i} agents find at most $max\{-1, \lfloor log_2 L_{2i} \rfloor\} + 1$ blocked edges within time interval $[2i+1, 2i+2)$ for $i = 0, 1, 2, ..., 2\lfloor \frac{t^*}{2} \rfloor$ in ALG^D. Similarly, the L_{2i+1} agents that leave the left tree in time interval $[2i+1, 2i+2)$ find at most $max\{-1, \lfloor log_2 L_{2i+1} \rfloor\}+1$ blocked edges within time interval $[2i+2, 2i+3)$ in ALG^D. To draw a smallest competitive ratio, we assume that the L_{2i+1} agents encounter $max\{-1, \lfloor log_2 L_{2i+1} \rfloor\} + 1$ blocked edges at time $2i + 2$, i.e. the earliest possible time. Thus, the L_{2i} and L_{2i+1} agents can find at most $max\{-1, \lfloor log_2 L_{2i} \rfloor\} + max\{-1, \lfloor log_2 L_{2i+1} \rfloor\} + 2$ within time interval $(2i, 2i + 2]$. Note that $L_{2i+1} \leq L - L_{2i}$ according to Lemma 3.3.1. Therefore, the agents can find at most $max\{-1, \lfloor log_2 L_{2i} \rfloor\} + max\{-1, \lfloor log_2 (L - L_{2i}) \rfloor\} + 2$ blocked edges within time interval $(2i, 2i + 2]$. Since $max\{-1, \lfloor log_2 L_{2i} \rfloor\} + max\{-1, \lfloor log_2 (L - L_{2i}) \rfloor\} + 2$ is less than or equal to $\lambda = max_{L^* \in \{0,1,2,...,L\}}\{max\{-1, \lfloor log_2 L^* \rfloor\} + max\{-1, \lfloor log_2 (L - L^*) \rfloor\} + 2\}$ for $i = 0, 1, 2, ..., 2\lfloor \frac{t^*}{2} \rfloor$, the lemma follows.

□

It takes at least $\lfloor \frac{k}{\lambda} \rfloor$ time intervals with length two from time zero to find the k blocked edges. It takes one unit of time to arrive at D when all of the blocked edges are identified. Thus, no deterministic strategy end earlier than time $2\lfloor \frac{k}{\lambda} \rfloor + 1$. The theorem follows since it takes one unit of time from O to D in the offline optimal strategy.

□

Corollary 3.3.1. *The lower bound of* $2\lfloor \frac{k}{L} \rfloor + 1$ *on the competitive ratio of deterministic strategies is not tight for the multi-agent k-CTP with complete communication.*

3.3.4 Concluding remarks

We proved an improved lower bound on the competitive ratio of deterministic online strategies for the multi-agent k-CTP with complete communication. We designed instances of an input graph in which the O-D paths have common edges. Our analysis shows that the competitive ratio of deterministic strategies may reduce in the cases where the O-D paths on the input graph contain common edges in comparison to the cases in which the O-D paths on the input graph are edge-disjoint.

3.4 Summary of the results

Table 3.2 presents a summary of the results of Chapter 3.

Table 3.2: Summary of the results in Chapter 3

Problem	Result	Case	Network Type	Publication Status
Multi-agent k-CTP without communication	Lower bound	Randomized	General networks	Published in [52]
Multi-agent k-CTP with limited communication	Lower bound	Deterministic	General networks	Published in [51]
Multi-agent k-CTP with limited communication	Optimal strategy	Deterministic	Node-disjoint O-D paths	Published in [51]
Multi-agent k-CTP with limited communication	Lower bound	Randomized	General networks	Published in [52]
Multi-agent k-CTP with limited communication	Optimal strategy	Randomized	Node-disjoint O-D paths	Published in [52]
Multi-agent k-CTP with limited communication	Lower bound	Deterministic	General networks	Published in [51]
Multi-agent k-CTP with complete communication	Optimal strategy	Deterministic	Node-disjoint O-D paths	Published in [51]
Multi-agent k-CTP with complete communication	Lower bound	Randomized	General networks	Published in [52]
Multi-agent k-CTP with complete communication	Optimal strategy	Randomized	Node-disjoint O-D paths	Published in [52]
Multi-agent k-CTP with complete communication	Lower bound	Deterministic	General networks	In preparation

Chapter 4

ONLINE MINIMUM LATENCY PROBLEM WITH EDGE UNCERTAINTY

4.1 Introduction

The *minimum latency problem* (MLP) is a well-studied problem in combinatorial optimization. In the MLP, an undirected simple connected graph $G = (V, E)$ is given to an agent, where $V = \{v_0, v_1, v_2, ..., v_n\}$ and $v_0 \in V$ is a root node. Non-negative edge distances are also given. The agent should start from v_0 and complete a tour visiting all the nodes. The latency of v_i is denoted by l_i, which represents the distance traveled before first visiting v_i. Naturally, l_0 is zero. The objective of the agent is to find a tour on G, starting from v_0, that minimizes $\sum_{i=1}^{n} l(i)$. This problem is also known as the *deliveryman problem* [1] or the *traveling repairman problem* [30]. The MLP is an NP-hard problem [50] and it is APX-hard, implying the non-existence of a polynomial-time approximation scheme (PTAS) unless P=NP [54]. Several exact algorithms have been proposed for the MLP (see [44], [65], [45], [7] and [6]). Approximation algorithms for the MLP have been extensively investigated (see [19], [32], [9] and [25]), and the best approximation ratio achieved to date is 3.59 which is presented in [25].

We consider an online variant of the problem, in which k edges of G are blocked, and the agent only learns that an edge $e \in E$ is blocked, if she reaches at one of the end-nodes of e. The graph remains connected if the blocked edges are removed from it. The objective of the problem is to provide an online strategy such that the agent finds a feasible tour, i.e. one without blocked edges, starting from v_0 which minimizes $\sum_{i=1}^{n} l(i)$. This problem is called the *online minimum latency problem with* *edge uncertainty* (OMLP) and has been recently studied in [70].

Online strategies are divided into two categories as deterministic and randomized. In a deterministic strategy, actions of the decision maker do not depend on probabilistic outcomes, whereas in a randomized strategy the actions of the decision maker are taken according to some probability distribution. To evaluate the performance of online strategies, the notion of *competitive ratio* has been introduced by Sleator and Tarjan [55] and adopted by many researchers afterwards. For a deterministic strategy, ALG^D, the competitive ratio is the maximum ratio of the cost of a feasible solution found by ALG^D to the cost of the offline optimum over all instances of the problem. For a randomized strategy, ALG^R, the expected competitive ratio is the maximum ratio of the expected cost of a feasible solution found by ALG^R to the cost of the offline optimum over all instances of the problem. In the offline OMLP the blocked edges are removed from the graph. Hence, solving the offline OMLP is equivalent to solving an MLP.

4.1.1 Our contributions

A lower bound of $2k + 1$ has been derived for the competitive ratio of deterministic online strategies for OMLP in [70]. However, a deterministic online strategy which meets the lower bound of $2k + 1$ is not provided. In this section, we prove that the lower bound of $2k+1$ is tight by introducing an optimal deterministic online strategy whose competitive ratio matches the lower bound. Furthermore, we prove that no randomized online strategy can achieve an expected competitive ratio better than $k + 1$ for OMLP.

4.2 An optimal deterministic strategy

In this section we present an optimal deterministic online strategy for OMLP. Our strategy is iterative and terminates in at most $k + 1$ iterations. At the beginning of the qth ($q \in \{1, 2, ..., k + 1\}$) iteration, the agent removes the found blocked edges from the graph and calls an exact MLP to compute a tour T_q which starts from v_0 and minimizes total latency of all the nodes. We note that several exact MLP formulations are designed for complete graphs (see [7] and [6]). To apply these formulations, G can be transformed to a complete graph, $G^{'} = (V, E^{'})$, such that an edge $e_{ij} \in E^{'}$ corresponds to a shortest path between v_i and v_j on G, for $i, j \in \{0, 1, ..., n\}$. For any tour $T^{'}$ on $G^{'}$, we can construct a corresponding tour T on G by replacing each edge in $T^{'}$ with the corresponding shortest path on G. Wu et al. [65] discussed that the sum of the latencies of the nodes on T is less than or equal to the sum of the latencies of the nodes on $T^{'}$. Thus, an exact solution on G can be obtained. After T_q is constructed, the agent travels on T_q and either visits all of the nodes in V or finds a new blocked edge. In the former case, the iteration ends and the strategy stops. In the latter case, the agent backtracks to v_0 and the iteration ends. Due to the nature of our strategy, we call it the *Backtrack* strategy.

Backtrack Strategy

- **Initialization.** The agent takes the graph $G = (V, E)$ as input. Let F denote the set of found blocked edges and set $F = \emptyset$. Let q be a counter variable which represents the iteration number and set $q = 1$. Let G_q represent the graph at the beginning of the qth iteration and set $G_1 = G$.

- **Step 1.** The agent applies an exact MLP formulation and obtains an optimal tour T_q on G_q which starts from v_0 and minimizes $\sum_{i=1}^{n} d_i^q$, where d_i^q ($i \in \{1, 2, ..., n\}$) is the distance from v_0 to the first visit of v_i on T_q. Then, she starts traversing T_q. If there is no blocked edge on T_q, the strategy ends. Otherwise, a new blocked edge is found. In this case, the agent adds the found blocked edge to F and backtracks to v_0. Then, she sets $q = q + 1$, $G_q = (V, E - F)$, and goes to the beginning of Step 1.

Let k denote the number of blocked edges in G. Below, we show that the backtrack strategy matches the lower bound of $2k + 1$ given in [70].

Theorem 4.2.1. *The backtrack strategy achieves the competitive ratio of* $2k+1$ *for the OMLP.*

Proof. Let $d_i^{q^*}$ ($q^* \in \{1, 2, ..., k+1\}$) denote the distance between v_0 and v_i in the offline optimum tour, i.e. the total latency of the offline optimum is $\sum_{i=1}^{n} d_i^{q^*}$. Also let E^q ($q \in \{1, 2, ..., k+1\}$) denote the set of edges in G_q at the beginning of the qth iteration. Since $E^1 \supset E^2 \supset ... \supset E^{q^*}$, we have

$$\sum_{i=1}^{n} d_i^1 \leq \sum_{i=1}^{n} d_i^2 \leq ... \leq \sum_{i=1}^{n} d_i^{q^*-1} \leq \sum_{i=1}^{n} d_i^{q^*}.$$

The rest of our proof is by induction on k.

- **Base case.** For $k = 1$, the strategy terminates in at most two iterations. If the strategy ends in the first iteration, the competitive ratio would be one. Otherwise, the agent finds a blocked edge on T_1 and backtracks to v_0. Suppose that the agent has visited x number of nodes on T_1 before facing a blocked edge. Let $X \subset V$ denote the set of visited nodes. Let Δ denote the distance on T_1 that the agent has taken from v_0 to the end-node of the blocked edge. The total latency of the strategy can be represented as

  $$\sum_{i \in X} d_i^1 + 2\Delta(n-x) + \sum_{u \in V-X} d_u^{q^*}.$$

  It is straightforward to show that

  $$\sum_{i \in X} d_i^1 + \Delta(n-x) \leq \sum_{i=1}^{n} d_i^1 \leq \sum_{i=1}^{n} d_i^{q^*}.$$

  It follows that

  $$\sum_{i \in X} d_i^1 + 2\Delta(n-x) \leq 2(\sum_{i \in X} d_i^1 + \Delta(n-x)) \leq 2(\sum_{i=1}^{n} d_i^1) \leq 2(\sum_{i=1}^{n} d_i^{q^*}).$$

  Note that $\sum_{u \in V-X} d_u^{q^*} \leq \sum_{i=1}^{n} d_i^{q^*}$. Hence the total latency of the strategy can be bounded from above by $3\sum_{i=1}^{n} d_i^{q^*}$. Since the total latency of the offline optimum is $\sum_{i=1}^{n} d_i^{q^*}$, the base case follows.

- **Induction.** We assume that our claim is valid for $k = q - 1$ and we prove its correctness for $k = q$. If the strategy stops in the first iteration, the competitive ratio would be one. Otherwise, the agent finds a blocked edge on T_1 and backtracks to v_0. Suppose that the agent has visited x number of nodes on T_1 before facing a blocked edge. Let $X \subset V$ denote the set of visited nodes. Let Δ denote the distance on T_1 that the agent has taken from v_0 to the end-node of the blocked edge. Similar to the case with $k = 1$, we represent the total latency of the strategy at the end of the first iteration as

$$\sum_{i \in X} d_i^1 + 2\Delta(n - x) + C,$$

where C denotes the total latency of the nodes in $V - X$ from the end of the first iteration. Note that at the end of the first iteration the agent is at v_0 and there are $k - 1$ blocked edges in the graph. Thus, C is at most $(2q-1)\sum_{i=1}^{n} d_i^{q^*}$ according to the induction assumption. Also note that $\sum_{i \in X} d_i^1 + 2\Delta(n - x) \leq 2\sum_{i=1}^{n} d_i^{q^*}$. The theorem follows.

□

Note that the backtrack strategy does not run in polynomial time since it uses an exact MLP strategy to compute T_q ($q \in \{1, 2, ..., k+1\}$) in Step 1. One can utilize a deterministic approximation strategy for MLP to compute T_q in Step 1 of the backtrack strategy to obtain a polynomial time deterministic strategy for OMLP. Let ALG_α^D be a deterministic strategy for OMLP in which the agent applies an $\alpha-$approximation strategy instead of an exact one in Step 1 of the backtrack strategy. Similar to the proof of Theorem 4.2.1, one can verify that ALG_α^D achieves the competitive ratio of $\alpha(2k + 1)$ for OMLP. Note that best known α to date is 3.59 [25].

4.3 A lower bound on the expected competitive ratio of randomized strategies for OMLP

In this section we prove that no randomized online strategy can achieve an expected competitive ratio better than $k+1$ for OMLP. We apply Yao's Principle [68] to derive our lower bound. Yao [68] showed that the expected cost of a randomized strategy on the worst-case input is no better than that of a worst-case random probability distribution of the deterministic strategy which performs the best for that distribution. We refer the reader to [64] and [52] for applications of Yao's Principle to the k-Canadian Traveler's Problem and its variants.

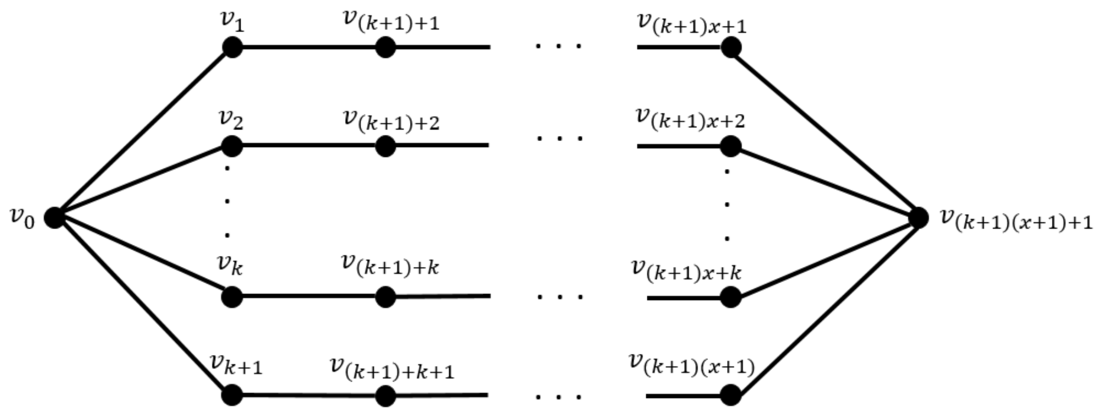

Figure 4.1: The graph for deriving the lower bound of $k+1$

Proposition 4.3.1. *No randomized strategy achieves an expected competitive ratio better than $k+1$ for OMLP.*

Proof. We consider the graph in Figure 4.1, where

$$V = \{v_0, v_{(k+1)(x+1)+1}\} \cup \{v_{(k+1)u+q} | u = 0, 1, ..., x, q = 1, 2, ..., k+1\}$$

and

$$E = \{(v_0, v_q), (v_{(k+1)u+q}, v_{(k+1)(u+1)+q}), (v_{(k+1)x+q}, v_{(k+1)(x+1)+1})\} | u = 0, 1, ..., x-1, 1 \leq q \leq k+1\}.$$

Let the distance of the edges (v_0, v_q) be one for $q = 1, 2, ..., k+1$ and let the distances of the other edges be ϵ, where ϵ is a positive number which approaches zero. That is, we do not consider ϵ values in our analysis. We choose $r \in \{1, 2, ..., k+1\}$ uniformly at random. Let $(v_r, v_{(k+1)+r})$ be traversable and block the edges $(v_{q'}, v_{(k+1)+q'})$ for $q^{'} \in \{\{1, 2, ..., k+1\} - \{r\}\}$. Also let $V^{'} = \{V - \{v_0, v_1, v_2, ..., v_{k+1}\}\}$.

In the offline optimum, the agent starts from v_0 and traverses the edge (v_0, v_r) to arrive at v_r. Then, she traverses all of the edges with distance ϵ to visit the $x(k+1)+1$ number of nodes in $V^{'}$. Thus, the latency of a node $v_i \in V^{'}$ is one in the offline optimum, i.e. the total latency of the nodes in $V^{'}$ is $x(k+1)+1$. Since the edges $(v_{q'}, v_{(k+1)+q'})$ are blocked, the agent has to traverse the edge $(v_0, v_{q'})$ to visit $v_{q'}$ for $q^{'} \in \{\{1, 2, ..., k+1\} - \{r\}\}$. Hence, the total latency of the nodes in $V - V'$ is $\sum_{q=1}^{k+1} 2q - 1 = (k+1)^2$. Therefore, the expected total latency of the offline optimum can be represented as $x(k+1) + 1 + (k+1)^2$.

We consider an arbitrary deterministic strategy, ALG^D, applied to the graph in Figure 4.1. In ALG^D, the agent starts from v_0 and takes an edge (v_0, v_q) ($q \in \{1, 2, ..., k+1\}$). Then, two cases may happen. In the first case, the agent learns that the edge $(v_q, v_{(k+1)+q})$ is traversable and takes it. Then, she traverses all of the edges with distance ϵ to visit the $x(k+1)+1$ number of nodes in $V^{'}$. In the second case, she learns that the edge $(v_q, v_{(k+1)+q})$ is blocked, backtracks to v_0 and tries a new edge. It is straightforward to show that the agent visits v_r after visiting $q-1$ nodes which belong to $\{\{v_1, v_2, ..., v_{k+1}\} - \{v_r\}\}$ with probability $\frac{1}{k+1}$ in ALG^D for $q = 1, 2, ..., k+1$. If the agent has visited $q-1$ nodes in $\{\{v_1, v_2, ..., v_{k+1}\} - \{v_r\}\}$ before visiting v_r, the total latency of the nodes in $V^{'}$ can be represented as $(2q-1)(x(k+1)+1)$. Note that since the edges $(v_{q'}, v_{(k+1)+q'})$ are blocked, the agent has to traverse the edge $(v_0, v_{q'})$ to visit $v_{q'}$ for $q^{'} \in \{\{1, 2, ..., k+1\} - \{r\}\}$. Hence, the total latency of the nodes in $V - V'$ is $\sum_{q=1}^{k+1} 2q - 1 = (k+1)^2$ in ALG^D. Therefore, the expected total latency of

Table 4.1: Summary of the results in Chapter 4

Problem	Result	Case	Network Type	Publication Status
OMLP	Optimal strategy	Deterministic	General networks	In preparation
OMLP	Lower bound	Randomized	General networks	In preparation

ALG^D can be written as

$$(k+1)^2 + \frac{1}{k+1}\sum_{q=1}^{k+1}(2q-1)(x(k+1)+1) = (k+1)^2 + \frac{x(k+1)+1}{k+1}\sum_{q=1}^{k+1}(2q-1),$$

which is equal to $(k+1)^2 + (x(k+1)+1)(k+1)$. Since the expected total latency of the offline optimum is $x(k+1)+1+(k+1)^2$, when x approaches $+\infty$, the proposition follows by Yao's Principle.

□

4.4 Concluding remarks

We proved that our backtrack strategy is optimal for OMLP since it achieves the competitive ratio of $2k+1$. That is, we showed that the lower bound of $2k+1$ on the competitive ratio of deterministic online strategies is tight for OMLP. We also proved that no randomized online strategy can achieve an expected competitive ratio better than $k+1$ for OMLP. However, devising a randomized online strategy which matches the expected competitive ratio of $k+1$ for OMLP remains as an open research problem.

Table 4.1 presents a summary of the results of Chapter 4.

Chapter 5

ONLINE DISCRETE SEARCH PROBLEM WITH TRAVELING AND SEARCH COST ON UNDIRECTED GRAPHS

5.1 Introduction

Search theory is one of the oldest research areas within the field of Operations Research. Nunn [47] stated that problems involving search arise in such diverse areas as the military looking for enemy submarines, the coast guard searching for small boats lost in a storm, prospectors surveying for mineral deposits, a crew searching missing backpackers in the forest, law enforcement officers looking for lost weapons or escaped criminals, a secretary looking for missing file, or an analyst scanning a computer print-out for missing data. Locating a victim during search-and-rescue operations after a disaster can be added to this list.

In problems involving search, there is a *searcher* who probes for a *hider*. The hider can be *static* or *moving* depending on the context. The focus of our study is on search problems with a static hider. For problems with a moving hider, see [22], [23], and [60]. The static hider can be positioned in a *discrete* or *continuous* search domain. We call the search problems with a discrete search domain *discrete search problems* and the search problems with a continuous search domain *continuous search problems*.

In search problems, the common unknown information to the searcher is the location of the hider. The objective of the searcher is to provide a strategy which finds the hider with minimum total cost. The searcher has to devise his strategy under incomplete information. The approach usually taken for such problems is to consider some probabilistic model associated with uncertain information and act on this basis. Our approach is to compare the performance of the strategy that operates under incomplete information, i.e. the *online* strategy, with the performance of the strategy that operates in presence of complete information, i.e. the *offline* strategy. This approach requires no probabilistic knowledge of the future and is therefore a worst-case measure of performance. This type of analysis was first suggested in [55] and later called *competitive analysis* in [36]. Online strategies are divided into two categories as deterministic and randomized. In a deterministic strategy, actions of the decision maker do not depend on probabilistic outcomes, whereas in a randomized strategy the actions of the decision maker are taken according to some probability distribution [3].

There are two main motivations to analyze discrete search problems in terms of competitive analysis. The first one is that in several real-world applications of these problems, it is very difficult to obtain probabilistic knowledge about the location of the hider. The second motivation is that in many real-world applications of discrete search problems such as security and defense, for instance in a bomb exploration operation, having a strategy with a good worst-case performance is vital since human life is at stake.

5.1.1 Problem Definition

We study an online variant of discrete search problems as follows. Given an undirected connected graph $G = (V, E)$, where $V = \{v_0, v_1, v_2, ..., v_n\}$, node v_i ($i \in \{1, 2, ..., n\}$) of the graph is associated with a given non-negative search cost s_i. A non-negative edge cost d_e is also given for each edge $e \in E$. A static hider is at one of the nodes v_{i^*} ($i^* \in \{1, 2, ..., n\}$) which is not known to the searcher. The hider is not found unless the searcher arrives at v_{i^*} and incurs the search cost of v_{i^*}. Starting from v_0, the searcher wants to devise an online strategy to locate the hider with minimum total cost. We call this problem the *online discrete search problem on undirected graphs* (ODSP). The ODSP has not been studied from the competitive analysis perspective in the literature. We note that the second stream of search problems on graphs are the most related type of work in the literature to the ODSP.

To evaluate the performance of online strategies, the notion of *competitive ratio* has been introduced by Sleator and Tarjan [55] and adopted by many researchers. For a deterministic strategy, the competitive ratio is the maximum ratio of the cost of the online strategy to the cost of the offline strategy over all instances of the problem. For a randomized strategy, the expected competitive ratio is the maximum ratio of the expected cost of the online strategy to the cost of the offline strategy over all instances of the problem. In the offline version of the problem, the hideout node of the hider (v_{i^*}) is known to the searcher. Hence, the cost of the offline optimum equals the cost of the shortest (cheapest) path from v_0 to v_{i^*} plus the search cost of v_{i^*}.

5.1.2 Our Contributions

Several past studies have conducted competitive analysis on the variants of online continuous search problems which are defined on graphs, see [8], [26] and [35]. However, to the best of our knowledge, online discrete search problems have not been studied from the competitive analysis point of view. In this section, we investigate an online discrete search problem which we call the ODSP from the competitive analysis perspective for the first time. The ODSP finds applications in diverse areas such as security, defense, and search-and-rescue. We provide policies that are optimal with respect to the worst-case scenarios for such applications. We derive a tight lower bound on the competitive ratio of deterministic strategies and propose an optimal deterministic strategy. We also provide a tight lower bound on the expected competitive ratio of randomized strategies and prove its tightness by introducing an optimal randomized strategy. In this way, we show that randomized strategies can achieve a better competitive ratio in comparison to deterministic strategies for the ODSP in the expected sense. We note that our proofs for randomized strategies are more challenging.

5.2 Tight lower bound on the competitive ratio of deterministic strategies for the ODSP

To prove a lower bound LB^D on the competitive ratio of deterministic strategies for an arbitrary online problem, it suffices to prove that for an arbitrary deterministic strategy, ALG^D, there exists at least one instance of inputs I^D such that ALG^D cannot achieve a competitive ratio better than LB^D on I^D. We remind the reader that $V = \{v_0, v_1, v_2, ..., v_n\}$ and we assume that the search domain is $\{v_1, v_2, ..., v_n\}$.

Theorem 5.2.1. *There is no deterministic strategy with a competitive ratio less than* $2n - 1$ *for the ODSP.*

Proof. We consider an arbitrary deterministic strategy, ALG^D for the ODSP, and prove that there exists at least one instance of inputs, I^D, such that the competitive ratio of ALG^D is not better than $2n - 1$ on I^D. We let the input graph G contain $n + 1$ nodes $v_0, v_1, v_2, ..., v_n$ and have the star topology shown in Figure 5.1, where v_0 is the root (hub) node. Let v_0 be the starting node of the searcher. In G, v_0 is connected to v_i via an undirected edge with cost one for $i = 1, 2, ..., n$, i.e. there are n edges in G. Let the search cost of v_i be zero, for $i \in \{1, 2, ..., n\}$. Note that ALG^D applied to G corresponds to a permutation which specifies in which order the n nodes should be searched.

For ALG^D, we let I^D be the instance in which the node of the hider is the node that is searched last. Therefore, the cost of ALG^D is $2n - 1$. Note that the cost of the offline optimum is one. Thus, the competitive ratio of ALG^D is $2n - 1$.

□

5.3 An optimal deterministic strategy for the ODSP

In this section, we provide an optimal deterministic strategy for the ODSP. To provide the rest of our results in this section, we need to present the following definition.

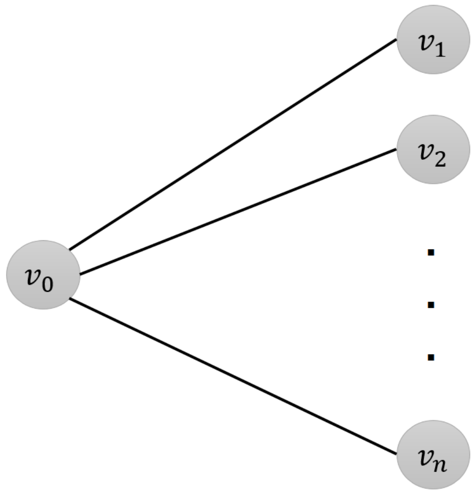

Figure 5.1: A star graph with n nodes

Definition 5.3.1. We define the *operational cost* of v_i ($v_i \in V$) as the summation of the cost of the cheapest path between v_0 and v_i together with the search cost of v_i for $i = 1, 2, ..., n$. We denote the value of the operational cost of v_i by c_i for $i = 1, 2, ..., n$.

Now, we can present our optimal deterministic strategy which we call the *backtrack strategy.*

Backtrack Strategy:

- **Initialization.** Take $G = (V, E)$, the starting node of the searcher v_0, s_i for $i = 1, 2, ..., n$, together with d_e for all $e \in E$ as input. Compute the operational costs of the nodes in $V - \{v_0\}$. Re-label the nodes excluding v_0 from v_1 to v_n such that $c_i \leq c_j$ for $1 \leq i < j \leq n$, i.e. in non-decreasing order of their operational costs. Define k as a counter variable and set its initial value to one.

- **Step 1.** Take the cheapest path from v_0 to v_k and conduct a search at v_k. If the hider is found, stop. Otherwise, retrace back to v_0 by taking the cheapest path from v_k to v_0, set $k = k + 1$ and go to the beginning of Step 1.

Theorem 5.3.1. *The backtrack strategy achieves the optimal competitive ratio of*

$2n-1$ for the ODSP.

Proof. Note that the backtrack strategy labels the starting node of the searcher by v_0 and labels the other nodes by v_i for $i = 1, 2, ..., n$ such that $c_1 \leq c_2 \leq ... \leq c_n$ (e.g., shown in Figure 5.2 for an instance of the ODSP). Suppose that the hider is positioned at v_{i^*} ($i^* \in \{1, 2, ..., n\}$). Hence the cost of the offline optimum is c_{i^*}. Note that the competitive ratio of the backtrack strategy can be bounded from above by

$$\frac{\sum_{k=1}^{i^*-1} 2c_k}{c_{i^*}} + 1,$$

which is at most $2n-1$ when $i^* = n$ and $c_1 = c_2 = ... = c_n$. □

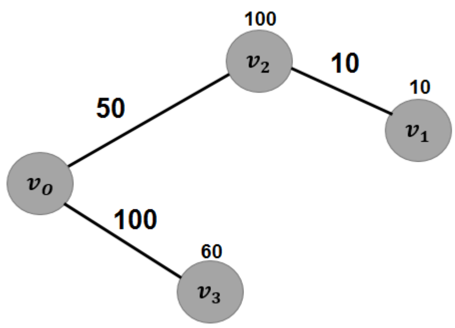

Figure 5.2: An instance of the ODSP

In the next section, we investigate whether randomized strategies can achieve a better expected competitive ratio in comparison to the competitive ratio of the best deterministic strategy (backtrack strategy) for the ODSP.

5.4 *Tight lower bound on the expected competitive ratio of randomized strategies for the ODSP*

A lower bound on the competitive ratio is usually derived by providing a set of specific instances on which no online strategy can perform well compared to an optimal offline strategy. For deterministic strategies finding a suitable instance is comparatively easy

[40]. For randomized strategies, however, it is usually very difficult to bound the expected cost of an arbitrary randomized strategy on a specific instance from below [40].

Yao's Principle is a standard tool for providing lower bounds on the expected competitive ratio of randomized strategies. Yao [68] showed that the expected cost of a randomized strategy on the worst-case input is no better than that of a worst-case random probability distribution of the deterministic strategy which performs the best for that distribution. This principle was proven in [68] and presented to be applied for driving the lower bounds on the expected competitive ratio of randomized strategies for the online problems in [40]. Yao's Principle allows us to trade randomization in an online strategy for randomization in the input [40]. We apply Yao's Principle [68] in the next lemma in order to provide a tight lower bound on the expected competitive ratio of randomized strategies for the ODSP.

Theorem 5.4.1. *No randomized strategy achieves an expected competitive ratio better than n for the ODSP.*

Proof. We consider the input graph G which is described in the proof of Theorem 5.2.1. We remind that G contains $n+1$ nodes $v_0, v_1, v_2, ..., v_n$ and have the star topology, where v_0 is the root (hub) node. The node v_0 is the starting node of the searcher and is connected to v_i via an undirected edge with cost one for $i = 1, 2, ..., n$. Also, the search cost of v_i is zero for $i \in \{1, 2, ..., n\}$. We choose $i^* \in \{1, 2, ..., n\}$ uniformly at random and assume that the hider is positioned at node v_{i^*}. Hence the expected cost of the offline optimum is one. We consider an arbitrary deterministic strategy ALG^D for the ODSP applied to G, and organize the rest of the proof in two parts.

- **Definition of iterations of ALG^D.** Note that ALG^D applied to G corresponds to a permutation which specifies in which order the nodes of the graph, excluding v_0, are being searched. Hence, we define the concept of iteration for ALG^D applied to G as follows. At the beginning of each iteration, the searcher takes one of the edges in G to arrive at one of the nodes that is not yet searched and conducts a search at that node. ALG^D ends if the hider is found, i.e. the cost of the iteration equals one and the iteration ends. Otherwise, the searcher backtracks to v_0 and the iteration ends, i.e. the cost of the iteration equals two. Also note that ALG^D ends within n iterations.

- **Computation of the expected competitive ratio of ALG^D.** Note that the searcher probes v_{i^*} at iteration i ($i \in \{1, 2, ..., n\}$) with probability $\frac{1}{n}$ in ALG^D, since we selected i^* according to the uniform probability distribution. If ALG^D ends at iteration i, the searcher incurs total cost of $2i - 1$. Thus, the expected cost of ALG^D is
$$\frac{1}{n}\sum_{i=1}^{n}(2i-1) = n.$$

We just showed that the expected cost of ALG^D with respect to the uniform distribution given on the input is n. It follows that no randomized strategy achieves an expected competitive ratio less than n on G against its worst-case input, by Yao's Principle. □

5.5 An optimal randomized strategy for the ODSP

We need to present the following definition to describe our strategy. Note that Bender and Westphal [17] defined the below property in a different structure for the k-Canadian Traveler Problem. Below, we provide the definition in a generalized structure.

Definition 5.5.1. The elements $\theta_1, \theta_2, ..., \theta_t$ which are associated with costs $\delta_1 \leq \delta_2 \leq ... \leq \delta_t$ have the *similar costs property* if for all $i = 1, 2, ..., t$, it holds that

$$\delta_i \leq \frac{2}{t}\sum_{j=1}^{t}\delta_j.$$

Hereafter, we say that the nodes $v_1, v_2, ..., v_t$ with operational costs $c_1, c_2, ..., c_t$

satisfy the similar costs property if for all $i = 1, 2, ..., t$, it holds that $c_i \leq \frac{2}{t} \sum_{j=1}^{t} c_j$ (Definition 5.5.1). We also use the following lemma to devise our optimal randomized strategy. We note that this lemma is proven in [17] in a different context, namely for the k-Canadian Traveler Problem. Below, we present the lemma in a generalized structure.

Lemma 5.5.1. *Suppose that the elements $\theta_1, \theta_2, ..., \theta_t$ which are associated with costs $\delta_1 \leq \delta_2 \leq ... \leq \delta_t$ satisfy the similar costs property. Then the probability distribution $\Omega_t = \lambda^* p^{'}$ belongs to the polyhedron Q_t, where Ω_t and $p^{'}$ are t-vectors, $\lambda^* = \sum_{i=1}^{t} \frac{1}{p_i^{'}} \in [0, 1]$,*

$$p_i^{'} = \frac{(2-t)\delta_i + \sum_{j=1 j\neq i}^{t} 2\delta_j}{t^2 \delta_i} \qquad \forall i = 1, 2, ..., t$$

and

$$Q_t = \{p \in \mathbb{R}_+^t : (2-t)p_i + \sum_{j=1 j\neq i}^{t} 2\frac{\delta_j}{\delta_i} p_j \leq 1 \;\; \forall i = 1, 2, ..., t, \sum_{i=1}^{t} p_i = 1\}.$$

Now, we can present our optimal randomized strategy which we call the *randomized backtrack strategy.*

Randomized Backtrack Strategy:

- **Initialization.** Take an undirected graph $G = (V, E)$, the starting node of the searcher v_0, s_i for all v_i for $i = 1, 2, ..., n$, together with d_e for all $e \in E$ as input. Define S as the *selection list* and let $S = \emptyset$, initially. Compute the operational costs of the nodes in $V - \{v_0\}$. Re-label the nodes excluding v_0 from v_1 to v_n such that $c_i \leq c_j$ for $1 \leq i < j \leq n$, i.e. in non-decreasing order of their operational costs. Define $S^{'}$ as the *search list* and set $S^{'} = \{v_1, v_2, ..., v_n\}$ initially. For any arbitrary set of t nodes $v_1, v_2, ..., v_t$ with operational costs $c_1 \leq c_2 \leq ... \leq c_t$ that satisfy the similar costs property, let the probability distribution $\Omega_t = (p_1, p_2, ..., p_t) \in Q_t$ be the probability distribution that is defined in Lemma 5.5.1.

- **Step 1.** Remove the nodes from S to make it empty. Add the nodes from $S^{'}$ to S by going through them in non-decreasing order of their operational costs until adding the next node violates the similar costs property. Go to Step 2.

- **Step 2.** Let t_S denote the number of nodes in S. Take one of the nodes in S according to the probability distribution Ω_{t_S}. If the hider is found, stop. Otherwise, a new node is searched and the searcher backtracks to v_0. Remove the searched node from $S^{'}$. Go to Step 1.

To prove that the randomized backtrack strategy is optimal, we utilize the following lemma regarding the elements $\theta_1, \theta_2, ..., \theta_t$ (which are associated with costs $\delta_1 \leq \delta_2 \leq ... \leq \delta_t$) that fulfill the similar costs property and the probability distribution $\Omega_t = (p_1, p_2, ..., p_t) \in Q_t$ (Lemma 5.5.1). We note that the following lemma is proven in [52] in a different structure for the k-Canadian Traveler Problem. Below, we state the lemma in a generalized structure.

Lemma 5.5.2. *Consider the vector $\Pi_t = (\pi_1, \pi_2, ..., \pi_t)$ such that $\pi_i = \frac{1}{t}$ for $i = 1, 2, ..., t$. It holds that $\sum_{i=1}^{t} p_i \delta_i \leq \sum_{i=1}^{t} \pi_i \delta_i$.*

We apply the above lemma in the context of the ODSP to prove that the randomized backtrack strategy meets the lower bound of n. That is, we consider the nodes $v_1, v_2, ..., v_t$ which are associated with operational costs $c_1, c_2, ..., c_t$ instead of the elements $\theta_1, \theta_2, ..., \theta_t$ which are associated with costs $\delta_1, \delta_2, ..., \delta_t$. We remind that n denotes the size of the search domain.

Theorem 5.5.1. *The expected competitive ratio of the randomized backtrack strategy is n for the ODSP.*

Proof. Our proof is by induction on n.

- **Base case.** When n is one the searcher takes the cheapest path from v_0 to the only node in the search domain with probability $p_1 = 1$, incurs the search cost of the node, and finds the hider. Thus the cost of the randomized backtrack strategy and the cost of the offline optimum would be c_1. Hence the competitive ratio is one.

- **Induction.** Let $v_1, v_2, ..., v_n$ $(n \geq 2)$ with operational costs $c_1 \leq c_2 \leq ... \leq c_n$ be the nodes in the search domain. Also let v_{i^*} $(i^* \in \{1, 2, ..., n\})$ be the node in which the hider is positioned. Hence the cost of the offline optimum is c_{i^*}. At the first implementation of Step 1, the nodes are added to the selection list in non-decreasing order of their operational costs until adding the next node to the selection list violates the similar costs property. Let $v_1, v_2, ..., v_t$ with operational costs $c_1, c_2, ..., c_t$ be the nodes that are added to the selection list after the first implementation of Step 1. Then the strategy enters Step 2. We present the rest of our proof by considering two cases.

  - **Case 1.** v_{i^*} **is added to the selection list.** The searcher takes the cheapest path from v_0 to $v_{i'}$ $(i' \in \{1, 2, ..., t\})$ according to the probability distribution $\Omega_t = (p_1, p_2, ..., p_t)$. Then he arrives at $v_{i'}$ and incurs its search cost. If the hider is not found, he backtracks to v_0 and discards $v_{i'}$ from the search list, i.e. the searcher incurs a cost of at most $2c_{i'}$. Otherwise, the hider is found and the strategy ends. Suppose that the hider is not found, and let C^{n-1} denote the expected cost of the randomized backtrack strategy from the end of the first iteration, i.e. when the searcher arrives back at v_0 for the first time, until the end of the strategy. The expected competitive ratio can be bounded from above by

$$p_{i^*} + \sum_{j=1 j \neq i^*}^{t} p_j \frac{2c_j + C^{n-1}}{c_{i^*}} \leq p_{i^*} + \sum_{j=1 j \neq i^*}^{n} p_j \frac{2c_j + C^{n-1}}{c_{i^*}}.$$

    Note that C^{n-1} is at most $(n-1)c_{i^*}$ by the induction assumption (for $n = 2$, $C^1 = c_1 = c_{i^*}$ according to the base case). Hence, the expected competitive ratio is bounded from above by

$$p_{i^*} + \sum_{j=1 j\neq i^*}^{n} p_j(\frac{2c_j}{c_{i^*}} + n - 1).$$

We claim that the right-hand side is at most n for all $i^* = 1, 2, ..., n$, i.e.

$$(2-n)p_i + \sum_{j=1 j\neq i}^{n} p_j \frac{2c_j}{c_i} \leq 1$$

for all $i = 1, 2, ..., n$. Since the probability distribution $\Omega_n = (p_1, p_2, ..., p_n)$ belongs to the polyhedron Q_n, the claim follows by the definition of Q_n in Lemma 5.5.1.

– **Case 2.** v_{i^*} **is not added to the selection list.** In this case the searcher takes the cheapest path from v_0 to $v_{i'}$ ($i' \in \{1, 2, ..., t\}$) according to the probability distribution $\Omega_t = (p_1, p_2, ..., p_t)$. Then he arrives at $v_{i'}$ and incurs its search cost. Note that the hider is not found at $v_{i'}$ since v_{i^*} is not added to the selection list. Then, the searcher backtracks to v_0 and discards $v_{i'}$ from the search list, i.e. the searcher incurs a cost of at most $2c_{i'}$. Hence, an expected cost of less than or equal to $2\sum_{i=1}^{t} p_i c_i$ is incurred. Note that $2\sum_{i=1}^{t} p_i c_i \leq 2\sum_{i=1}^{t} \frac{1}{t} c_i$ according to Lemma 5.5.2 since the nodes $v_1, v_2, ..., v_t$ satisfy the similar costs property by Definition 5.5.1. Let C^{n-1} denote the expected cost of the randomized backtrack strategy from the end of the first iteration, i.e. when the searcher arrives back at v_0 for the first time, until the end of the strategy. Note that C^{n-1} is at most $(n-1)c_{i^*}$ by the induction assumption. Thus the expected cost of the randomized backtrack strategy is at most $2(\sum_{i=1}^{t} \frac{1}{t} c_i) + (n-1)c_{i^*}$. Since v_{i^*} is not in the selection list, it follows that v_{i^*} does not fulfill the similar costs property with $v_1, v_2, ..., v_t$ and $c_t < c_{i^*}$. Thus, $\frac{2(\sum_{i=1}^{t} c_i) + 2c_{i^*}}{t+1} < c_{i^*}$ according to Definition 5.5.1. We first multiply both sides by $t+1$ and then eliminate $2c_{i^*}$ from both sides to obtain $2\sum_{i=1}^{t} \frac{1}{t} c_i < 2\sum_{i=1}^{t} \frac{1}{t-1} c_i < c_{i^*}$. Therefore, the expected cost of the randomized backtrack strategy is at most nc_{i^*}. It follows that the expected competitive ratio is at most n since the cost of the offline optimum is c_{i^*}.

□

Corollary 5.5.1. *The randomized backtrack strategy achieves a better competitive ratio than the optimal deterministic strategy (backtrack strategy) in the expected sense for the ODSP when* $n > 1$.

Proof. Note that the optimal deterministic strategy achieves the competitive ratio of $2n-1$ and the randomized backtrack strategy achieves the expected competitive ratio of n. Since $n < 2n-1$ for $n > 1$, the corollary follows. □

5.6 Concluding remarks

We studied an online variant of discrete search problems with a static hider that we call the ODSP from the competitive analysis point of view for the first time. In this variant, search costs are given on the nodes in addition to travel costs on the edges. The hider is positioned at a node of the input graph. We provided a tight lower bound on the competitive ratio of deterministic strategies together with an optimal deterministic strategy named the backtrack strategy. We also proved a tight lower bound on the expected competitive ratio of randomized strategies and introduced an optimal randomized strategy named the randomized backtrack strategy. We showed that randomized strategies can achieve a better competitive ratio in comparison to deterministic strategies in the expected sense. As a future research topic, one may study a version of online discrete search problems in which the traveling costs are negligible in comparison to the search costs.

Table 5.1 presents a summary of the results of Chapter 5.

Table 5.1: Summary of the results in Chapter 5

Problem	Result	Case	Network Type	Publication Status
ODSP	Tight lower bound	Deterministic	General networks	Under review
ODSP	Optimal strategy	Deterministic	General networks	Under review
ODSP	Tight lower bound	Randomized	General networks	Under review
ODSP	Optimal strategy	Randomized	General networks	Under review

Chapter 6

CONCLUSION AND FUTURE RESEARCH

In this chapter, we first summarize the thesis and then briefly discuss some open problems and possible directions for future research.

6.1 *Summary of the thesis*

In Chapter 1, we provided an overview of online optimization and methods of analyzing online strategies. We presented definitions of problems that are studied in this thesis and stated the related literature. Finally, we summarized our results and contributions.

In Chapter 2, we reconsidered the implementation of the RBS on graphs which contain n node-disjoint O-D paths for the online k-Canadian Traveler Problem. We showed that to implement the strategy, a certain property (strong similar costs property) regarding the costs of the O-D paths in the input graph must hold. That is we proved that the RBS is not applicable in some cases when $k > 2$. We showed that the RBS is applicable when the cost of the $(min\{k+1, n\})$th shortest O-D path is at most twice of the shortest path in the input graph. Furthermore, we modified the RBS to obtain an optimal strategy which is applicable on graphs having only node-disjoint O-D paths.

In the first section of Chapter 3, we analyzed the online multi-agent O-D k-Canadian Traveler Problem. We provided updated results including the lower bounds on the competitive ratio of deterministic strategies of the problem for the case where the communication is limited. We argued that it is vital to consider and utilize the higher levels of agents' intelligence in online problems by defining three levels of agents' intelligence. We introduced an online strategy in O-D edge-disjoint graphs which is optimal in both cases with complete and limited communication when the travel schedules are shared at the initial stage of the problem. We showed that enabling all of the agents to communicate does not improve the competitive ratio in O-D edge-disjoint graphs. Furthermore, we showed that there are instances with O-D edge-disjoint graphs in which the competitive ratio of deterministic strategies on P_2 improves by increasing the number of R-type agents.

In the second section of Chapter 3, we studied randomized online strategies for the multi-agent k-CTP. We analyzed the problem in three different cases: 1) without communication, 2) with limited communication, and 3) with complete communication. We proved lower bounds on the competitive ratio of the randomized online strategies for these cases. We introduced an optimal randomized strategy for the cases with limited and complete communication on O-D edge-disjoint graphs which finds real-life applications. We showed that our optimal randomized strategy S_3 achieves a better expected competitive performance in comparison to the optimal deterministic strategy (M-PLS) that is given in the literature. We also showed that having complete communication does not improve the competitive ratio of the optimal randomized strategy on O-D edge-disjoint graphs in comparison to the case when communication is limited. Additionally, we showed that increasing the number of agents can improve the competitive ratio of the randomized strategies when there is no communication between agents.

In the third section of Chapter 3, we derived an improved lower bound on the competitive ratio of deterministic strategies for the multi-agent k-CTP by investigating graphs in which the O-D paths contain common edges.

In Chapter 4, we proved that our backtrack strategy is optimal for the online minimum latency problem with edge uncertainty (OMLP) since it achieves the competitive ratio of $2k + 1$. That is, we showed that the lower bound of $2k + 1$ on the competitive ratio of deterministic online strategies is tight for the OMLP. We also proved that no randomized online strategy can achieve an expected competitive ratio better than $k + 1$ for the OMLP.

In Chapter 5, we investigated an online variant of discrete search problems with a static hider that we call the ODSP from the competitive analysis point of view for the first time. In this variant, search costs are given on the nodes in addition to travel costs on the edges. The hider is positioned at a node of the input graph. We provided a tight lower bound on the competitive ratio of deterministic strategies together with an optimal deterministic strategy named the backtrack strategy. We also proved a tight lower bound on the expected competitive ratio of randomized strategies and introduced an optimal randomized strategy named the randomized backtrack strategy. We showed that randomized strategies can achieve a better competitive ratio in comparison to deterministic strategies in the expected sense.

6.2 Implications of the study

Our results including optimal policies and characterization of the worst-case scenarios can be used in real-life applications in the areas of disaster response, search-and-rescue, security, and defense. The policies specify precise actions for the decision makers and the worst-case scenarios are useful to identify how an adversary would behave to cause the most challenge so that knowing this, the decision makers can take the necessary preventive actions under such a scenario.

We use intuitive heuristic ideas to derive our optimal deterministic strategies. Our policies can be utilized in order to devise solution strategies for network optimization problems with similar type of online uncertainty which are effective in the worst-case. We also use a common framework to design our randomized strategies by applying a specific probability distribution. Our framework can be helpful to obtain efficient randomized strategies with respect to the worst-case for network optimization problems having similar type of online uncertainty.

The limitation of our study is that our findings are based on worst-case analyses. However, our results can be considered as a first step for future results including policies which are effective on the average over a set of scenarios.

6.3 Future research directions

The following are several directions in which the research in this thesis can be extended.

- We note that the problem of designing an optimal randomized online strategy for the online k-CTP for the general case is an open research problem.

- We note that the problem of designing an optimal deterministic online strategy for the multi-agent k-CTP is an open research problem.

- We note that the problem of designing a randomized online strategy in the case without communication that meets the lower bound of the problem on O-D edge-disjoint graphs for the multi-agent k-CTP is an open research problem.

- As a future research topic, one may study a version of online discrete search problems in which the traveling costs are negligible in comparison to the search costs.

BIBLIOGRAPHY

[1] Foto N. Afrati, Stavros S. Cosmadakis, Christos H. Papadimitriou, George Papageorgiou, and Nadia Papaconstantinou. The complexity of the traveling repairman problem. *RAIRO-Theoretical Informatics and Applications*, 20:79–87, 1986.

[2] Vural Aksakalli, O. Furkan Sahin, and Ibrahim Ari. An ao* based exact algorithm for the canadian traveler problem. *Informs Journal on Computing*, 28:96–111, 2016.

[3] Susanne Albers. Online algorithms: A survey. *Mathematical Programming*, 97:3–26, 2003.

[4] Steve Alpern, Vic Baston, and Shmuel Gal. Searching symmetric networks with utilitarian-postman paths. *Networks*, 53:392–402, 2009.

[5] Steve Alpern and Shmuel Gal. *The Theory of Search Games and Rendezvous*. International Series in Operations Research and Management Science, Boston, 2003.

[6] F. Angel-Bello, Y. Cardona-Valdes, and A. Alvarez. Mixed integer formulations for the multiple minimum latency problem. *Operational Research*, 17:1–30, 2017.

[7] Francisco Angel-Bello, Ada Alvarez, and Irma Garcia. Two improved formulations for the minimum latency problem. *Applied Mathematical Modeling*, 37:2257–2266, 2013.

[8] Spyros Angelopoulos, Diogo Arsenio, and Christoph Durr. Infinite linear programming and online searching with turn cost. *Theoretical Computer Science*, 670:11–22, 2017.

[9] Aaron Archer, Asaf Levin, and David P. Williamson. A faster, better approximation algorithm for the minimum latency problem. *Siam J. Computing*, 37:1472–1498, 2008.

[10] Giorgio Ausiello, Vincenzo Bonifacia, and Luigi Lauraa. The online asymmetric traveling salesman problem. *Journal of Discrete Algorithms*, 6:290–298, 2008.

[11] Giorgio Ausiello, Vincenzo Bonifacia, and Luigi Lauraa. The online prize-collecting traveling salesman problem. *Information Processing Letters*, 107:199–204, 2008.

[12] Giorgio Ausiello, Marc Demange, Luigi Laura, and Vangelis Paschos. Algorithms for the online quota traveling salesman problem. *Information Processing Letters*, 92:89–94, 2004.

[13] Giorgio Ausiello, Esteban Feuerstein, S. Leonardi, L. Stougie, and Maurizio Talamo. Algorithms for the online traveling salesman. *Algorithmica*, 29:560–581, 2001.

[14] Amotz Bar-Noy and Baruch Schieber. The canadian traveler problem. *SODA '91 Proceedings of the second annual ACM-SIAM symposium on Discrete algorithms*, pages 261–270, 1991.

[15] Vic Baston and Kensaku Kikuta. Search games on networks with travelling and search costs and with arbitrary searcher starting points. *Networks*, 62:72–79, 2013.

[16] Vic Baston and Kensaku Kikuta. Search games on a network with travelling and search costs. *International Journal of Game Theory*, 44:347–365, 2014.

[17] Marko Bender and Stephan Westphal. An optimal randomized online algorithm for the k-canadian traveller problem on node-disjoint paths. *Journal of Combinatorial Optimization*, 30:87–96, 2015.

[18] Michiel Blom, Sven O. Krumke, Willem E. de Paepe, and Leen Stougie. The online tsp against fair adversaries. *Informs Journal on Computing*, 13:137–149, 2001.

[19] Avrim Blum, Prasad Chalasani, Don Coppersmith, Bill Pulleyblank, Prabhakar Raghavan, and Madhu Sudan. The minimum latency problem. *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 163–171, 1994.

[20] Zahy Bnaya, Ariel Felner, Dror Fried, Olga Maksin, and Solomon Eyal Shimony. Repeated-task canadian traveler problem. *AAAI Publications, Fourth Annual Symposium on Combinatorial Search*, pages 24–30, 2011.

[21] Allan Borodin and Yan El-Yaniv. *Online computation and competitive analysis.* Cambridge University Press, Cambridge, 1998.

[22] Scott Shorey Brown and Daniel H. Wagner. Necessary and sufficient conditions for optimal search plans for moving targets. *Mathematics of Operations Research*, 4:431–440, 1979.

[23] Scott Shorey Brown and Daniel H. Wagner. Optimal search for a moving target in discrete time and space. *Operations Research*, 28:1275–1289, 1980.

[24] Sabine Buttner and Sven O. Krumke. The canadian tour operator problem on paths: tight bounds and resource augmentation. *Journal of Combinatorial Optimization*, 32:842–854, 2016.

[25] K. Chaudhuri, B. Godfrey, S. Rao, and K. Talwar. Paths, trees, and minimum latency tours. *Proceedings of 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 36–45, 2003.

[26] Erik D. Demaine, Sandor P. Fekete, and Shmuel Gal. Online searching with turn cost. *Theoretical Computer Science*, 361:342–355, 2006.

[27] Erik D. Demaine, Yamming Huang, Chung-Shou Liao, and Kunihiko Sadakane. Canadians should travel randomly. *International Colloquium on Automata, Languages, and Programming*, pages 380–391, 2014.

[28] David Eppstein. Finding the k shortest paths. *Siam J. Computing*, 28:652–673, 1998.

[29] Amos Fiat and J. Gerhard Woeginger. *Online Algorithms: The State of the Art*. Springer, 1998.

[30] Matteo Fischetti, Gilbert Laporte, and Silvano Martello. The delivery man problem and cumulative matroids. *Operations Research*, 41:1055–1064, 1993.

[31] Dror Fried, Solomon Eyal Shimony, Amit Benbassat, and Cenny Wenner. Complexity of canadian traveler problem variants. *Theoretical Computer Science*, 487:1–16, 2013.

[32] Michel Goemans and Jon Kleinberg. An improved approximation ratio for the minimum latency problem. *Mathematical Programming*, 82:111–124, 1998.

[33] Patrick Jaillet and Xin Lu. Online traveling salesman problems with service flexibility. *Networks*, 58:137–146, 2011.

[34] Patrick Jaillet and Xin Lu. Online traveling salesman problems with rejection options. *Networks*, 64:84–95, 2014.

[35] Patrick Jaillet and Matthew Stafford. Online searching. *Operations Research*, 49:501–515, 2001.

[36] Anna R. Karlin, Mark S. Manasse, Larry Rudolph, and Daniel D. Sleator. Competitive snoopy caching. *Algorithmica*, 3:79–119, 1988.

[37] Kensaku Kikuta. A search game on a cyclic graph. *Naval Research Logistics*, 51:977–993, 2004.

[38] Dennis Komm. *An Introduction to Online Computation*. Springer, 2016.

[39] O. Bernard. Koopman. Search and screening. *OEG Repoert No. 56*, 1946.

[40] Sven O. Krumke. Online optimization: Competitive analysis and beyond, 2002. URL: `https://pdfs.semanticscholar.org/583b/4e88723d156931889c772c2dd98a71267c6d.pdf`.

[41] Chung-Shou Liao and Yamming Huang. The covering canadian traveler problem. *Theoretical Computer Science*, 530:80–88, 2014.

[42] Chung-Shou Liao and Yamming Huang. Generalized canadian traveller problems. *Journal of Combinatorial Optimization*, 29:701–712, 2015.

[43] Udo Loessner and Ingo Wegener. Discrete sequential search with positive switch cost. *Mathematics of Operations Research*, 7:426–440, 1982.

[44] Abilio Lucena. Timedependent traveling salesman problemthe deliveryman case. *Networks*, 20:753–763, 1990.

[45] Isabel Mendez-Diaz, Paula Zabala, and Abilio Lucena. A new formulation for the traveling deliveryman problem. *Discrete Applied Mathematics*, 156:3223–3237, 2008.

[46] Evdokia Nikolova and David Ron Karger. Route planning under uncertainty: The canadian traveller problem. *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, 2008.

[47] Laura H. Nunn. An introduction to the literature of search theory. *Operations Evaluation Group, Center for Naval Analyses*, 305, 1981.

[48] Christos Papadimitriou and Mihalis Yannakakis. Shortest paths without a map. *Theoretical Computer Science*, 84:127–150, 1991.

[49] O. Furkan Sahin and Vural Aksakalli. A comparison of penalty and rollout-based algorithms for the canadian traveler problem. *International Journal of Machine Learning and Computing*, 5:319–324, 2015.

[50] Sartaj Sahni and Teofilo Gonzales. P-complete problems and approximate solutions. *Proceedings of the IEEE conference record of symposium on switching and automata theory*, pages 28–32, 1974.

[51] Davood Shiri and Sibel Salman. On the online multi-agent o-d k-canadian traveler problem. *Journal of Combinatorial Optimization*, 34:453–461, 2017.

[52] Davood Shiri and Sibel Salman. Competitive analysis of randomized online strategies for the online k-canadian traveler problem. *Journal of Combinatorial Optimization*, 1:1–18, 2018. doi:10.1007/s1087.

[53] Davood Shiri and Sibel Salman. On the randomized online strategies for the k-canadian traveler problem. *Journal of Combinatorial Optimization*, 1:1–14, 2019. 10.1007/s10878-019-00378-1.

[54] David Simchi-levi and Oded Berman. Minimizing the total flow time of n jobs on a network. *IEE Transactions*, 23:236–244, 1991.

[55] Daniel Sleator and Robert Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28:202–208, 1985.

[56] Lawrence D. Stone. Theory of optimal search. *Academic Press*, Newyork, 1975.

[57] Lawrence D. Stone. Or forum- what's happened in search theory since the 1975 lanchester prize? *Operations Research*, 37:501–506, 1989.

[58] Bing Su and Yinfeng Xu. Online recoverable canadian traveler problem. *In: Proceedings of the international conference on management science and engineering*, pages 633–639, 2004.

[59] Bing Su, Yinfeng Xu, Peng Xiao, and Lei Tian. A risk-reward competitive analysis for the recoverable canadian traveller problem. *In: Proceedings of the second international conference on combinatorial optimization and applications*, pages 417–426, 2008.

[60] Alan R. Washburn. Search for a moving target: The fab algorithm. *Operations Research*, 31:739–751, 1983.

[61] Ingo Wegener. Optimal search with positive switch cost is np-hard. *Information Processing Letters*, 21:49–52, 1985.

[62] Xingang Wen, Yinfeng Xu, and Huili Zhang. Online traveling salesman problem with deadline and advanced information. *Computers and Industrial Engineering*, 63:1048–1053, 2012.

[63] Xingang Wen, Yinfeng Xu, and Huili Zhang. Online traveling salesman problem with deadlines and service flexibility. *Journal of Combinatorial Optimization*, 30:545–562, 2015.

[64] Stephan Westphal. A note on the k-canadian traveler problem. *Information Processing Letters*, 106:87–89, 2008.

[65] Bang Ye Wu, Zheng-Nan Huang, and Fu-Jie Zhan. Exact algorithms for the minimum latency problem. *Information Processing letters*, 92:303–309, 2004.

[66] Yinfeng Xu, Maolin Hu, Bing Su, Binhai Zhu, and Zhijun Zhu. The canadian traveler problem and its competitive analysis. *Journal of Combinatorial Optimization*, 18:195–205, 2009.

[67] Yinfeng Xu and Huili Zhang. How much the grid network and rescuers communication can improve the rescue efficiency in worst-case analysis. *Journal of Combinatorial Optimization*, 30:1062–1076, 2015.

[68] Andrew C. Yao. Probabilistic computations: Towards a unified measure of complexity. *Proceedings of the 18th Annual IEEE Symposium on the Foundations of Computer Science*, pages 222–227, 1977.

[69] Jin Yen. Finding the k shortest loopless paths in a network. *Management Science*, 17:712–716, 1971.

[70] Huili Zhang, Weitian Tong, Guohui Lin, and Yinfeng Xu. Online minimum latency problem with edge uncertainty. *European Journal of Operational Research*, 2018. DOI=10.1016/j.ejor.2018.08.017.

[71] Huili Zhang, Weitian Tong, Yinfeng Xu, and Guohui Lin. The steiner traveling salesman problem with online edge blockages. *European Journal of Operational Research*, 243:30–40, 2015.

[72] Huili Zhang, Weitian Tong, Yinfeng Xu, and Guohui Lin. The steiner traveling salesman problem with online advanced edge blockages. *Computers and Operations Research*, 70:26–38, 2016.

[73] Huili Zhang and Yinfeng Xu. Online covering salesman problem. *Journal of Combinatorial Optimization*, 35:941–954, 2018.

[74] Huili Zhang, Yinfeng Xu, and Lan Qin. The k-canadian travelers problem with communication. *Journal of Combinatorial Optimization*, 26:251–265, 2013.