2020

Essays In Optimization Methods For Resource Allocation

Markos Spyridon Epitropou
University of Pennsylvania

Essays In Optimization Methods For Resource Allocation

Abstract

This dissertation proposes and investigates the use of mathematical programming techniques to solve resource allocation problems that are typically handled using other techniques. This approach both simplifies proofs of earlier results as well as extends them.

The first setting addresses a network of agents, initially endowed with resources, exchanging goods and services via bilateral contracts. Under full substitutability of preferences, it is known via fixed point arguments that a competitive equilibrium exists in trading networks. I formulate the problem of finding an efficient set of trades as a generalized submodular flow problem in a suitable network. Existence of a competitive equilibrium follows directly from the optimality conditions of the flow problem. This formulation enables me to perform comparative statics with respect to the number of buyers, sellers, and trades. For instance, I establish that if a new buyer is added to the economy, at equilibrium the prices of all existing trades increase. In addition, a polynomial time algorithm for finding competitive equilibria in trading networks is given.

The second setting relates to dynamic resource allocation with the presence of uncertainty for future rewards. Prophet inequalities involve a set of results relating the reward attained in an on-line selection setting to the reward generated by a prophet possessing perfect information. I develop new, approximately efficient rules leveraging the reduced-form representation of on-line selection problems. I apply the method in an on-line mechanism design problem with verification and the on-line fractional knapsack selection problem.

Degree Type

Dissertation

Degree Name

Doctor of Philosophy (PhD)

First Advisor

Rakesh Vohra

Keywords

Flow Algorithms, Optimal Stopping, Prophet Inequalities, Resource Allocation, Trading Networks

Subject Categories

Computer Sciences | Operational Research

ESSAYS IN OPTIMIZATION METHODS FOR RESOURCE ALLOCATION

Markos Epitropou

A DISSERTATION

in

Electrical and Systems Engineering

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2020

Supervisor of Dissertation

Rakesh Vohra, Professor of Economics and Electrical and Systems Engineering

Graduate Group Chairperson

Victor M. Preciado, Associate Professor of Electrical and Systems Engineering

Dissertation Committee

Victor M. Preciado, Associate Professor of Electrical and Systems Engineering

Eduardo Azevedo, Associate Professor of Business Economics and Public Policy

Ozan Candogan, Associate Professor of Operations Management

Dedicated to Paris, Anthi, Panagiotis, Nikos, Anastasia, and Monique.

ACKNOWLEDGEMENT

This dissertation would not have been possible without the support of various parties on an academic and personal level. First of all, I want to pay special tribute to the Department of Electrical and Systems Engineering that provided a welcoming and highly academic environment to perform my doctoral studies. Thank you for making me part of a community of engineers excelling in multiple research areas and providing me with financial support to sustain my studies. Special thanks to the Defense Advanced Research Projects Agency of the US Department of Defense for supporting the last years of my studies.

Most importantly, I would like to thank Prof. Rakesh Vohra, who supervised my studies. I am immensely grateful for allowing me to engage in doctoral studies and the tireless effort to guide me through the process. I find myself extremely fortunate, having him as my academic advisor.

I want to add special thanks to Prof. Ozan Candogan for orchestrating a major part of this dissertation. His excitement and valuable discussions were extremely insightful and motivated me to pursue our research goals.

This academic journey was mostly wonderful because of the uncountable number of remarkable people that accompanied me along the way. It is hard to acknowledge all in a paragraph, but I would like to mention several friends that I feel fortunate to have spent time with and share experiences. Thank you, Leonidas, Luiz, Santiago, Melissa, Miguel, Venetia, and Pavlos for all the fun moments. I feel very fortunate to have made many new friends during my studies at the University of Pennsylvania: Yannis, Fragkiskos, Aris, Kostas, Tasos, Jing, Lucas, Fernando, Santiago, Aryan, Weiyu, Mark, Luana, Mahyar, Collin, Hideto, Bo. Last but not least, I would like to thank many friends from Greece that I have missed so much: Manolis, Vasilis, Thanasis, Thodoris, Yannis, Andreas, Kostis, Kostas, Fotis, Dimitris, Panos.

ABSTRACT

ESSAYS IN OPTIMIZATION METHODS FOR RESOURCE ALLOCATION

Markos Epitropou

Rakesh Vohra

This dissertation proposes and investigates the use of mathematical programming techniques to solve resource allocation problems that are typically handled using other techniques. This approach both simplifies proofs of earlier results as well as extends them.

The first setting addresses a network of agents, initially endowed with resources, exchanging goods and services via bilateral contracts. Under full substitutability of preferences, it is known via fixed point arguments that a competitive equilibrium exists in trading networks. I formulate the problem of finding an efficient set of trades as a generalized submodular flow problem in a suitable network. Existence of a competitive equilibrium follows directly from the optimality conditions of the flow problem. This formulation enables me to perform comparative statics with respect to the number of buyers, sellers, and trades. For instance, I establish that if a new buyer is added to the economy, at equilibrium the prices of all existing trades increase. In addition, a polynomial time algorithm for finding competitive equilibria in trading networks is given.

The second setting relates to dynamic resource allocation with the presence of uncertainty for future rewards. Prophet inequalities involve a set of results relating the reward attained in an on-line selection setting to the reward generated by a prophet possessing perfect information. I develop new, approximately efficient rules leveraging the reduced-form representation of on-line selection problems. I apply the method in an on-line mechanism design problem with verification and the on-line fractional knapsack selection problem.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

CHAPTER 1 : Introduction

The best use of scarce resources is a primary objective of economic activity and has many different flavors depending on the circumstances the various decisions are taken. Instantiations of the resource allocation problem include production planning, scheduling, portfolio selection, ad allocation, and labor markets. Mathematical programming frameworks have been widely used to address resource allocation problems by providing well-defined descriptions of the relevant problem. An objective function models the notion of efficiency, and a constraint set formalizes the scarcity of resources. An algorithm for solving a mathematical program can serve as a decision-making strategy to allocate the relevant scarce resources. However, the challenges inherent in resource allocation problems expand way beyond a simple notion of efficiency. For a variety of instances, there are concerns related to the design of information structures, the implementation of decision-making procedures under strategic behavior, fairness, and privacy.

This research will focus on two settings related to resource allocation and how they can be addressed using a carefully chosen optimization framework. The first setting addresses a network of agents, initially endowed with resources, exchanging goods and services, via bilateral contracts. The second setting relates to dynamic resource allocation with the presence of uncertainty for future rewards. This research examines the relevant optimization frameworks in order to provide qualitative insights for each setting.

1.1. Trading Networks: A Network Flow Approach

The terms of trade for the exchange of goods and services are often set via bilateral contracts. For instance, in labor markets, firms contract with individual employees who offer their labor and time. In supply chains, manufacturers bilaterally contract with their suppliers to procure materials needed for the production of their goods. Service firms, such as those providing tax and consulting services, (commercial as well as passenger) transportation, and construction use bilateral contracts with their clients to specify the scope of their projects as well as the payment for the service they provide.

Trading networks, introduced in Hatfield et al. (2013), are a way to represent the non-price aspects of bilateral trades such as quantity, quality, delivery dates, and returns policy. The vertices of the underlying network correspond to agents, or *traders*, who buy and/or sell goods or services. Each arc represents the non-price aspects of a bilateral trade between the agents corresponding to the incident vertices. The orientation of the arc identifies which agent is the "buyer" and which the "seller" in that particular trade. Agents have quasilinear preferences over the set of incident arcs and their corresponding prices. Preferences need not be additive over the trades and, in general, can exhibit a fairly rich combinatorial structure. For instance, in a supply chain, a firm is unable to sell goods to its downstream buyer without first procuring necessary materials from upstream suppliers – a feature that can be encoded by appropriately specifying agents' preferences. A central question is whether there exist prices for the trades that "clear" the market, i.e., competitive equilibria.

The classic assignment model of Shapley and Shubik (1971) is a special case. Each agent is either a buyer or a seller, but not both. Furthermore, no agent can participate in more than one trade. In this setting the underlying network is bipartite. The existence of market-clearing prices is a consequence of linear programming duality. By contrast, trading networks are rich enough to model an agent who takes the "buy" side in some trades and the "sell" side in others. This is an important feature, for instance, in supply chain networks, where firms simultaneously participate as buyers of their production inputs and sellers of their finished products. In supply chain networks, the underlying directed network is often acyclic. Under this condition, Ostrovsky (2008) establishes[1] the existence of competitive equilibria.

Acyclicity of trading networks is violated if there is a market for resale of goods. For example, in the used car sales market, some individuals participate as buyers and others participate as sellers, and dealers trade with both buyers and sellers of used cars and with each other (Hatfield et al., 2013); hence the underlying trading network involves directed

[1] In fact, it is shown that existence obtains under preferences more general than quasilinear.

cycles. Similar directed cycles are present in financial markets, where financial institutions may simultaneously buy securities from some institutions while selling them to others. Finally, firms offering different services can also induce cycles, e.g., a firm offering consulting services to an audit firm while also receiving audit services for its operations.

Hatfield et al. (2013, 2019a) have shown that under a *full substitutability* condition on agents' preferences, a competitive equilibrium exists in trading networks, whether acyclic or not. The full substitutability condition generalizes the *gross substitutability* condition used to establish the existence of a competitive equilibrium in two-sided markets (Kelso and Crawford, 1982; Gul and Stacchetti, 1999; Sun and Yang, 2006). Thus, the trading network model extends the competitive equilibrium existence results to multi-sided settings. Competitive equilibria in trading networks are also *stable outcomes* in that they cannot be blocked by any coalition of agents and trades. A *blocking set* is a set of (feasible) trades and corresponding prices such that all agents who can participate in these trades prefer them (while possibly declining some of their equilibrium contracts) (Hatfield et al., 2013). Conversely, in any stable outcome, it is possible to price the trades not involved in this outcome to support the stable outcome as a competitive equilibrium. In fact, the stability condition is equivalent to the seemingly weaker *chain stability* condition (Hatfield et al., 2019b). The latter condition restricts blocking sets to be paths/cycles of trades in the underlying trading network.

A first contribution is to show that under the full substitutability assumption, all these results can be obtained simply and directly from the optimality conditions of a *generalized submodular flow problem* in a suitable network. An optimal flow corresponds to competitive equilibrium trades, and its optimal dual solutions (potential values) are supporting prices. Moreover, in generalized submodular flow problems, a feasible flow is optimal if and only if there is no *improvement cycle*. This optimality condition yields the equivalence between a competitive equilibrium outcome and (chain) stability. As a result, it is possible to obtain polynomial algorithms for finding a competitive equilibrium, testing whether a given payoff vector can be supported in a competitive equilibrium, as well as identifying a blocking chain when an outcome is not stable. In addition, I exploit the connection to the submodular flow problem to give new comparative statics on the behavior of equilibrium prices as the set of buyers, sellers, and trades changes.

The starting point is to express the problem of identifying the set of trades that maximize welfare as a network flow problem in an appropriately defined *flow network*. The flow network is related to, but distinct from the underlying trading network. In the flow network, each vertex corresponds to an agent-trade pair of the trading network. Since exactly two agents are involved in each trade, the flow network has two vertices for each feasible trade (one associated with the buyer and the other associated with the seller). These vertices are connected by an arc in the flow network, though, the flow network itself need not be connected.

Full substitutability of agents' preferences corresponds to $M^\natural$-concavity of agents' value functions (Hatfield et al., 2019a). This observation allows us to represent the problem of finding the set of welfare-maximizing trades as a generalized submodular flow problem in the flow network. In doing so, I do not impose flow conservation at each vertex. Instead, I impose an M-convex penalty term on the net outflow at vertices associated with the same agent in the flow network. Intuitively, the net outflow encodes the trades in which an agent participates as a buyer/seller, and the penalty term captures the total value the agent gains from these trades. Minimum cost flows in this network correspond to trades in the original network that maximize total welfare. The optimal dual solutions to this problem are competitive equilibrium prices that support the welfare-maximizing set of trades. Thus, the approach generates the equilibrium trades and prices through the solution of an optimization problem. This is in contrast to Hatfield et al. (2013), who construct an auxiliary two-sided market and invoke the associated competitive equilibrium existence results of Kelso and Crawford (1982). These existence results rely on both discrete prices and explicitly constructing a price (salary) adjustment process that converges to prices that complement an efficient set of trades. My approach relaxes the discreteness requirement on prices and skips the reduction to Kelso and Crawford (1982), and leverages duality results to establish the existence of a competitive equilibrium.

I establish the equivalence between stability, chain stability, and competitive equilibrium outcomes directly from the fact that a given flow is optimal if and only if it admits no improvement cycle. The proof technique used also provides an algorithm that (i) checks whether an outcome is (chain) stable, and (ii) identifies a blocking chain if it is not. In particular, given a set of trades and associated prices, I first consider a (reduced) trading network, which consists of the remaining trades (after an appropriate modification of the payoff functions), and the corresponding flow network. The algorithm starts with the (trivial) flow, which does not use any arc in the flow network that is associated with the trades in the (reduced) trading network. Then, the algorithm searches for an improvement cycle. If such a cycle is absent, I conclude that the initial set of trades/prices constitutes a (chain) stable outcome. Otherwise, an improvement cycle with the least number of arcs reveals a blocking chain. The computational complexity of this approach is equivalent to that of constructing the flow network and identifying a negative cycle with the least number of arcs in this network. The overall complexity is polynomial in the number of vertices/arcs in the underlying trading network. Thus, the network flow approach presented here not only gives simpler proofs of the properties of trading networks (e.g., existence of a competitive equilibrium, and its equivalence to stability), but also provides a tractable algorithm for determining competitive equilibria, testing whether a given payoff vector can be supported in a competitive equilibrium, testing (chain) stability, and identifying blocking sets of trades whenever they exist.

Hatfield et al. (2019b) observed an equivalence between stability and chain stability that resembles an analogous equivalence in classic network flows. Those authors argued that there are important differences between the two settings:

> “[I]n the ‘network flows’ environment, there is a single type of good ‘flowing’

> through the network, and the objective function is the maximization or minimization of the aggregate flow, whereas in our setting many different types of goods may be present, and the preferences of agents in the market may be more complex."

My work shows this difference to be superficial. An outcome is not stable if the corresponding flow is suboptimal. In the generalized submodular flow problems, suboptimality implies the existence of an improvement cycle. This indicates that whenever the initial outcome is not stable, it can be blocked by relying on a "simple" set of trades, which corresponds to a chain in the underlying trading network.

This optimization approach allows us to determine how competitive equilibria change if a new trade, buyer, or seller is added to the economy. It is shown that if a new trade is added to the economy, the new equilibrium trades can be found by augmenting the existing trades with a set of trades that form an undirected chain in the trading network, i.e., by (i) including among equilibrium trades the trades of this chain that do not belong to the initial equilibrium, and (ii) removing the remaining trades associated with the arcs in the chain from the set of the equilibrium trades. Adding a new buyer (seller) to the economy is equivalent to adding a collection of such trades, and hence new equilibrium trades can be found by repeatedly augmenting the existing ones. In addition, it is shown that if a new buyer (seller) is added to the economy, the prices of all existing trades, even those of traders not adjacent to the new agent, increase (decrease). Hence, the equilibrium payoffs of all existing buyers decrease (increase), and those of all sellers increase (decrease). Using these results as building blocks, I provide comparative statics for richer settings where the new trades are not all adjacent to a single trader (and can involve multiple buyers/sellers). I also outline how to incorporate trade frictions into the model, and generalize the results on competitive equilibria and comparative statics. Finally, I discuss applications of the comparative statics to changes in trade frictions (e.g., excise taxes or transportation costs) as well as to quotas in trading networks.

I summarize the related literature below. Trading networks, the M-convex submodular flow problem, and their relation are introduced in Chapter 2. Applications of this approach are presented in Chapter 3.

Related literature: Gross substitutability of agents' preferences is a sufficient condition for the existence of a competitive equilibrium in settings with indivisible goods (Kelso and Crawford, 1982; Gul and Stacchetti, 1999). It is equivalent to $M^\natural$-concavity of agents' value functions (Fujishige and Yang, 2003; Murota and Tamura, 2003a; Leme, 2017; Shioura and Tamura, 2015). It has been used in mathematical economics to generate direct proofs of the existence of competitive equilibria in exchange economies (Danilov et al., 2001, 2003; Murota and Tamura, 2003a,b). For a survey see Murota (2016).

Kelso and Crawford (1982) and Gul and Stacchetti (1999) were concerned with a two-sided market of buyers and sellers. The trading network literature (Ostrovsky, 2008; Hatfield et al., 2013, 2019a,b) extends these results to multi-sided settings, where agents can simultaneously participate as buyers and sellers in the market. This is done by extending the gross substitutes condition on preferences[2] to full substitutability. Full substitutability of agents' preferences corresponds to $M^\natural$-concavity of agents' value functions (Hatfield et al., 2019a).

In Murota (2003) and Murota and Tamura (2003b), the problem of finding the efficient allocation in a two-sided economy with multiple buyers and sellers was formulated as a generalized submodular flow problem in a bipartite network. In our case, the presence of agents who participate as buyers in some trades and sellers in others makes the reduction in Murota (2003) and Murota and Tamura (2003b) inapplicable. I provide an alternative network flow formulation for identifying the efficient set of trades. Additionally, this formulation shows the equivalence of competitive equilibrium to (chain) stable outcomes and characterizes them using a generalized submodular flow formulation. Thus, together with

[2]Some papers, e.g., Ostrovsky (2008) and Hatfield et al. (2019b), relax the assumption of quasilinear preferences.

the results of Murota (2003) and Murota and Tamura (2003b), this work indicates that a generalized submodular flow formulation provides a unifying framework for the study of various competitive equilibrium results in the literature.

Prior to this dissertation, Ikebe and Tamura (2015) and Ikebe et al. (2015) also used ideas from discrete convexity to study trading networks. Ikebe and Tamura (2015) focused on acyclic trading networks (called supply chain networks) under $M^\natural$-concavity of agents' value functions. They allowed trades between agents to occur with some (integer) intensity, capturing the fact that multiple units of the same trade can take place. They also provided algorithms for finding chain stable outcomes. Importantly, the authors focused on a setting where there are finitely many contracts (trade and price tuples). This implies that the supporting prices are exogenously restricted to a finite set. Given the restriction to finitely many contracts and acyclic networks, the algorithms provided in their paper for testing chain stability do not apply to my setting. Finally, this dissertation provides comparative statics and sheds light on the equivalence between stability, chain stability, and the competitive equilibrium outcome.

Ikebe et al. (2015) focused on general trading networks, where, as in Ikebe and Tamura (2015), agents may engage in multiple units of the same trade. In their setting trades between different pairs of agents are considered distinct. In particular, trades that represent two distinct sellers selling the same commodity to a common buyer are considered distinct. Under these restrictions, the authors established the existence of a competitive equilibrium where all identical trades have the same price. This result is related to but weaker than the one presented in Section 2.3.2 below. When identical trades are defined I do not restrict attention to pairs of agents. Hence, I allow for settings where the same commodity is sold by different sellers to a common buyer. I establish the existence of competitive equilibria where all identical trades adjacent to an agent have the same price.

Ikebe et al. (2015) employed a definition of stability that differs from the one used here along two dimensions. First, when a trade is included in a blocking set, the prices of other identical trades (which belong to the original outcome) are allowed to change as well. Second, agents are not allowed to drop old trades if identical trades are included in the blocking set. Under this alternative definition of stability, Ikebe et al. (2015) showed that a stable outcome need not correspond to a competitive equilibrium. By contrast, I establish an equivalence between the two outcomes.

Like us, Ikebe et al. (2015) exploited properties of discrete convexity to derive their results. However, they did not obtain the network flow formulation, which allows us to provide short proofs of the equivalence of various solution concepts. The network flow formulation is conceptually important because it explains why chain stability is a natural solution concept for trading networks in the first place. As in the classic network flow problem, if a flow is suboptimal (i.e., the corresponding set of trades is inefficient), it is always possible to find an improvement cycle, which in the context of my network flow formulation closely relates to a blocking chain. Furthermore, leveraging network flow ideas allows us to develop efficient algorithms for identifying blocking chains and provide interesting comparative statics, both of which are beyond the scope of Ikebe et al. (2015).

One can exploit gross substitutability to derive a tâtonnement that converges to a competitive equilibrium. In two-sided markets, Ausubel (2006) provided such a tâtonnement process, similar in spirit to the one in Gul and Stacchetti (2000). Sun and Yang (2009) analyzed a tâtonnement process called double-track, which converges to a competitive equilibrium for the case of substitutes and complements. The double-track procedure was generalized to the case of multiple complementary goods in Sun and Yang (2014). Using the network formulation of the efficient allocation problem, I outline a similar tâtonnement process for trading network models.

Comparative statics have received significant attention in the matching and trading network literature (Kelso and Crawford, 1982; Blum et al., 1997; Hatfield and Milgrom, 2005; Ostrovsky, 2008; Fleiner et al., 2018). The network flow literature contains a rich set of sensitivity analysis results on how optimal flows and corresponding potential values change as arc capacities, costs, and supply/demand at different vertices change (Granot and Veinott, 1985; Ahuja et al., 1993). By extending these results to my network flow formulation, I obtain new results on how the equilibrium trades, prices, and payoffs change as new trades/traders are introduced into the economy.

1.2. Prophet Inequalities in Resource Allocation Under Uncertainty

Stopping problems are concerned with choosing a time to take a given action based on sequentially observed random variables in order to maximize an expected payoff. The action taken may be to reject a hypothesis, replace a machine, hire a secretary, or exercise an American option. They are often solved using dynamic programming.

In a simple version of a stopping problem, introduced by Krengel and Sucheston (1977), one is shown n non-negative numbers, sequentially, that are independent draws from known distributions (not necessarily identical). I refer to it as the *basic stopping problem* hereafter. The goal is to maximize the number on which one stops relative to the expected maximum value in hindsight. Krengel and Sucheston (1977) obtained a tight approximation guarantee of $\frac{1}{2}$. In other words, the optimal reward of the stopping problem is at least half of the expected value of the largest of the n random numbers. This approximation guarantee is usually referred to as a *prophet inequality.*

A simple example shows that the approximation factor of $\frac{1}{2}$ is the best possible. There are two prizes, where the first one gives a reward of 1, while the second one gives a reward of $\frac{1}{\epsilon}$ with probability ϵ or 0 otherwise. Both strategies of stopping or not stopping in the first period return the same expected reward equal to 1. However, a prophet could solve an off-line version of the problem, and choose the second prize only when a reward of ϵ is realized, i.e., with probability $\frac{1}{\epsilon}$. Hence, the expected prophet's reward is equal to $2 - \epsilon$. The on-line decision-maker guarantees $\frac{1}{2-\epsilon}$ of the prophet's reward. Thus, $\frac{1}{2}$ is the approximation factor that most algorithms attempt to attain unless the structure of the problem is more welcoming.

The problem has attracted an enthusiastic following which has extended the problem in a variety of ways. Hill and Kertz (1992) provide an early survey, and Hartline (2012) and Lucier (2017) provide surveys discussing the implications of prophet inequalities for mechanism design and pricing. Part of the literature considers *on-line selection problems*, a series of problems where the selected elements must satisfy combinatorial constraints. Alaei (2011) considers an extension where k numbers can be chosen and Kleinberg and Weinberg (2019) deal with a general setting, where the set of numbers chosen must form an independent set of a specified matroid.

Interestingly, research on prophet inequalities finds applications in mechanism design. Apart from the efficiency guarantees, several proofs reveal simple rules for well-known mechanism design problems. Samuel-Cahn (1984) provided a simple proof of the prophet inequality, where the *median* of the largest order statistic of n numbers is chosen as a threshold to decide which number to choose. Chawla et al. (2010) establish a connection with mechanism design, by showing that this threshold can serve as a posted price in an auction setting. Recently, Duetting et al. (2017) covered a unifying technique for general settings which is approximately efficient and operates by producing *balanced prices*.

This dissertation follows a different path by modeling the prophet problem via linear programming. The modeling process is closely related to works in mathematical finance where the decisions made are conditioned on the paths of the relevant stochastic process generating the rewards. The main contributions of this part of the dissertation is a new representation of the stopping problem, an alternative viewpoint of the prophet inequality, as well as applications in on-line selection problems with complex constraints.

Initially, the basic stopping problem is considered and its application in auction theory. First, a naive linear programming formulation is presented. Unfortunately, the linear programming representation of the problem exploits a set of variables conditioned on the possible paths of the reward generation stochastic process, which exponentially grows on the number of agents. As a remedy, the reduced-form representation of the basic stopping problem is introduced. The reduced-form representation is a tractable description of the stopping problem, where a polynomial number of variables is used. The reduced-form representation has been employed in auction theory before, mainly through the work of C Border (1991), with the main contribution known as *Border's theorem*. The variables in Border's theorem are called *interim allocation*, representing the expected allocation to an agent given her type. The same idea applies to the optimal stopping setting, from an auction-based viewpoint. Using this new representation, I devise a new algorithm which is approximately efficient, along the lines of the classic prophet inequality. This constitutes a new proof of the classic prophet inequality.

A first application involves a mechanism design problem with verification. The problem can be thought in the context of labor markets. I consider a principal who must hire an employee from a set of candidates arriving sequentially, each of whom prefers to have the job than not. Each agent has access to private information about the principal's payoff if he gets hired. The decision to allocate the job to an employee must be made upon his arrival and is irreversible. There are no monetary transfers but the principal can verify agents' reports at a cost and punish them (by not hiring them). The optimal allocation and verification rules are given as a solution to a compact linear program. There exists a strategy that can achieve on expectation at least half of the second-best of a prophet, i.e., when all agents arrive simultaneously but still the principal has to elicit truthful reports via verification.

Last, I provide a generalized version of on-line selection problems and a linear programming formulation addressing the problem. The formulation is also valid for simple on-line selection problems since the linear programming formulation for their continuous counterparts has integer optimal solutions. I will consider a setting based on the fractional knapsack selection problem and provide an algorithm that scales the prophet's reduced form by $\frac{1}{2}$ and then implements it. I prove that it is well-defined for the fractional knapsack selection setting, i.e., the interim allocation that it constructs is implementable, and trivially achieves a $\frac{1}{2}$-approximation of the prophet's rewards (linear in interim allocation variables). It is worth noting that the above algorithm details a way to convert a solution of the classic allocation problem to a solution when the actions must be taken sequentially. Finally, we see that the linear programming formulation captures simple on-line selection problems too, i.e., problems defined in a combinatorial domain. For simple settings, the prophet inequalities attained are the same, regardless of whether the domain is continuous or combinatorial.

In Chapter 4 I introduce the basic stopping problem. I then describe three different techniques to derive the classic prophet inequality. The techniques are based on different algorithms for the stopping problem. First, I describe pricing strategies, where a reward is chosen when it is larger than an a priori defined threshold, which can be thought of as a price paid in an auction setting where bidders arrive sequentially. The pricing schemes come from different works found in the literature. Second, I describe a duality-based technique to derive the prophet inequality. The argument examines the optimal stopping rule and is based on the celebrated work of Davis and Karatzas (1994) in mathematical finance. I include this proof to better illustrate duality in prophet inequalities. Last, I formulate the optimal stopping problem as a linear program. First, I provide a proof of the prophet inequality using an appropriate price, based on Guha and Munagala (2007). The prophet inequality leverages a relaxed linear programming formulation of the stopping problem using the reduced form. Next, i describe the reduced-form representation of the optimal stopping problem. Leveraging this representation, I present the algorithm based on scaling the reduced form of the prophet's strategy.

In Chapter 5 I describe applications of some of the above techniques. Specifically, I introduce the on-line verification problem and describe a linear programming formulation of it. The linear program implies an algorithm to solve it as well as a direct argument to lower bound its efficiency. Finally, in Chapter 6, I introduce the fractional knapsack problem and provide a novel application of it in computational sprinting. The above technique of scaling the reduced form is used to derive a new prophet inequality for the fractional knapsack problem. Finally, I end the chapter by showing that the formulation for the fractional version of the problem, in simple settings, carries on to settings with integrality constraints.

CHAPTER 2 : Trading Networks through Network Flows

The subsequent Section 2.1 introduces notation and the model. Section 2.2 describes the submodular flow problem and its optimality conditions. Section 2.3 describes the transformation of the problem of finding an efficient set of trades into an instance of the submodular flow problem.

2.1. Model

A trading network is represented by a directed multigraph $G = (N, E)$, where N is the set of vertices and E the set of arcs. Each vertex corresponds to an agent and each arc corresponds to the non-price elements of a trade that can take place between the incident pair of vertices. For each arc $e \in E$, let e^+ and e^- denote the tail and head of this arc, respectively. Vertex e^+ corresponds to the seller and vertex e^- corresponds to the buyer of the trade associated with e. Let $\delta_+(i)$ and $\delta_-(i)$ respectively be the outgoing and incoming arcs incident to vertex $i \in N$, and set $\delta(i) = \delta_+(i) \cup \delta_-(i)$. A price vector is dentoed by $p \in \mathbb{R}^E$, where p_e is the price associated with the trade that corresponds to arc e. Denote by p^X the price vector restricted to the arcs in X.

Denote agent i's value function for any set of trades involving agent i by[1] $w_i : 2^{\delta(i)} \to \mathbb{R} \cup \{-\infty\}$. Agent i's payoff function is $u_i : 2^{\delta(i)} \times \mathbb{R}^{\delta(i)} \to \mathbb{R} \cup \{-\infty\}$. For each $S \subset \delta(i)$ and $p \in \mathbb{R}^E$, the agent's payoff can be expressed as:

$$u_i(S, p) = w_i(S) + \sum_{e \in S \cap \delta_+(i)} p_e - \sum_{e \in S \cap \delta_-(i)} p_e.$$

The demand correspondence for agent $i \in N$, given a price vector $p \in \mathbb{R}^{\delta(i)}$, is

$$D_i(p) = \arg \max_{Y \subset \delta(i)} u_i(Y, p).$$

[1] Having $-\infty$ in the range of the value function allows us to incorporate trading constraints as described in Hatfield et al. (2013); e.g., if a trader cannot sell goods without procuring her inputs first, this can be incorporated by specifying $-\infty$ for bundles of trades.

Definition 2.1.1. A set of trades $X \subset E$ along with a price vector $p \in \mathbb{R}^E$ is a *competitive equilibrium* (X, p) if, for all $i \in N$,

$$X \cap \delta(i) \in D_i(p).$$

Definition 2.1.2. A set of trades $X \subset E$ is *efficient* if

$$X \in \arg\max_{S \subset E} \sum_{i \in N} w_i(S \cap \delta(i)).$$

2.2. M-convex Submodular Flow Problem

Here I describe the M-convex submodular flow problem that generalizes the classic network flow problem (see Chapter 9 in Murota (2003)). A directed graph $F = (V, A)$ is given, where V is the set of vertices and A is the set of arcs. For each arc $a \in A$, denote by a^+ and a^- the tail and head vertices respectively. For each $v \in V$ denote by $\delta_+(v)$ and $\delta_-(v)$ the set of outgoing and incoming arcs incident to vertex v, respectively.

As in the classic network flow problem, each arc $a \in A$ has a cost $c_a \in \mathbb{R}$ per unit of flow, and lower and upper capacities $\underline{k}_a \in \mathbb{Z} \cup \{-\infty\}$, $\overline{k}_a \in \mathbb{Z} \cup \{\infty\}$. Denote by x_a the amount flowing through $a \in A$, and let x denote the vector of $\{x_a\}_{a \in A}$. Given flows on the arcs, let y_v be the net outflow (positive or negative) from vertex v, and denote the vector of $\{y_v\}_{v \in V}$ by y. In what follows I focus on integer flow problems, where x, y have integer entries.

The added feature of the M-convex submodular flow problem (MSFP) is a function $f : \mathbb{Z}^V \to \mathbb{R} \cup \{\infty\}$ in the objective function that penalizes the net outflow at each vertex. The function f is assumed to be M-convex (defined below). MSFP[2] can be formulated as follows:

[2]This version of MSFP is called the *M-convex submodular integer flow problem*, due to the integrality constraint on (x, y). As in the classic network flow problem, under mild conditions, the integrality condition is without loss of optimality; i.e., even when the problem is formulated over the reals, an integral optimal solution exists. See Remark 2.2.

$$\begin{aligned}
\min_{x \in \mathbb{Z}^A, y \in \mathbb{Z}^V} & \sum_{a \in A} c_a x_a + f(y) \\
\text{s.t.} & \sum_{a \in \delta_+(v)} x_a - \sum_{a \in \delta_-(v)} x_a = y_v \quad \forall v \in V, \\
& \underline{k}_a \leq x_a \leq \overline{k}_a \quad \forall a \in A.
\end{aligned}$$

To define M-convexity, let $\chi^j \in \mathbb{Z}^n$ denote the 0-1 vector with exactly one nonzero entry in component j. A function $f : \mathbb{Z}^n \to \mathbb{R} \cup \{\infty\}$ on the integer lattice is M-convex if it satisfies the following exchange axiom:

(M-EXC[$\mathbb{Z}$]): For all $z, z' \in \mathbb{Z}^n$ and for all $u \in supp^+(z - z')$,

$$f(z) + f(z') \geq \min_{v \in supp^-(z-z')} f(z - \chi^u + \chi^v) + f(z' + \chi^u - \chi^v),$$

where $supp^+(z - z')$ ($supp^-(z - z')$) is the set of indices in $\{1, \ldots, n\}$ such that $z_i - z'_i > 0$ ($z_i - z'_i < 0$). A function f is called M-concave if $-f$ is M-convex.

A set $B \subset \mathbb{Z}^n$ is called M-convex if given $z, z' \in B$, for all $u \in supp^+(z - z')$, there exists $v \in supp^-(z - z')$ such that $z - \chi^u + \chi^v \in B$, and $z' + \chi^u - \chi^v \in B$. An M-convex function f's *effective domain* (i.e., $\text{dom} f \triangleq \{z \in \mathbb{Z}^n | -\infty < f(z) < \infty\}$) as well as the function's minimizers are M-convex sets.

In the classic network flow problem, f's effective domain is a single point ($\{y^0\}$), which allows for a fixed amount of net inflow (outflow) at demand (supply) vertices satisfying $\{i | y_i^0 < 0\}$ ($\{i | y_i^0 > 0\}$), while imposing flow conservation at the rest of the vertices (i.e., the inflow to a vertex is equal to the outflow). In MSFP, the flow conservation constraints are relaxed through the use of the penalty function f. In the next section, when I study trading networks, I formulate the problem of finding the efficient set of trades as an MSFP. In this formulation, a unit of flow from a vertex associated with agent i to a vertex associated with agent j represents the trade between these agents (where i is a seller and j is a buyer), and the associated valuations of agents is encoded by the appropriately defined net outflow penalties. Note that in the efficient set of trades, an agent may participate in more trades as a buyer than as a seller (or vice versa). Also, the valuations of agents need not be additive over the trades they participate in. As a result, it is impossible to formulate the problem of finding the efficient set of trades in trading networks as a classic network flow problem. However, as I illustrate in Section 2.3, the aforementioned problem can be formulated as an MSFP with appropriately constructed penalties.

One can generalize the optimality conditions of the classic flow problem with a linear objective function to MSFP (see Murota (2003)). In particular, the optimality of a flow is characterized by the nonexistence of a negative cycle in an auxiliary network as well as in terms of a set of potential values associated with the vertices of the network.

Before I state the optimality conditions, some necessary definitions are introduced. First, define an auxiliary network F^{aux}, which is an extension of the idea of the residual network used in the classic network flow problem to account for the non-linearities in f. Let x be a feasible flow in $F = (V, A)$ and y be the associated vector of net outflows at each vertex. Consider three sets of arcs incident to the set of vertices V:

1. $A^{aux}(x) = \{(u, v) | (u, v) \in A, x_{(u,v)} < \overline{k}_{(u,v)}\}$,

2. $B^{aux}(x) = \{(v, u) | (u, v) \in A, x_{(u,v)} > \underline{k}_{(u,v)}\}$, and

3. $C^{aux}(y) = \{(u, v) | u, v \in V, f(y - \chi^u + \chi^v) < +\infty\}$.

Let $F^{aux}(x, y) = (V, A^{aux}(x) \cup B^{aux}(x) \cup C^{aux}(y))$ be the directed multigraph where all three sets of arcs are present. Note that, in general, F^{aux} can be a directed multigraph. However, in this work, I focus on settings where F is a directed graph that has at most one arc between any pair of vertices, and the effective domain of f is such that the arcs in C^{aux} do not overlap with those in A^{aux} and B^{aux}. As a result, F^{aux} is always a simple directed graph. The auxiliary network F^{aux} has no arc capacities. The cost for each arc a

in $F^{aux}(x,y)$ is given by

$$c_a^{aux}(x,y) = \begin{cases} c_{(a^+,a^-)} & \text{if } a \in A^{aux}(x) \\ -c_{(a^-,a^+)} & \text{if } a \in B^{aux}(x) \\ f(y - \chi^{a^+} + \chi^{a^-}) - f(y) & \text{otherwise.} \end{cases} \tag{2.1}$$

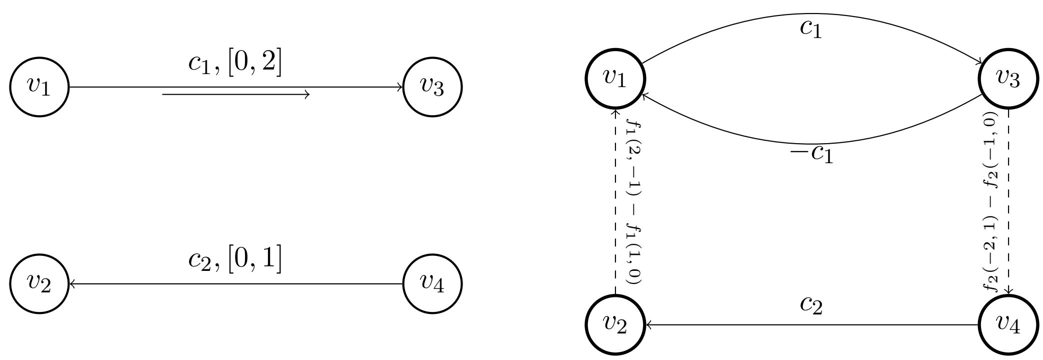

Figure 1: (a) Feasible Flow (b) Auxiliary Network

Figure 1 illustrates the construction of the auxiliary network. The flow network is displayed in (a). The penalty function is given by $f_1(y_1, y_2) + f_2(y_3, y_4)$, where y_i denotes the net outflow at vertex v_i. The effective domain of f_1 is $\{(1,0),(2,-1)\}$ and that of f_2 is[3] $\{(-1,0),(-2,1)\}$. Function f_1 penalizes the net outflow at vertices $\{v_1, v_2\}$ while f_2 penalizes the net outflow at $\{v_3, v_4\}$. The functions $f_1 : \{(1,0),(2,-1)\} \to \mathbb{R}$, and $f_2 : \{(-1,0),(-2,1)\} \to \mathbb{R}$ are specified such that $f_1(y_1,y_2) + f_2(y_3,y_4)$ is M-convex over its effective domain. The upper arc capacities in this example are 2 units for the first arc and 1 unit for the second arc, and the lower arc capacities are zero for both arcs. Suppose that one unit of flow is sent on the first arc, as denoted by the arrow below the arc. The corresponding auxiliary network is displayed in (b). The arcs in $A^{aux}(x) \cup B^{aux}(x)$ appear as solid arrows while the arcs in $C^{aux}(y)$ appear as dashed arrows.

In classic network flow problems, $f(y) = \infty$ unless $y = y^0$ for some y^0, which allows for a fixed net inflow/outflow for a subset of vertices, while imposing flow conservation at the

[3] Any functions f_1 and f_2 that take finite values at $\{(1,0),(2,-1)\}$ and $\{(-1,0),(-2,1)\}$, respectively, and take a value of ∞ elsewhere, satisfy M-EXC[$\mathbb{Z}$] and are M-convex.

rest of the vertices. Consequently, $C^{aux}(y) = \emptyset$, and hence the auxiliary network reduces to the residual network that is standard in the network flow literature. By contrast, in MSFP the auxiliary network involves additional arcs whose costs are determined by the associated outflow penalty. Consider the classic network flow problem given in Figure 1, where it is assumed that $y^0 = (1, 0, -1, 0)$. Note that the only feasible solution is to send one unit of flow on the upper arc and zero unit of flow on the lower arc. Then, the auxiliary network consists of the pair of arcs between v_1 and v_3, as well as the single arc from v_4 to v_2. By contrast, in more general MSFP formulations, depending on the penalty function f, one or both of the arcs in the network may have nonzero flow in the optimal solution. In this case, flow conservation may not be preserved, but the effect of the net outflows is still reflected in the penalty function f. Moreover, as discussed above, the auxiliary network may involve arcs from $C^{aux}(y)$.

The sum of the arc costs associated with a directed path/cycle of the auxiliary network is interpreted as the "length" of the path/cycle. The distance from a vertex to another vertex is defined as the smallest length achieved by a directed path connecting them. Any path achieving this distance is referred to as the shortest path. A directed cycle of negative length is called a *negative cycle*. The following theorem summarizes the optimality criteria for MSFP.

Theorem 2.2.1. (Theorems 3.1 and 3.2 in Murota (1999)) Given a feasible solution (x, y) to an MSFP, the following three conditions are equivalent for the MSFP:

1. (x, y) is an optimal solution to the MSFP.

2. There does not exist a negative cycle in $F^{aux}(x, y)$.

3. There exists a potential function $\pi : V \to \mathbb{R}$ such that

   (a) for each $(u, v) \in A$,

      (i) $c_{(u,v)} + \pi(u) - \pi(v) > 0 \Rightarrow x_{(u,v)} = \underline{k}_{(u,v)}$

(ii) $c_{(u,v)} + \pi(u) - \pi(v) < 0 \Rightarrow x_{(u,v)} = \overline{k}_{(u,v)}$

(b) $f(y) - \sum_{v \in V} \pi(v) y_v \leq f(y') - \sum_{v \in V} \pi(v) y'_v$ for all $y' \in \mathbb{Z}^V$.

Conditions 2 and 3 are called the negative cycle criterion and the potential function optimality criterion, respectively. The negative cycle criterion furnishes a means to improve a given suboptimal solution. Specifically, suppose that (x, y) is a suboptimal (integral) solution to an MSFP. Therefore, the auxiliary network has a negative cycle. Choose K to be a negative cycle in $F^{aux}(x, y)$ with the least number of arcs. Consider *augmenting* the current flow x along K and updating the associated net outflow y accordingly, i.e., :

- $x_{(a^+,a^-)} = x_{(a^+,a^-)} + 1$, if $a \in K \cap A^{aux}(x)$,
- $x_{(a^-,a^+)} = x_{(a^-,a^+)} - 1$, if $a \in K \cap B^{aux}(x)$,
- $y = y + \sum_{a \in K \cap C^{aux}(y)} (-\chi^{a^+} + \chi^{a^-})$.

Intuitively, the flow on arcs along K that are common to the underlying network F and have excess capacity (i.e., arcs in $A^{aux}(x)$) is increased. On the other hand, some arc a in the network may carry flow exceeding the associated lower bound, and hence there may be a corresponding arc in $F^{aux}(x, y)$ with a reversed orientation (which belongs to $B^{aux}(x)$). If this arc also belongs to K, then, the flow on a is reduced after augmentation. The net outflow is also updated so that it is consistent with the resulting induced flow.

It can be shown that augmenting a flow along the negative cycle K with the least number of arcs lowers the cost of the MSFP. Moreover, successive augmentation along such negative cycles[4] guarantees convergence to an optimal solution of the MSFP; see Murota (2003).

The potential values associated with the vertices of the flow network can be viewed as dual variables for the MSFP, and those satisfying Condition 3 are referred to as the *optimal*

[4]Successively augmenting the flow along any negative cycle guarantees convergence to an optimal solution in the classic minimum-cost network flow problems (without M-convex penalties) as well. However, unlike in the classic setting, in MSFP one must augment the flow along the negative cycle with the least number of arcs.

potential values. I conclude this section by providing an equivalent statement of the potential function optimality criterion of Theorem 2.2.1, which is used in the subsequent analysis. This characterization is in terms of the reduced costs of the arcs in the auxiliary network $F^{aux}(x,y)$. For a given set of potential values π at the vertices of the network, the *reduced cost* of each arc a in $F^{aux}(x,y)$ is given by $c_a^\pi = c_a^{aux}(x,y) + \pi(a^+) - \pi(a^-)$.

Theorem 2.2.2 (Reduced-Cost Optimality Condition)**.** A feasible solution (x^*, y^*) satisfies the optimality conditions of Theorem 2.2.1 with vertex potential function π, if and only if the following reduced-cost optimality condition holds:

$$c_a^\pi \geq 0 \text{ for each arc } a \text{ in } F^{aux}(x^*, y^*).$$

Proof. It can be readily seen that the conditions in Theorem 2.2.1, part 3(a), are equivalent to the reduced-cost optimality conditions for the subsets of arcs $A^{aux}(x^*)$ and $B^{aux}(x^*)$ in the auxiliary network $F^{aux}(x^*, y^*)$. Thus, to prove the claim it suffices to show that the reduced-cost optimality conditions for the remaining subset of arcs $C^{aux}(y^*)$ in $F^{aux}(x^*, y^*)$ hold if and only if the conditions of Theorem 2.2.1, part 3(b), hold.

This immediately follows from the fact that (i) $\bar{f}(y) = f(y) - \sum_{v \in V} \pi(v) y_v$ is an M-convex function, and (ii) for an M-convex function $\bar{f}$, local optimality is equivalent to global optimality, i.e., $\bar{f}(y_1) \leq \bar{f}(y_2)$ for all $y_2 = y_1 - \chi^u + \chi^v$ and vertices $u, v \in V$ if and only if $\bar{f}(y_1) \leq \bar{f}(y)$ for all y (see, e.g., Murota (2003)). Note that given net outflow y^*, we have an arc $(u,v) \in C^{aux}(y^*)$ if and only if $f(y^* - \chi^u + \chi^v) < \infty$. It follows that reduced costs are nonnegative for arcs in $C^{aux}(y^*)$, if we have $\bar{f}(y^*) \leq \bar{f}(y^* - \chi^u + \chi^v)$ for all $u, v \in V$, and vice versa. This in turn implies that nonnegativity of reduced costs in $C^{aux}(y^*)$ is equivalent to having $f(y^*) - \sum_{v \in V} \pi(v) y_v^* = \bar{f}(y^*) \leq \bar{f}(y) = \bar{f}(y) - \sum_{v \in V} \pi(v) y_v$ for any y. Since the latter condition is equivalent to Theorem 2.2.1 (3b), the claim follows. $\square$

Remark. MSFP can be defined without restricting (x, y) to be integral. For this version of MSFP, the domain of the penalty term on net outflows is the reals and the penalty term is required to be an *integral polyhedral M-convex function.* An integral polyhedral M-convex function $\bar{f}$ is obtained by extending an M-convex function f to $\mathbb{R}^n$ via its convex closure (Murota (2003), Section 6.11) defined as follows:

$$\bar{f}(x) = \sup_{\alpha \in \mathbb{R}^{n+1}} \left\{ \sum_{i=1}^{n} \alpha_i x_i + \alpha_0 \Big| \sum_{i=1}^{n} \alpha_i y_i + \alpha_0 \leq f(y) \ \forall y \in \mathbb{Z}^n \right\} \quad \text{for all } x \in \mathbb{R}^n.$$

Theorem 9.15 in Murota (2003) guarantees the existence of an optimal integer flow to MSFP when the capacities are integer-valued and the penalty function is integral polyhedral M-convex – a result analogous to integrality of optimal solution in classic network flow problems. Moreover, the optimality conditions provided in Theorem 2.2.1 continue to hold in such settings; see, e.g., Murota (2003).

2.2.1. $M^\natural$-Concave Functions

An $M^\natural$-convex function $f : \mathbb{Z}^n \to \mathbb{R}$ is a function satisfying the following exchange axiom

$$(M^\natural\textbf{-EXC}[\mathbb{Z}])\text{: For all } x, y \in \mathbb{Z}^n \text{ and for all } u \in supp^+(x-y),$$
$$f(x) + f(y) \geq \min[f(x - \chi^u) + f(y + \chi^u), \min_{v \in supp^-(x-y)} f(x - \chi^u + \chi^v) + f(x + \chi^u - \chi^v)].$$

An $M^\natural$-convex function is supermodular. A function f is $M^\natural$-concave if $-f$ is $M^\natural$-convex.

Any $M^\natural$-convex function $f : \mathbb{Z}^n \to \mathbb{R} \cup \infty$ can be represented as an M-convex function $f' : \mathbb{Z}^{n+1} \to \mathbb{R} \cup \infty$, where

$$f'(x_0, x) = \begin{cases} f(x) & \text{if } x_0 = -\sum_{i=1}^{n} x_i \\ +\infty & \text{otherwise.} \end{cases} \tag{2.2}$$

2.2.2. Full Substitutability

In Section 2.1, each w_i was defined as a function over subsets of $\delta(i)$. If sets are represented by their characteristic vectors, we can treat each w_i as a function over $\{0, -1\}^{\delta_-(i)} \times \{0, 1\}^{\delta_+(i)}$. I extend the domain of w_i to $\mathbb{Z}^{\delta(i)}$ by following the convention that $w_i(y) = -\infty$ for $y \in \mathbb{Z}^{\delta(i)}$ such that $y \notin \{0, -1\}^{\delta_-(i)} \times \{0, 1\}^{\delta_+(i)}$. An analogous convention applies to each payoff function u_i. Next, assume that for each $i \in N$, w_i is $M^\natural$-concave. This is equivalent to the property that each agent's preferences are fully substitutable (see Hatfield et al. (2013, 2019a) and Theorem 7 in Murota and Tamura (2003a)).

Definition 2.2.1. Agent i's preferences are *fully substitutable* if:

1. For all $p, \tilde{p} \in \mathbb{R}^{\delta(i)}$ such that $p_e = \tilde{p}_e$ for all $e \in \delta_+(i)$ and $\tilde{p}_e \geq p_e$ for all $e \in \delta_-(i)$, for every $Y^i \in D_i(p)$ there exists $\tilde{Y}^i \in D_i(\tilde{p})$ such that $(Y^i \cap \{e|p_e = \tilde{p}_e\}) \cap \delta_-(i) \subset \tilde{Y}^i \cap \delta_-(i)$ and $\tilde{Y}^i \cap \delta_+(i) \subset Y^i \cap \delta_+(i)$.

2. For all $p, \tilde{p} \in \mathbb{R}^{\delta(i)}$ such that $p_e = \tilde{p}_e$ for all $e \in \delta_-(i)$ and $\tilde{p}_e \leq p_e$ for all $e \in \delta_+(i)$, for every $Y^i \in D_i(p)$ there exists $\tilde{Y}^i \in D_i(\tilde{p})$ such that $(Y^i \cap \{e|p_e = \tilde{p}_e\}) \cap \delta_+(i) \subset \tilde{Y}^i \cap \delta_+(i)$ and $\tilde{Y}^i \cap \delta_-(i) \subset Y^i \cap \delta_-(i)$.

2.3. Transformation to MSFP

In this section, the optimality conditions of MSFP are used to show that a competitive equilibrium exists when agents have $M^\natural$-concave value functions, and shed light on its structure. To do so, I first transform the problem of finding an efficient set of trades into an instance of the MSFP.

I introduce a flow network $F = (V, A)$, associated with the trading network $G = (N, E)$. Recall that there is an $M^\natural$-concave function $w_i : \mathbb{Z}^{\delta(i)} \to \mathbb{R}$ associated with each vertex $i \in N$. I slightly modify this representation in Section 2.2.2, and represent the set of trades agent i participates in by a vector $y^i \in \mathbb{Z} \times \mathbb{Z}^{\delta(i)}$. Index the entries of y^i with 0 (to capture its first entry) and $e \in \delta(i)$. For each trade $e \in \delta(i)$ that occurs, set $y^i_e = 1$ if $e \in \delta_+(i)$ and $y^i_e = -1$ if $e \in \delta_-(i)$ (i.e., the entries of y^i corresponding to $e \in \delta(i)$ constitute the characteristic vector of trades agent i participates in). Also set $y^i_0 = -\sum_{e\in\delta(i)} y^i_e$, so that the entries of the y^i vector sum up to zero. With this representation each w_i can be replaced by an M-concave function $w'_i : \mathbb{Z} \times \mathbb{Z}^{\delta(i)} \to \mathbb{R}$, such that

$$w'_i(z_0, z) = \begin{cases} w_i(z) & \text{if } z_0 = -\sum\limits_{e \in \delta(i)} z_e \\ -\infty & \text{otherwise.} \end{cases} \tag{2.3}$$

For $y = \{y^i\}_{i \in N}$, the social welfare of the trading network is given by $-f(y) = \sum_{i \in N} w'_i(y^i)$. M-concavity of w'_i for all $i \in N$ implies that f is M-convex[5] as the arguments of the M-convex functions in the summand are disjoint.

In the flow network, each $i \in N$ is represented by a set V^i of vertices associated with the arguments of the M-concave function w'_i, i.e., $|V^i| = |\delta(i)| + 1$. Formally,

$$V = \bigcup_{i \in N} V^i,$$

where $V^i = \{v^i_e | e \in \{0\} \cup \delta(i)\}$. I refer to vertices of the form v^i_0 as *special vertices*. I add a directed arc between every pair of special vertices. In what follows, the orientation of this arc does not matter, and hence I pick it arbitrarily. The set of all arcs between special vertices is dentoed by A_0. Additionally, for each $e \in E$ I introduce an arc $a = (v^{e^+}_e, v^{e^-}_e)$. Intuitively, one unit of flow on this arc represents agents e^+ and e^- executing the trade e. These arcs form set $A_1 = \{(v^{e^+}_e, v^{e^-}_e) | e \in E\}$. Informally, each w'_i is a function of the characteristic vector of arcs incident to V^i that carry positive flow, as there is a one-to-one correspondence between the vertices V^i and the incident arcs.

Formally, the set of arcs in F is given by $A = A_0 \cup A_1$. Furthermore, I assume that for any arc $a \in A$, the associated flow costs are zero, i.e., $c_a = 0$, and the lower and upper capacities are set[6] as follows:

$$\underline{k}_a = -\infty, \overline{k}_a = +\infty. \tag{2.4}$$

[5]In general, the sum of M-convex functions is not M-convex. However, this property trivially holds when M-convex functions with disjoint arguments are considered.

[6]Because of this assumption on capacities, given any feasible flow on $F = (V, A)$, in the associated auxiliary network there are two directed arcs with opposite orientations between any pair of special vertices.

An example of this construction is displayed in Figure 2. Figure 2(a) is the trading network, where each vertex appears as a large black circle. Figure 2(b) shows the associated flow network. The special vertices appear as dotted circles. The vertices associated with a given agent appear together in the relevant rectangle. I refer to the induced subnetwork consisting of the vertices in a given rectangle as the corresponding agent's internal network. The net outflow at the vertices of an agent's internal network encode the trades the agent participates in.

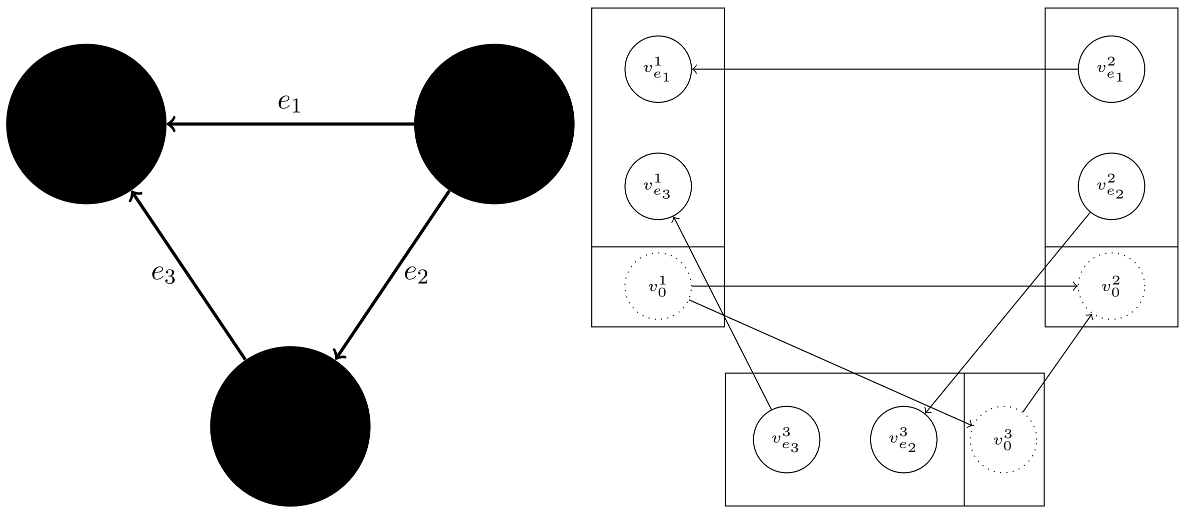

Figure 2: (a) Trading Network $G = (N, E)$ (b) Corresponding Flow Network $F = (V, A)$

I define the following instance of MSFP on $F = (V, A)$:

$$\begin{aligned}
\min_{x \in \mathbb{Z}^A, y \in \mathbb{Z}^V} \quad & f(y) \\
\text{s.t.} \quad & \sum_{a \in \delta_+(v)} x_a - \sum_{a \in \delta_-(v)} x_a = y_v \quad \forall v \in V \\
& \underline{k}_a \leq x_a \leq \overline{k}_a \quad \forall a \in A.
\end{aligned}$$

Recall that, by construction, $f(y)$ is M-convex. Suppose that a set $S \subset E$ of trades in the trading network $G = (N, E)$ is executed. A corresponding flow in $F = (V, A)$ can be obtained by sending one unit of flow on each arc in A_1 associated with these trades, and choosing the flow through arcs between special vertices to keep the total net outflow at the vertices in V^i equal to zero (which is possible since by (2.4) I can set negative flow values). Observe that the absolute value of the associated flow cost is equal to the welfare corresponding to S. Conversely, by the construction of f, it can be seen that any flow with bounded cost is such that the net outflow at the vertices in V^i is equal to zero for all i (see (2.3)), and each arc in A_1 carries at most one unit of flow. Moreover, the absolute value of the cost of any such flow is equivalent to the total welfare associated with the trades that correspond to the arcs in A_1 with nonzero flow. Hence, integer flows with bounded cost in F correspond to feasible sets of trades in G. Thus, the optimal solution of the MSFP corresponds to an efficient set of trades for the trading network $G = (N, E)$.

The construction of the auxiliary network associated with a given feasible solution (x, y) of this problem is displayed in Figure 3. The network in Figure 3(a) is the flow network from Figure 2. Suppose that one unit of flow is sent through $(v^2_{e_1}, v^1_{e_1})$ and (v^1_0, v^2_0), as indicated by the arrows, which corresponds to executing trade e_1 in Figure 2. The network in Figure 3(b) is the corresponding auxiliary network.

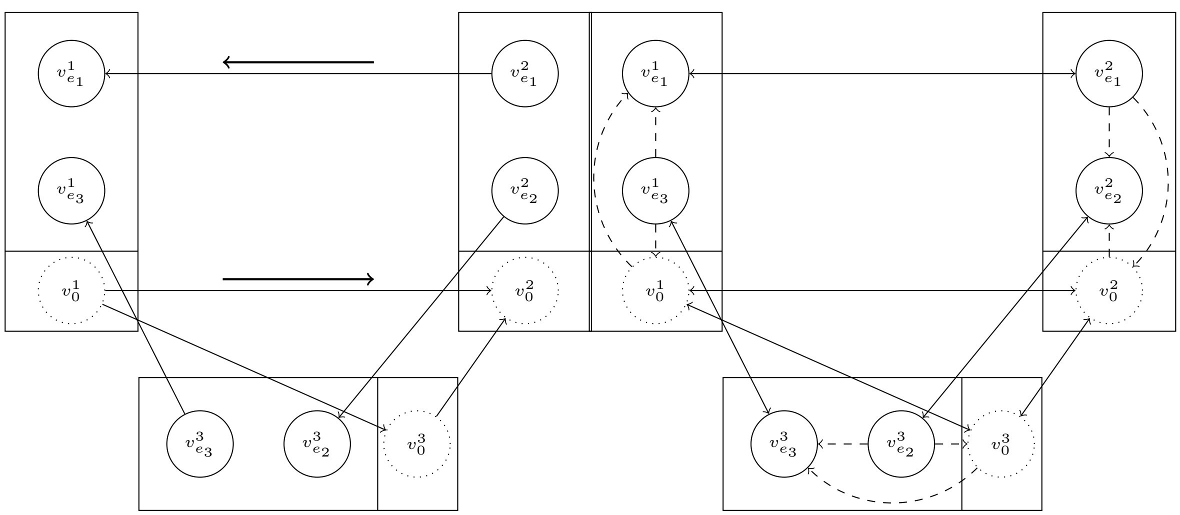

Figure 3: (a) Feasible Solution (b) Auxiliary Network (solid arcs: $A^{aux}(x) \cup B^{aux}(x)$, dashed arcs: $C^{aux}(y)$)

Consider an optimal solution (x, y) to the MSFP in F. There exists a vertex potential function π that satisfies the optimality conditions in Theorem 2.2.1, part 3. Moreover, by

Theorem 2.2.2, under this potential function the associated reduced costs of the arcs in the auxiliary network are nonnegative. Recalling that the arc costs of the auxiliary network are as in (2.1), and the arc cost of $a \in A$ satisfies $c_a = 0$, the reduced-cost optimality condition can be stated as follows:

$$\pi(v) - \pi(u) \leq 0 \qquad \forall (u, v) \in A^{aux}(x) \cup B^{aux}(x), \tag{2.5}$$

$$\pi(v) - \pi(u) \leq f(y - \chi^u + \chi^v) - f(y) \qquad \forall (u, v) \in C^{aux}(y). \tag{2.6}$$

Recall by (2.4) that $(u, v) \in A^{aux}(x)$ if and only if $(v, u) \in B^{aux}(x)$. Together with the above optimality conditions this implies that $\pi(u) = \pi(v)$ for all $(u, v) \in A$. For $(u, v) \in A_1 \subset A$, this potential value is the candidate price for the trade associated with (u, v). Similarly, since there is an arc in $A_0 \subset A$ between any pair of special vertices, it follows that each special vertex has a potential value equal to some π_0. Furthermore, given a potential function $\pi : V \to \mathbb{R}$ satisfying conditions (2.5) and (2.6), setting $\pi'(u) = \pi(u) + c$ for all $u \in V$ gives another potential function satisfying these conditions. By setting $c = -\pi_0$ the potential function is normalized to take a value of zero at the special vertices throughout my analysis.

Potential values are defined at vertices. Recall that in the construction of flow networks each vertex corresponds to a particular agent-trade pair and the optimal potential values of two adjacent vertices (associated with the same trade) are equal. Theorem 2.2.1 (3b) implies that if these potential values are interpreted as prices, and the set of trades y^i chosen for some agent i is unilaterally modified (through the choice of a different outflow at the vertices in V^i), then the payoff of agent i cannot be improved. Thus, it follows that an optimal solution (x, y) of the MSFP and the prices that correspond to potential values satisfying Theorem 2.2.1 (3b) constitute a competitive equilibrium.

Conversely, given a competitive equilibrium, the prices for trades define potential values at all vertices of the flow network, where the potential value of a vertex is the price of the corresponding trade and the special vertices get a potential value of zero. The equilibrium conditions imply that the equilibrium prices and trades satisfy Theorem 2.2.1 (3a–3b). Hence, the flow associated with the equilibrium trades solves the MSFP. Thus, the equivalence of optimal solutions of the MSFP to efficient sets of trades, as well as the equivalence of optimal potential values to competitive prices, follow.

2.3.1. Immediate Consequences

Theorem 2.3.1. (Theorem 1 in Hatfield et al. (2013)) There exists a competitive equilibrium.

Proof. Given a trading network $G = (N, E)$, I map it to the associated flow problem in the flow network (V, A). The MSFP in (V, A) has an optimal solution (x^*, y^*), since it is a discrete problem and "no flow" is a feasible solution. Theorem 2.2.1 implies that there exists an optimal potential function π^* satisfying Condition 3. The trades that correspond to (x^*, y^*), along with the prices associated with the potential function π^*, constitute a competitive equilibrium, and the claim follows. □

The set of trades associated with a competitive equilibrium is efficient.

Theorem 2.3.2. (First Welfare Theorem, Theorem 2 in Hatfield et al. (2013)) Suppose that (X, p) is a competitive equilibrium. Then, X is an efficient set of trades.

Proof. Let (x, y) be a feasible flow associated with the set of trades X. The competitive prices define a potential function for the flow (x, y). By Theorem 2.2.1, (x, y) is optimal; therefore, the set of trades X is efficient. □

Next, it is shown that competitive prices support all efficient sets of trades; i.e., competitive prices along with any efficient set of trades constitute a competitive equilibrium.

Theorem 2.3.3. (Second Welfare Theorem (strong version), Theorem 3 in Hatfield et al. (2013)) For any competitive equilibrium (X, p) and efficient set of trades X', (X', p) is also a competitive equilibrium.

Proof. The sets of trades X, X' correspond to optimal flows (x, y) and (x', y'), respectively. The prices define a potential function associated with the optimal flow (x, y). The second part of Theorem 3.1 in Murota (1999) states that the potential function satisfies the conditions of Theorem 2.2.1 (3) for the flow (x', y'). I conclude that (X', p) is a competitive equilibrium. □

The set of competitive prices has a lattice structure.

Theorem 2.3.4. (Theorem 4 in Hatfield et al. (2013)) The set of competitive price vectors is a lattice.

Proof. The claim[7] is immediate from the fact that the feasible region of the system of difference constraints (2.5) and (2.6) is a lattice. □

Remark. Hatfield et al. (2013) established the existence of a competitive equilibrium in trading networks, by reducing the trading network to a two-sided market. Then, they invoked the existence results of Kelso and Crawford (1982) for two-sided markets. Finally, the authors used this equilibrium to construct a corresponding equilibrium for the underlying trading network model, which implies the existence of competitive equilibria in trading networks. The proof of Kelso and Crawford (1982) relies on discrete prices and a price update process, and requires establishing that this process terminates in a finite number of steps. The limiting point is always a competitive equilibrium. My approach eliminates the need for discrete prices and the need to devise a specific price update process that converges

[7]Theorem 9.15 in Murota (2003) implies a stronger version of this result, which can be explained through L-convexity. A set $B \subset \mathbb{Z}^n$ is L-convex if (i) for $p, q \in B$ we have $p \vee q, p \wedge q \in B$, and (ii) for $p \in B$ we have $p \pm \mathbf{1} \in B$. Here, $p \vee q$ ($p \wedge q$) is a vector, which is obtained by taking the component-wise maximum (minimum) of the p and q vectors, and $\mathbf{1}$ is the vector of ones. B' is $L^\natural$-convex if $B' = \{p | (0, p) \in B\}$ for some L-convex set B. Convex hulls of L-convex ($L^\natural$-convex) sets are referred to as L-convex ($L^\natural$-convex) polyhedra. Theorem 9.15 in Murota (2003) implies that the set of optimal potential functions can be represented as an L-convex polyhedron. Therefore, the set of competitive prices, which is a restriction of the potential values to the coordinate plane, is an $L^\natural$-convex polyhedron. $L^\natural$-convex polyhedra exhibit lattice structure.

to a competitive equilibrium. Nor does it rely on any fixed-point arguments. Instead, the competitive equilibrium allocation and prices are obtained directly in terms of primal/dual optimal solutions of an optimization problem (i.e., the optimal flows and potential functions in MSFP).

2.3.2. Multiple Identical Trades

In the trading network model one can interpret a trade as the sale of a unit of a good from a seller to a buyer. For the case where there are multiple identical units of a good offered by the seller, Hatfield et al. (2013) gave a sufficient condition for the existence of a competitive equilibrium, where all "identical" trades receive the same price. The connection to MSFP is used to extend this sufficient condition. I define what it means for two trades to be perfect substitutes for each other.

Definition 2.3.1. Agent i's trades $e, e' \in \delta_+(i)$ (similarly $e, e' \in \delta_-(i)$) are *perfect substitutes* if $w_i(X \cup \{e\}) = w_i(X \cup \{e'\})$ for all $X \subset \delta(i) \setminus \{e, e'\}$.

This definition immediately implies that the value function of agent i depends only on the *number* of trades chosen in an equivalence class of perfectly substitutable trades Y associated with her, i.e., $w_i(X \cup S) = w_i(X \cup S')$ for all $S, S' \subset Y$ such that $|S| = |S'|$ and for all $X \subset \delta(i) \setminus Y$.

In Hatfield et al. (2013) it was established that there exists a competitive equilibrium where trades that are perfect substitutes receive the same price, provided that these trades are also *mutually incompatible*, i.e., accepting more than two such trades leads to a payoff of $-\infty$. The next result shows that such an equilibrium still exists, when the mutual incompatibility assumption is relaxed. Importantly, this relaxation allows the seller to produce and sell *multiple* identical goods. A similar result, where each seller offers multiple identical goods to each buyer, but offers distinct goods to distinct buyers, is given in Ikebe et al. (2015). Theorem 7 generalizes this result by also allowing the sellers to offer identical goods to different buyers. The proof of this result is a simple consequence of the MSFP formulation.

Theorem 2.3.5. There exists a competitive equilibrium where any two trades that are perfect substitutes for some agent receive the same price.

Proof. As before, I introduce a flow network $F = (V, A)$, associated with the trading network G. Recall that an M-concave function w'_i is assigned to each agent with each argument capturing the net outflow at a vertex $u \in V^i$. Suppose that agent i has L_i disjoint sets of trades, and the trades in each set are perfect substitutes for each other. Specifically, for every $k \in \{1, \ldots, L_i\}$, $Y^i_k \subset \delta(i)$ denotes a set of trades incident to agent i, such that any two trades in Y^i_k are perfect substitutes for agent i. Denote by $V_{Y^i_k} \subset V^i$ the vertices associated with trades Y^i_k in F. Merge all vertices in $V_{Y^i_k}$ into a vertex $v_{Y^i_k}$. Make each arc incident to a vertex in $V_{Y^i_k}$ incident to $v_{Y^i_k}$. I obtain a new set of vertices associated with agent i given by $V^i_m = \left(V^i \setminus R_i\right) \cup_{k=1}^{L_i} \{v_{Y^i_k}\}$, where $R_i = \cup_{k=1}^{L_i} V_{Y^i_k}$.

Define a new function $\hat{w}_i : \mathbb{Z}^{V^i_m} \to \mathbb{R}$ such that

$$\hat{w}_i(y^i_{V^i \setminus R_i}, y_{Y^i_1}, \ldots, y_{Y^i_{L_i}}) = \sup \left\{ w'_i(y^i_{V^i \setminus R_i}, z) \middle| \sum_{e \in Y^i_k} z_e = y_{Y^i_k}, \forall k \in \{1, \ldots, L_i\} \right\}.$$

The function $\hat{w}_i$ is generated by (repeated) aggregation of the original M-concave function w'_i. Aggregation preserves M-concavity (see Theorem 6.13 in Murota (2003)).

Consider the MSFP formulation associated with the network obtained after merging all vertices in each $V_{Y^i_k}$ into a single vertex $v_{Y^i_k}$ and imposing the penalty function $\hat{f}() = -\sum_i \hat{w}_i()$. The optimal flow in this formulation corresponds to the set of trades in the trading network, where the net outflow from a vertex $v_{Y^i_k}$ represents the total number of trades executed in the set of trades Y^i_k. Moreover, since all trades in Y^i_k are perfectly substitutable, by the construction of $\hat{w}_i$, the absolute value of the cost of the optimal flow is equivalent to the maximum total welfare.

The theorem follows from the equivalence between potential values and competitive prices. As argued before, Theorem 2.2.1 (3b) gives a potential function π which associates values with the vertices of the flow network that can be interpreted as prices. Moreover, under these prices agents maximize their payoff by choosing the trades associated with the optimal flow; i.e., the aforementioned trades and prices constitute a competitive equilibrium. By construction, all arcs corresponding to trades in Y_k^i are adjacent to a single vertex $v_{Y_k^i}$. Thus, there exists a unique potential value/price $\pi(v_{Y_k^i})$ for trades Y_k^i. Hence, it follows that a competitive equilibrium where all trades in Y_k^i receive identical prices exists. $\square$

CHAPTER 3 : Applications of the Network Flow Approach in Trading Networks

Section 3.1 discusses the equivalence of various stability notions, Section 3.2 discusses algorithms for obtaining competitive equilibria and testing (chain) stability, and Section 3.3 presents several comparative statics results. I conclude in Section 3.4.

3.1. Stable Outcomes

In this section I first review various notions of stability for trading networks proposed in Hatfield et al. (2013). Informally, a stable outcome has the property that no subset of agents has an incentive to deviate from it. Given a set of trades X, the prices of the corresponding trades are denoted by p^X, and the set of trades agent i demands once she is restricted to the trades in X is denoted by $D_i(p^X) \subset X \cap \delta(i)$. I refer to the tuple (X, p^X) as an outcome. Call an outcome (X, p^X) *individually rational* if

$$X \cap \delta(i) \in \operatorname*{arg\,max}_{Y \subset X \cap \delta(i)} w_i(Y) + \sum_{e \in Y \cap \delta_+(i)} p_e^X - \sum_{e \in Y \cap \delta_-(i)} p_e^X \quad \forall i \in N.$$

Definition 3.1.1. An outcome (X, p^X) is *stable* if it is individually rational and is unblocked:

There is no feasible, nonempty blocking set $Z \subset E$, along with prices p^Z, such that

1. $Z \cap X = \emptyset$, and
2. for all agents i involved in Z, for all $Y^i \in D_i(p^{Z \cup X})$, we have $Z \cap \delta(i) \subset Y^i$.

The closely related notion of strongly stable outcome is defined next.

Definition 3.1.2. An outcome (X, p^X) is *strongly stable* if it is individually rational and is strongly unblocked:

There is no feasible, nonempty blocking set $Z \subset E$, along with prices p^Z, such that

1. $Z \cap X = \emptyset$, and

2. for all agents i involved in Z, there exists a $Y^i \subset (Z \cup X) \cap \delta(i)$ such that $Z \cap \delta(i) \subset Y^i$ and

$$w_i(Y^i) + \sum_{e \in Y^i \cap \delta_+(i)} p_e^{Z \cup X} - \sum_{e \in Y^i \cap \delta_-(i)} p_e^{Z \cup X} > w_i(X \cap \delta(i)) + \sum_{e \in X \cap \delta_+(i)} p_e^X - \sum_{e \in X \cap \delta_-(i)} p_e^X.$$

Clearly, a strongly stable outcome is stable.

The next notion of stability is analogous to pairwise stability in bipartite matching. I refer to a set of consecutive arcs in a graph G, i.e., a set of m arcs $S = \{e_1, \ldots, e_m\}$, such that $e_i^- = e_{i+1}^+$ for all $i = 1, \ldots, m-1$, as a *chain*.

Definition 3.1.3. An outcome (X, p^X) is *chain stable* if it is individually rational and is unblocked by a chain:
There is no feasible, nonempty blocking chain $Z \subset E$, along with prices p^Z, such that

1. $Z \cap X = \emptyset$, and

2. for all agents i involved in Z, for all $Y^i \in D_i(p^{Z \cup X})$, we have $Z \cap \delta(i) \subset Y^i$.

The related notion of strong chain stability is defined below.

Definition 3.1.4. An outcome (X, p^X) is *strongly chain stable* if it is individually rational and is strongly unblocked by a chain:
There is no feasible, nonempty blocking chain $Z \subset E$, along with prices p^Z, such that

1. $Z \cap X = \emptyset$, and

2. for all agents i involved in Z, there exists a $Y^i \subset (Z \cup X) \cap \delta(i)$ such that $Z \cap \delta(i) \subset Y^i$ and

$$w_i(Y^i) + \sum_{e \in Y^i \cap \delta_+(i)} p_e^{Z \cup X} - \sum_{e \in Y^i \cap \delta_-(i)} p_e^{Z \cup X} > w_i(X \cap \delta(i)) + \sum_{e \in X \cap \delta_+(i)} p_e^X - \sum_{e \in X \cap \delta_-(i)} p_e^X.$$

Clearly, a strongly chain stable outcome is chain stable. Definitions 3.1.1–3.1.4 also imply that a (strongly) stable outcome is (strongly) chain stable, since if there exists no (strongly) blocking set, there exists no such set with a chain structure.

Before I show the equivalence of these stability concepts, I focus on the case where not executing any trades is inefficient. In this case it is shown that it is always possible to find a chain that improves welfare. Intuitively, this preliminary result implies that it may be possible to restrict attention to chains when searching for a blocking set. I subsequently formalize this intuition in Corollary 3.1.2 for outcomes where no trade is executed.

Lemma 3.1.1. Consider a trading network $G = (N, E)$. If not executing any trades is inefficient, there exists a chain of trades that improve welfare.

Proof. Consider the MSFP formulation of the welfare-maximization problem in G, and let (x, y) denote a feasible solution of the MSFP associated with flow network $F = (V, A)$ that corresponds to executing no trades, i.e., that associates zero flow with all arcs in A_1, and hence guarantees $y = 0$. Since executing no trades in G is inefficient, according to Theorem 2.2.1 there exists a negative cycle in the auxiliary network $F^{aux}(x, y)$. Pick a negative cycle K with the least number of arcs. Observe that this cycle visits each vertex of $F^{aux}(x, y)$ at most once, as otherwise there would exist a negative cycle with fewer arcs.

I claim that there exists such a cycle K that satisfies the following conditions:

1. $0 > \sum_{a \in K} c_a^{aux}(x, y) = \sum_{(u,v) \in K \cap C^{aux}(y)} [f(y - \chi^u + \chi^v) - f(y)]$.
2. The cycle contains at most one special vertex *or* two incident special vertices.
3. If $(u, v) \in K \cap B^{aux}(x)$, then $(v, u) \in A_0$.
4. For all arcs in $h \in K \cap A_1$, there exist $h_s, h_b \in K \cap C^{aux}(y)$ such that $h_s - h - h_b$ is a sequence of arcs along K.

The first condition follows since K is a negative cycle, and arc costs are nonzero only for arcs in $C^{aux}(y)$. Suppose that the second condition is violated. Then there are two special vertices that are not connected by an arc along K. Add the arcs with zero cost between them to K. Then, we get two cycles with fewer arcs, such that at least one has negative length. This contradicts the assumption that K is the negative cycle with the least number of arcs.

To prove the third condition, observe that $C^{aux}(y) = \{(u,v)|u,v \in V, f(y - \chi^u + \chi^v) < \infty\}$ consists only of arcs (u,v), where (i) $u,v \in V^i$ for some agent i, and (ii) $u \in \{v_e^i | e \in \{0\} \cup \delta_-(i)\}$ and $v \in \{v_e^i | e \in \{0\} \cup \delta_+(i)\}$. To see (i), note that $y = 0$, and, by construction, $f(z) < \infty$ only when the total net outflow at the vertices in V^i is zero for all i. Thus, if this claim does not hold, then $f(y - \chi^u + \chi^v) = \infty$, indicating that $(u,v) \notin C^{aux}(y)$. Similarly, property (ii) follows since, by construction, $f = -\sum_i w'_i$, and the definition of w'_i implies that $f(y - \chi^u + \chi^v) = \infty$ unless this property holds.

Suppose by way of contradiction that $(u,v) \in K \cap B^{aux}(x)$ and $(v,u) \in A_1$. Observe that $(v,u) \in A_1$ implies that $v \in \{v_e^i | e \in \delta_+(i)\}$ for some agent i. Hence, the next arc (v,v') along K, cannot belong to $C^{aux}(y)$ (as this would violate (ii)). Since the arcs in A_1 are disjoint, this arc is given by $(v,v') = (v,u)$. By omitting both (u,v) and (v,u) from K, a negative cycle with the same length but fewer arcs can be obtained, thereby leading to a contradiction. Thus, the third condition follows.

For the fourth condition, fix $h = (u,v) \in K \cap A_1$. Since the arcs in A_1 correspond to disconnected components of $F = (V, A)$, it follows that the next (similarly previous) arc along K is either $(v,u) \in B^{aux}(x)$ or an element of $C^{aux}(y)$. The third condition together with the fact that $(u,v) \in A_1$ rules out the former case. The latter case implies the fourth condition.

Theorem 9.22 in Murota (2003) implies that

$$f(y) > f(y) + \sum_{(u,v) \in K \cap C^{aux}(y)} [f(y - \chi^u + \chi^v) - f(y)] \geq f\left(y + \sum_{(u,v) \in K \cap (A^{aux}(x) \cup B^{aux}(x))} (\chi^u - \chi^v)\right),$$

since K is a negative cycle with the fewest arcs. The term on the right-hand side is the cost of the flow obtained after modifying the original flow by sending one unit of flow on arcs $A_1 \cap K$ and adjusting the flow on A_0 according to $K \cap (A^{aux}(x) \cup B^{aux}(x))$ so that the net outflow at V^i is zero for all i. Thus, I conclude that executing the set of trades associated with arcs $A_1 \cap K$ improves welfare. I complete the proof by showing that this set of trades constitutes a chain in the trading network.

Assume that K does not involve any special vertices. Consider an arc $h \in K \cap C^{aux}(y)$, and recall that both end points of this arc belong to V^i for some agent i. Both the predecessor and successor of this arc along K belong to A_1, since the arcs in $C^{aux}(y)$ incident to a non-special vertex either all have this vertex as their head or they all have it as their tail. This, along with the fourth condition, implies that arcs along K alternate between $A^{aux}(x)$ and $C^{aux}(y)$. The successor of h connects a non-special vertex in V^i to a non-special vertex in V^j for some $j \neq i$, thereby capturing trades between i and j. Since arcs along K alternate between $A^{aux}(x)$ and $C^{aux}(y)$, it follows that the next arc's (say h''s) end points belong to V^j. By the same argument it can be seen that the arc after h' suggests a trade relation between j and some other agent k. Thus, proceeding iteratively, I conclude that the set of trades associated with arcs $A \cap K$ constitutes a chain[1] in G.

The same argument still holds when there is a single special vertex $v \in V^i$ for some i that belongs to K. Arcs alternate between $A^{aux}(x)$ and $C^{aux}(y)$, aside from the arcs adjacent to v. Since v is connected to other special vertices and vertices in V^i, and K visits a single special vertex, it follows that there exist $u, u' \in V^i$ such that $(u, v) - (v, u')$ belongs to K. Since u, u' are non-special vertices in V^i, the earlier argument implies that the successor (predecessor) of (v, u') ((u, v)) belongs to A_1. Thus, proceeding as before, I conclude that the set of trades associated with arcs $A \cap K$ constitutes a chain starting and ending at i.

Assume instead that K involves an arc between special vertices. Since K involves at most

[1] In this case, it can be seen that the aforementioned trades also constitute a cycle in G. This is because special vertices are not visited.

two special vertices, there can be only one such arc. Starting with such an arc, and proceeding as before, it follows that the remaining arcs along K suggest a chain of trades that correspond to the arcs $A \cap K$.

Hence, I conclude that the trades identified by the smallest negative cycle induce a chain of welfare-improving trades, as claimed. $\square$

The optimality conditions for MSFP and the structure of the flow network play a key role in the proof of Lemma 3.1.1. This result also has a straightforward corollary that characterizes blocking chains in terms of a *minimal* set T of trades that improve welfare, i.e., T such that no subset of T improves welfare when compared to executing no trades.

Corollary 3.1.2. Consider a trading network $G = (N, E)$. Assume that not executing any trades is inefficient. Then,

(i) any minimal set of trades that improve welfare constitutes a chain, and

(ii) there exist prices that together with these trades constitute a blocking chain.

Proof. (i) Assume $\emptyset$ is not efficient in G, and let $T \subset E$ be a minimal set of trades that strictly improve welfare. Consider a trading network $\hat{G} = (N, T)$, obtained by restricting the original set of trades to T. Observe that $\emptyset$ is also not welfare-maximizing in $\hat{G}$. Lemma 3.1.1 implies that there exists a welfare-improving set of trades that constitutes a chain in $\hat{G}$. Since T is the minimal (and only) set of trades that improves welfare, it follows that T is a chain.

(ii) Since T is a minimal welfare-improving set of trades, it follows that in $\hat{G} = (N, T)$ the unique efficient set of trades is T.

Let $\Delta > 0$ be such that $\sum_i w_i(T \cap \delta(i)) - 2\Delta|T| > \sum_i w_i(X \cap \delta(i)) - 2\Delta|X|$ for any $X \subsetneq T$. It suffices to choose a $\Delta > 0$, such that $2\Delta|T| < \sum_i w_i(T \cap \delta(i)) - \sum_i w_i(\emptyset)$ (recall that $\sum_i w_i(X \cap \delta(i)) \leq \sum_i w_i(\emptyset)$ for any $X \subsetneq T$, since T is a minimal welfare-improving set of trades). Consider another economy with the same network structure $\hat{G} = (N, T)$, but with valuations[2] $\bar{w}_i(Z) = w_i(Z) - \Delta|Z|$, where $Z \subset \delta(i)$. Observe that for any set of trades $X \subsetneq T$ we have

$$\begin{aligned}\sum_i \bar{w}_i(T \cap \delta(i)) = \sum_i \left(w_i(T \cap \delta(i)) - \Delta|T \cap \delta(i)|\right) = \sum_i w_i(T \cap \delta(i)) - 2\Delta|T| \\ > \sum_i w_i(X \cap \delta(i)) - 2\Delta|X| = \sum_i \bar{w}_i(X \cap \delta(i)).\end{aligned} \tag{3.1}$$

Thus, it follows that T is still the unique efficient set of trades in this economy. Denote a competitive equilibrium in this economy by (T, p^T).

I claim that (T, p^T) is a competitive equilibrium in the economy with value functions $\{w_i\}_{i \in N}$, where $D_i(p^T) = \{T \cap \delta(i)\}$. This is because, if a set of trades $T \cap \delta(i)$ is demanded in the economy with value functions $\{\bar{w}_i\}_{i \in N}$, for any $S \subsetneq T \cap \delta(i)$ we have

$$u_i(T \cap \delta(i), p^T) - \Delta|T \cap \delta(i)| = \bar{u}_i(T \cap \delta(i), p^T) \geq \bar{u}_i(S, p^T) = u_i(S, p^T) - \Delta|S|, \tag{3.2}$$

where $\bar{u}_i$ is the payoff function associated with $\bar{w}_i$. This implies that the payoff of agent i for the trades in $T \cap \delta(i)$ is strictly greater than her payoff for any set of trades $S \subsetneq T \cap \delta(i)$.

Thus, I conclude that in the economy with value functions $\{w_i\}_{i \in N}$ we have $D^i(p^T) = \{T \cap \delta(i)\}$. Hence, it follows that (T, p^T) is a blocking chain for the outcome $(\emptyset, p^{\emptyset})$. □

The definitions in this section imply that verifying stability of an outcome (X, p^X) requires focusing on trades that belong to $E \setminus X$ and establishing that there is no blocking set (or chain) in $E \setminus X$. Thus, to study the stability of an outcome (X, p^X) with $X \neq \emptyset$, it is necessary to study the preferences of agents for trades in $E \setminus X$. To this end, the idea of a *contraction of an economy* (see Hatfield et al. (2013)) is used. For an outcome (X, p^X),

[2] The use of "modified valuations" was employed in Hatfield et al. (2013) to establish that a stable outcome can be supported with appropriate prices to obtain a competitive equilibrium. I follow a similar construction to show that if the efficient allocation is unique, then there exist competitive equilibrium prices under which each agent strictly demands her equilibrium set of trades. Note, that this result is independent of the trading network structure, and is a byproduct of strict complementarity in optimization.

I define a new trading network $G^X = (N, E \setminus X)$, where agent $i \in N$ has a value function $\hat{w}_i : 2^{\delta(i) \cap (E \setminus X)} \to \mathbb{R}$ given as follows:

$$\hat{w}_i(S) = \max_{Y \subset X \cap \delta(i)} [w_i(S \cup Y) + \sum_{e \in Y \cap \delta_+(i)} p_e^X - \sum_{e \in Y \cap \delta_-(i)} p_e^X]. \tag{3.3}$$

It follows from Murota (2003) (Theorem 6.13 (3), Theorem 6.15) that $\hat{w}_i$ is $M^\natural$-concave for each $i \in N$. Refer to G^X as the contraction of economy G, with respect to (X, p^X).

3.1.1. Equivalence of Solution Concepts

Given a competitive equilibrium (X, p) I refer to the tuple (X, p^X), obtained after restricting the prices to the trades in X, as a competitive equilibrium outcome. It is next shown that all notions of stable outcomes coincide with competitive equilibrium outcomes. My approach involves the following two steps (established in Theorems 3.1.3 and 3.1.4, respectively):

1. A competitive equilibrium outcome is a (strongly) stable outcome.

2. A chain stable outcome is a competitive equilibrium outcome.

The first result follows from the definition of stability, while the second follows from Lemma 3.1.1 and Corollary 3.1.2, which exploit the network flow formulation.

Theorem 3.1.3. (Theorem 5 in Hatfield et al. (2013)) If (X, p) is a competitive equilibrium in trading network G and p^X is the restriction of p to the arcs in X, then, (X, p^X) is a (strongly) stable outcome in G.

Proof. Since (X, p) is a competitive equilibrium, it follows that (X, p^X) is individually rational. To complete the proof, it suffices to show that there is no set of trades that (strongly) blocks (X, p^X). Let G^X be the contraction with respect to (X, p^X). Since (X, p) is a competitive equilibrium in G, $(\emptyset, p^{(E \setminus X)})$ is a competitive equilibrium in G^X. Theorem 2.3.2 implies that $\emptyset$ is an efficient set of trades in G^X. Suppose, for a contradiction, there exist trades and prices (Z, p^Z) that (strongly) block (X, p^X) in G. This would imply that Z has higher welfare than $\emptyset$ in G^X, which contradicts the efficiency of $\emptyset$. $\square$

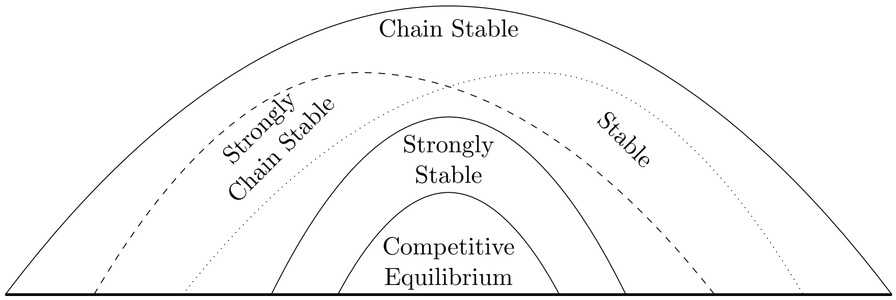

Figure 4: Outcomes Hierarchy

As any (strongly) stable outcome is (strongly) chain stable, Theorem 3.1.3 implies the hierarchy displayed in Figure 4.

Next, I establish the equivalence of all the stability notions, by showing that in any chain stable outcome (X, p^X), it is possible to find prices for trades $E \setminus X$ to support X as a competitive equilibrium. Thus, the "weakest" and "strongest" equilibrium/stability notions in Figure 4 are equivalent.

Theorem 3.1.4. Suppose that (X, p^X) is a chain stable outcome in trading network G. Then, there exists a price vector $p \in \mathbb{R}^E$, with $p_e = p_e^X$ for all $e \in X$, such that (X, p) is a competitive equilibrium in G.

Proof. Consider the contraction G^X of the trading network G with respect to (X, p^X). I claim that for some price vector $\hat{p}^{E\setminus X}$, $(\emptyset, \hat{p}^{E\setminus X})$ is a competitive equilibrium in G^X. Assume not, then, it follows from Theorems 2.3.1 and 2.3.3 that $\emptyset$ is not welfare maximizing in G^X. Then, Corollary 3.1.2 implies that this outcome is not chain stable, and there exists a set of trades $T \subset E \setminus X$, and prices p^T that constitute a blocking chain in G^X. This implies that (T, p^T) also blocks (X, p^X) in the original economy G. Thus, a contradiction is obtained, and it follows that $(\emptyset, \hat{p}^{E\setminus X})$ is a competitive equilibrium in G^X. Since (X, p^X) is chain stable and hence individually rational, this implies that in the economy G, under prices $(p^X, \hat{p}^{E\setminus X})$, each agent i demands $X \cap \delta(i)$. Hence, this outcome corresponds to a competitive equilibrium, and the claim follows. $\square$

3.2. Algorithms for Trading Networks

The MSFP formulation leads to computationally efficient algorithms for obtaining competitive equilibria and stable outcomes. In this section, I first outline algorithms that can be used to obtain competitive equilibria and various associated quantities. I then provide an algorithm that can be used to check whether a given outcome is (chain) stable.

3.2.1. Competitive Equilibrium Computation

An algorithm for solving MSFP in time that is polynomial in input size is described in Iwata et al. (2005). The transformation of the problem of finding an efficient set of trades into an instance of MSFP also has polynomial complexity. Thus, it follows that an efficient set of trades for trading networks can be obtained in polynomial time, by formulating and solving[3] the associated MSFP.

The algorithm outlined requires oracle knowledge of agents' valuations. Alternatively, it is possible to design a tâtonnement process that converges to competitive equilibria. Specifically, it is possible to set prices, collect the set of demanded trades by each trader, and adjust prices until convergence to a competitive equilibrium. Such iterative price update schemes are privacy-preserving, in the sense that they do not necessitate knowledge of the entire value function of the agents, and are employed in iterative auction design; see, e.g., Ausubel (2006).

Consider for instance Algorithm 1, which is initialized with arbitrary (integral) prices for trades. At each stage, it collects agents' demand reports, i.e., the sets of trades that are demanded at the current price vector p. Then, Algorithm 1 checks whether it is possible to choose a set of trades X that is consistent with agents' demand reports; i.e., X assigns to

[3] The algorithm in Iwata et al. (2005) runs in strongly polynomial time; i.e. the number of arithmetic operations performed does not depend on the magnitudes of agents' valuations. Thus, it is possible to compute the efficient set of trades in a number of operations and space bounded by a polynomial in input size. Similarly, it can be shown that by computing the shortest distances from special vertices to the remaining vertices of the auxiliary network, potential values that satisfy the difference constraints (2.5) and (2.6) can be readily obtained. This can be accomplished in strongly polynomial time, using shortest-path algorithms, e.g., the Bellman-Ford algorithm. Using these observations it is possible to strengthen all of the results on the polynomial complexity in this and the next subsection. Specifically, it can be shown that a competitive equilibrium can be obtained, and stability can be checked in strongly polynomial time for trading networks.

each agent demanded trades ($X \cap \delta(i) \in D_i(p)$ for each i).

ALGORITHM 1: Tâtonnement Process

1 Initialize price vector $p = p^0$, where $p^0 \in \mathbb{Z}^E$;
2 Collect agents' demand reports $D_i(p)$;
3 **if** *no set of trades is consistent with demand reports* **then**
4 $\quad \{\epsilon^*, S^*\} \in \underset{\epsilon \in \{-1,1\}, S \subset E}{\arg\min} \sum_{i \in N} \left[\max_{Z^i \in D_i(p)} \left\{ \sum_{e \in S \cap Z^i \cap \delta_+(i)} \epsilon - \sum_{e \in S \cap Z^i \cap \delta_-(i)} \epsilon \right\} \right];$
5 $\quad p := p + \epsilon^* \chi^{S^*}$;
6 $\quad$ GOTO step 2;
7 **else**
8 $\quad$ **return** p;
9 **end**

Note that if such a set of trades X exists, then it is also efficient, and p is a vector of competitive prices. Otherwise, Algorithm 1 increments/decrements the price of a set of trades. Suppose that the algorithm chooses (ϵ^*, S^*) and updates the prices by $\epsilon^* \chi^{S^*}$, where $\epsilon^* \in \{-1, 1\}$ and $\chi^S \in \{0,1\}^E$ is a characteristic vector for set S; i.e., its entry corresponding to $e \in E$ is equal to one if and only if $e \in S$. Then, the quantity

$$\max_{Z^i \in D_i(p)} \left\{ \sum_{e \in S^* \cap Z^i \cap \delta_+(i)} \epsilon^* - \sum_{e \in S^* \cap Z^i \cap \delta_-(i)} \epsilon^* \right\}$$

captures the corresponding change in the maximum payoff of agent i from the previously demanded sets in $D_i(p)$. Thus, Algorithm 1 updates the prices in a way that decreases the aggregate payoff of agents from demanded bundles as much as possible.

Convergence of Algorithm 1 to a competitive equilibrium can be established by showing that the price updates in the algorithm decrease the aggregate payoff of agents, given as follows:

$$g(p) = \sum_{i \in N} \max_{S \subset \delta(i)} u_i(S, p).$$

Thus, this function can be used as a Lyapunov function for establishing convergence. Note that for the efficient set of trades X, by choosing $S_i^* = X \cap \delta(i)$, it can be seen that $g(p) \geq \sum_i u_i(S_i^*, p) = \sum_i w_i(S_i^*)$. Thus, g is lower-bounded by the optimal welfare. Moreover, this lower bound is achieved whenever $u_i(S_i^*, p) = \max_{S \subset \delta(i)} u_i(S, p)$ for all i, or, equivalently, whenever p is a competitive equilibrium vector. Thus, when the algorithm converges, it converges to a competitive equilibrium.

The fact that the unit price updates given in the algorithm decrease $g(p)$ follows from the duality theory of M-convex optimization. The function $\max_{S \subset \delta(i)} u_i(S,p) = \max_{S \subset \delta(i)} w_i(S) - \sum_{e \in S \cap \delta_-(i)} p_e + \sum_{e \in S \cap \delta_+(i)} p_e$ is the convex conjugate of $-w_i$. Convex conjugates of $M^\natural$-convex functions, as well as their sums such as g, are *$L^\natural$-convex functions.* A function $h : \mathbb{Z}^n \to \mathbb{R}$ is L-convex if (i) for $p, q \in \mathbb{Z}^n$ we have $h(p)+h(q) \geq h(p \vee q)+h(p \wedge q)$, and (ii) $h(p+\mathbf{1}) = r+h(p)$ for some $r \in \mathbb{R}$. A function h is $L^\natural$-convex if $\tilde{h}(p_0, p) = h(p-p_0\mathbf{1})$ for some L-convex function $\tilde{h}$. $L^\natural$-convex functions have desirable properties. For instance, if p is not a minimizer of an $L^\natural$-convex function, then by jointly incrementing/decrementing a subset of the coordinates of p, the value of the function can be decreased (see Section 7 in Murota, 2003). Note that Algorithm 1 relies on such price updates. The details of convergence of this algorithm are standard, and hence omitted. Note that similar algorithms have also been used to determine competitive equilibria in two-sided markets (Sun and Yang, 2009; Murota, 2003).

I close this section by providing an approach for testing whether a given payoff vector can be supported in equilibrium. Formally, given $\sigma \in \mathbb{R}^N$, I investigate whether there exists a competitive equilibrium (X,p) such that $\sigma_i = u_i(X \cap \delta(i), p)$ for all $i \in N$.

Let X denote an efficient set of trades. Theorem 2.3.3 implies that (X,p) is a competitive equilibrium for any equilibrium price vector p. Since in a competitive equilibrium agents demand trades that maximize their payoff, it also follows that all competitive equilibria with price vector p share the same payoff vector; i.e., agent $i \in N$ receives the same payoff in all such equilibria. Thus, to characterize the set of payoff vectors that can be induced in equilibrium, it suffices to fix an efficient set of trades X and consider different equilibrium price vectors.

Recall that an efficient set of trades can be obtained by using the MSFP formulation. Let (x,y) denote an optimal solution of this problem, and let X denote the corresponding efficient set of trades. Consider the auxiliary network $F^{aux}(x,y)$ associated with (x,y), and recall that the reduced cost optimality conditions can be expressed as in (2.5) and (2.6). As discussed in Section 2.3, when the potential values of the special vertices in the flow network ($\{v_0^i\}_{i\in N}$) are set to zero, the potential values satisfying (2.5) and (2.6) characterize the set of all competitive equilibrium price vectors. In particular, given a solution π to this system by setting the price of any trade $e \in E$ to $p_e = \pi(v_e^{e^+})$ $(= \pi(v_e^{e^-}))$ yields a competitive equilibrium price vector and vice versa.

Using these observations it follows that the payoff vector σ can be supported in equilibrium if and only if the following system has a solution π:

$$\begin{aligned}
&w_i(X \cap \delta(i)) + \sum_{e \in X \cap \delta_+(i)} \pi(v_e^i) - \sum_{e \in X \cap \delta_-(i)} \pi(v_e^i) = \sigma_i \quad \forall i \in N \\
&\pi(v) - \pi(u) \le 0 \quad \forall (u,v) \in A^{aux}(x) \cup B^{aux}(x) \\
&\pi(v) - \pi(u) \le f(y - \chi^u + \chi^v) - f(y) \quad \forall (u,v) \in C^{aux}(y) \\
&\pi(v_0^i) = 0 \quad \forall i \in N,
\end{aligned} \tag{P}$$

where f is defined as in Section 2.3 (see the discussion following (2.3)).

Thus, I conclude that by first obtaining an optimal solution (x,y) to the MSFP formulation (and the associated efficient set of trades X), and then checking whether (P) admits a feasible solution (π), it can be determined whether σ constitutes a competitive equilibrium payoff vector or not. The latter step can be accomplished by solving a linear program. Moreover, both this linear program and the MSFP formulation can be solved in time that is polynomial in the number of trades in the economy. Therefore, I have established the following corollary:

Corollary 3.2.1. It is possible to check in time that is polynomial in $|E|$ whether a given vector $\sigma \in \mathbb{R}^N$ constitutes an equilibrium payoff vector.

3.2.2. Determining a Blocking Chain

Recall that an outcome is chain stable if it is individually rational and does not admit a blocking chain. I describe below a polynomial-time algorithm, Algorithm 2, for identifying a blocking chain for a given outcome or for certifying that none exists. This algorithm can also be used to test chain stability in polynomial time.

Fix an outcome (X, p^X). To verify individual rationality of this outcome it suffices to ensure that $u_i(X \cap \delta(i), p^X)$ is greater than $u_i((X \cap \delta(i)) \setminus \{e\}, p^X)$ for all $e \in X \cap \delta(i)$, and $u_i((X \cap \delta(i) \setminus \{e_1, e_2\}), p^X)$ for all $e_1 \in X \cap \delta_+(i)$, $e_2 \in X \cap \delta_-(i)$ and $i \in N$. This follows from Theorem 6.26 in Murota (2003), together with the fact that the value functions and hence $u_i(X \cap \delta(i), p^X)$ are $M^\natural$-concave. Thus, individual rationality can be checked by comparing $X \cap \delta(i)$ with polynomially many sets of trades incident to agent i, for all i.

Assume that outcome (X, p^X) is individually rational. Then, Algorithm 2 can be used to identify a blocking chain or certify that none exists. In the description of the algorithm the shorthand $(x, y) = R_{\bar{G}}(X)$ is used to denote the flow/net outflow that is consistent with a given set of trades X, in the flow network associated with a trading network $\bar{G}$.

Algorithm 2 proceeds in two phases. Phase one focuses on the contraction G^X (with respect to (X, p^X)), corresponding value functions $\{\hat{w}_i\}_{i \in N}$ (see (3.3)), and set of trades $\emptyset$. It uses the auxiliary network associated with G^X, and finds a minimal welfare-improving chain T if one exists. If none exists, then there is no blocking chain. Otherwise, the second phase returns the prices p^T, which together with T constitute a blocking chain for the empty outcome in G^X, and, equivalently, outcome (X, p^X) in G.

The algorithm relies on three functions: aux.construct(), $\mathrm{greedy}_{\mathrm{X}}()$, BellmanFord(). It starts phase one by considering the trading network $\bar{G} = G^X$, and flow $(x, y) = R_{\bar{G}}(\emptyset)$ in the corresponding flow network. Function aux.construct$(\bar{G}, (x, y))$ constructs the associated auxiliary network. The auxiliary network has at most $O(|N| + |E|)$ vertices (since it contains two vertices for each trade and a special vertex for each agent) and $O(|N|^2 + |E|^2)$ arcs

ALGORITHM 2: Determining a Blocking Chain

Input: Trading network $G = (N, E)$, valuations $\{w_i\}_{i \in N}$, and outcome (X, p^X).
Output: Blocking chain T with prices p^T.

```
\bar{G} = G^X; (x, y) = R_{\bar{G}}(\emptyset); F^{aux}(x, y) = aux.construct(\bar{G}, (x, y));
foreach (u, v) \in A^{aux}(x) \cup B^{aux}(x) do c^{aux}_{(u,v)}(x, y) = 0;
foreach i \in N do
    foreach u \in V^i do
        foreach v \in V^i do
            c^{aux}_{(u,v)}(x, y) = greedy_X(w_i, y^i) - greedy_X(w_i, y^i - \chi^u + \chi^v);
        end
    end
end
foreach u \in V do
    foreach v \in V do
        W[u, v, 1] = c^{aux}_{(u,v)}(x, y);
    end
end
for m = 2 \to |V| do
    foreach u \in V do
        foreach v \in V do
            W[u, v, m] = W[u, v, m - 1];
            foreach t \in V do
                W[u, v, m] = \min\{W[u, v, m], W[u, t, m - 1] + c^{aux}_{(t,v)}(x, y)\};
            end
        end
    end
end
Find smallest m such that W[u, u, m] < 0 for some u and negative cycle K from matrix of predecessors;
if not found then
    return No Blocking Chain;
else
    set T = \{e \in E | the corresponding arc in the auxiliary network belongs to K \cap A^{aux}(x)\};
end
---------------------------------------------------------------------------
\hat{G} = (N, T); (x, y) = R_{\hat{G}}(T); F^{aux}(x, y) = aux.construct(\hat{G}, (x, y)) ;
foreach (u, v) \in A^{aux}(x) \cup B^{aux}(x) do c^{aux}_{(u,v)}(x, y) = 0;
foreach i \in N do
    foreach u \in V^i do
        foreach v \in V^i do
            c^{aux}_{(u,v)}(x, y) = greedy_X(w_i, y^i) - greedy_X(w_i, y^i - \chi^u + \chi^v);
        end
    end
end
Set \Delta = \frac{1}{4|T|};
foreach a \in C^{aux}(y) do
    if a is adjacent to a special vertex then
        c^{aux}_a(x, y) = c^{aux}_a(x, y) - \Delta;
    else
        c^{aux}_a(x, y) = c^{aux}_a(x, y) - 2\Delta;
    end
end
Pick a special vertex s in F^{aux}(x, y);
d = BellmanFord(F^{aux}(x, y), s);
foreach e \in T do
    p^T[e] = d[v_e^{e^+}];
end
return (T, p^T);
```

(where the cardinality of $A^{aux}(x) \cup B^{aux}(x)$ is $O(|N|^2 + |E|)$, and the cardinality of $C^{aux}(y)$ is $O(|E|^2)$, since it is bounded by $\sum_{i \in N}(|\delta(i)| + 1)^2$). Note that traders who do not have any incident arcs cannot affect the stable outcome/blocking chains. Thus, in this analysis without loss of generality, such agents are omitted from the trading network, and I consider settings where $G = (N, E)$ is (weakly) connected[4] and satisfies $|E| \geq |N| - 1$. Hence, the

[4] If G is not connected then the algorithm can examine each connected component separately.

number of arcs in the auxiliary network is $O(|N|^2 + |E|^2) = O(|E|^2)$.

Algorithm 2 assigns costs to all arcs in $C^{aux}(y)$, using $\text{greedy}_{\text{X}}()$. As the auxiliary network is associated with the contraction G^X, these costs are a function of $\{\hat{w}_i\}_{i \in N}$. Algorithm 2 does not need to explicitly determine $\{\hat{w}_i\}_{i \in N}$. It suffices to evaluate these functions for each arc in the auxiliary network in $C^{aux}(y)$ to determine the arc costs. Specifically, recall that each arc $(u, v) \in C^{aux}(y)$ is such that $u, v \in V^i$ for some $i \in N$, and thus $c^{aux}_{(u,v)}(x, y) = f(y - \chi^u + \chi^v) - f(y) = \hat{w}_i'(y^i) - \hat{w}_i'(y^i - \chi^u + \chi^v)$, where $\hat{w}_i{}'$ is the M-concave function associated with $\hat{w}_i$ (recall the definition in (2.2)). The definition of $\{\hat{w}_i\}_{i \in N}$ in (3.3) and $M^\natural$-concavity of w_i imply that for a given y, it is possible to compute $\hat{w}_i'(y)$ with a greedy algorithm ($\text{greedy}_{\text{X}}()$) in polynomial[5] time (Shioura, 2004). The complexity of computing this quantity, and hence $c^{aux}_{(u,v)}(x, y)$ for each $(u, v) \in C^{aux}(y)$, is $O(|X|^2)$, which is bounded by $O(|E|^2)$. Thus, the overall complexity of computing all arc costs and constructing the auxiliary network is $O(|E|^4)$ (recall that the cardinality of $C^{aux}(y)$ is bounded by $O(|E|^2)$).

Recall that a negative cycle with the least number of arcs in the auxiliary network reveals a (minimal) chain of trades that improve welfare, which constitutes a blocking chain T (see Lemma 3.1.1 and Corollary 3.1.2). To find such a negative cycle, it suffices to compute the array W, whose (u, v, m) entry, $W[u, v, m]$, is the smallest length achieved by a directed path from vertex u to vertex v using at most m arcs. If $W[u, u, m]$ is nonnegative for each u and $m \leq |V|$, then the first phase of Algorithm 2 terminates by concluding that no negative cycle exists, and hence not executing any trades in G^X is efficient (Theorem 2.2.1). Thus, there is no blocking chain. Otherwise, the negative cycle with the least number of arcs is given by the element $W[u, u, m]$ which is negative for the smallest possible m. The arcs along the negative cycle can be found by keeping track of the arcs added to the shortest paths at the computation of W (i.e., the arcs (t, v) whose length determines $W[u, v, m]$ in line 20). In this case, the first phase of the algorithm terminates with a blocking chain T. Computing W and finding a blocking chain T (or establishing that it does not exist) takes

[5] In our case this greedy algorithm is strongly polynomial, since $\text{dom} f \subset \{-1, 0, 1\}^V$.

$O(|V|^4) = O((|N| + |E|)^4) = O(|E|^4)$ steps.

In phase two I construct the prices p^T accompanying the blocking chain T. The algorithm focuses on trading network $\hat{G} = (N, T)$, value functions $\{\hat{w}_i\}_{i \in N}$, and set of trades T, and constructs the prices following the approach in the proof of Corollary 3.1.2. Specifically, it first perturbs the value functions $\{\hat{w}_i\}_{i \in N}$ by $\Delta > 0$ (which is chosen to satisfy the conditions imposed in the proof of Corollary 3.1.2) to obtain valuations $\bar{w}_i(Z) = \hat{w}_i(Z) - \Delta|Z|$. Then, it computes a competitive equilibrium in the economy with respect to $\{\bar{w}_i\}_{i \in N}$. As in Corollary 3.1.2, a competitive equilibrium in this economy yields prices p^T such that under these prices, each agent i strictly demands $T \cap \delta(i)$ in G^X, thereby implying that (T, p^T) is a blocking chain for outcome (X, p^X) in G.

To compute the competitive equilibrium prices, the algorithm focuses on the flow network of $\hat{G}$, and the flow consistent with executing trades T. Then, it constructs the corresponding auxiliary network and its arc costs (according to $\{\bar{w}_i\}_{i \in N}$), which takes $O(|E|^4)$. To see this, note that arc costs according to $\{\bar{w}_i\}_{i \in N}$ and those according to $\{\hat{w}_i\}_{i \in N}$ are closely related.

In particular, since we start with a flow consistent with executing all trades T in $\hat{G}$, each arc in $C^{aux}(y)$ corresponds to dropping one existing trade (if incident to a special vertex) or two existing trades (such that an agent is a buyer in one and a seller in the other). In the former case (assuming trade e is dropped), the arc cost is given by:

$$\bar{w}_i(T) - \bar{w}_i(T \setminus \{e\}) = \hat{w}_i(T) - \Delta|T| - \hat{w}_i(T \setminus \{e\}) + \Delta(|T| - 1) = \hat{w}_i(T) - \hat{w}_i(T \setminus \{e\}) - \Delta.$$

Thus, the arc cost according to $\{\bar{w}_i\}_{i \in N}$ is obtained by subtracting Δ from the arc cost according to $\{\hat{w}_i\}_{i \in N}$. In the latter case, since the arc is not incident to a special vertex, the algorithm follows a similar approach and computes the arc costs by subtracting 2Δ from the arc costs associated with $\{\hat{w}_i\}_{i \in N}$. Instead of constructing the functions $\{\hat{w}_i, \bar{w}_i\}_{i \in N}$ explicitly, the algorithm relies on $\text{greedy}_{\text{X}}()$ to evaluate the arc costs according to $\{\hat{w}_i\}_{i \in N}$.

This step has complexity $O(|E|^4)$. Then, it perturbs these by Δ and 2Δ as appropriate, to construct the arc costs according to $\{\bar{w}_i\}_{i\in N}$. Since the cardinality of $C^{aux}(y)$ is $O(|E|^2)$, the overall complexity of constructing the auxiliary network and accompanying arc costs according to $\{\bar{w}_i\}_{i\in N}$ is $O(|E|^4)$.

Given the auxiliary network and arc costs, if potential values satisfying conditions (2.5) and (2.6) are known, they will be equilibrium prices p^T. Such potential values can be obtained by solving a linear program. Alternatively, consider the shortest distances from a special vertex to all vertices of the auxiliary network. Observe that these distances readily satisfy conditions (2.5) and (2.6), and hence give p^T. For an auxiliary network $F^{aux}(x, y)$ and (special) vertex s, the function BellmanFord($F^{aux}(x, y), s$) of the algorithm computes the shortest path distances on $F^{aux}(x, y)$ from vertex s using the Bellman-Ford algorithm. Constructing the prices following this approach has complexity $O(|V||A|) = O(|E|^3)$.

In sum, the overall complexity of the algorithm is $O(|E|^4 + |E|^4 + |E|^3) = O(|E|^4)$. Hence, using Algorithm 2 it is possible to identify a blocking chain or to certify that none exists in polynomial time. Since individual rationality can also be checked in polynomial time, my results imply that for trading networks chain stability can be tested in polynomial time.

3.3. Comparative Statics

The study of how the solutions of an economic model change as its parameters are changed is important because most of the testable predictions of a model are comparative statics predictions. Here, I first characterize how competitive equilibria change if (i) a new trade or (ii) a new buyer is added to the underlying economy. I build on these results to characterize how the addition of a collection of trades (possibly between more than two traders) may change the equilibrium. Finally, I discuss possible applications of my comparative statics. My characterization exploits how optimal flow/potential values change as the parameters of the corresponding MSFP are modified.

Sensitivity results are available for the classic minimum-cost network flow problems (see, e.g, Ahuja et al. (1993)). I provide a similar result for MSFP using the reduced costs associated with a given optimal flow and potential function.

Lemma 3.3.1. Consider a network $F = (V, A)$ and let $\hat{a}$ be an arc with zero capacity, i.e., $\underline{k}_{\hat{a}} = \overline{k}_{\hat{a}} = 0$. Let (x, y) and π respectively denote an optimal solution of the MSFP and vertex potential values that satisfy the optimality conditions in Theorem 2.2.1. Construct the auxiliary network $F^{aux}(x, y)$, and let $d : V \to \mathbb{R}$ be a function such that $d(v)$ is the length of the shortest path from $\hat{a}^-$ to $v \in V$ in $F^{aux}(x, y)$, where the length of each arc is given by the reduced cost under π.

Consider the instance of MSFP when $\overline{k}_{\hat{a}}$ is increased to one. For this problem an optimal solution $(\hat{x}, \hat{y})$ and an associated potential function $\hat{\pi}$ that satisfies the optimality conditions of Theorem 2.2.1 can be obtained as follows:

- Set $\hat{\pi}(v) = \pi(v) + d(v)$ for all $v \in V$.
- If $c^{\pi}_{\hat{a}} + d(\hat{a}^+) \geq 0$, set $(\hat{x}, \hat{y}) = (x, y)$. Otherwise, let P be a shortest path (with respect to the reduced costs) from $\hat{a}^-$ to $\hat{a}^+$ in $F^{aux}(x, y)$ with the least number of arcs. Obtain $\hat{x}$ by augmenting the flow x in F along P, and sending one unit of flow on arc $\hat{a}$. Define $\hat{y}$ as the associated net outflow.

Proof. Theorem 2.2.2 implies that in F, the vertex potential function π together with the optimal solution (x, y) of the MSFP satisfy the reduced-cost optimality conditions. Let $\hat{F}$ denote the network obtained after increasing the capacity of arc $\hat{a}$ in F by one unit. Observe that the optimal flow (x, y) of F continues to be feasible for $\hat{F}$ (since the capacities of arcs do not decrease). Let $\hat{F}^{aux}(x, y)$ denote the auxiliary network associated with $\hat{F}$, and (x, y). Note that $\hat{F}^{aux}(x, y)$ is obtained from $F^{aux}(x, y)$ by including arc $\hat{a}$. Thus, it follows that (x, y, π) satisfies the reduced-cost optimality conditions for all arcs in $\hat{F}^{aux}(x, y)$ except $\hat{a}$.

Case 1 ($c^{\pi}_{\hat{a}} + d(\hat{a}^+) \geq 0$): Construct a new potential function as defined in the statement of the lemma: $\hat{\pi}(v) = \pi(v) + d(v)$ for all $v \in V$. Observe that for any arc a in $F^{aux}(x, y)$, we have $d(a^-) \leq d(a^+) + c^{\pi}_{a}$, i.e., the shortest path from $\hat{a}^-$ to a^- is at most the length of the shortest path to a^+ plus the length of arc a. This implies that for all arcs $a \neq \hat{a}$ in $\hat{F}^{aux}(x, y)$, the reduced costs under the new potential function, $c_a^{\hat{\pi}} = c_a^{\pi} + d(a^+) - d(a^-)$, remain nonnegative. As for arc a, the new reduced cost is nonnegative according to the hypothesis of this case (since $c_{\hat{a}}^{\hat{\pi}} = c_{\hat{a}}^{\pi} + d(\hat{a}^+)$). Thus, (x, y) and $\hat{\pi}$ satisfy the reduced-cost optimality conditions in $\hat{F}$. Therefore, by Theorem 2.2.2, (x, y) is an optimal solution of the MSFP, and together with potential function $\hat{\pi}$ it also satisfies the optimality conditions of Theorem 2.2.1.

Case 2 ($c_{\hat{a}}^{\pi} + d(\hat{a}^+) < 0$): Update the potential function as in Case 1. Define a new solution $(\hat{x}, \hat{y})$ to the MSFP as described in the statement of the lemma. Denote by $\hat{A}^{aux}(x)$, $\hat{B}^{aux}(x)$, and $\hat{C}^{aux}(y)$, the subsets of the arcs in the auxiliary network $\hat{F}^{aux}(x, y)$ (as defined in Section 2.2). As in the first case, under the new potential function $\hat{\pi}$ any arc $a \neq \hat{a}$ in $\hat{F}^{aux}(x, y)$ has a nonnegative reduced cost. Consider the new auxiliary network $\hat{F}^{aux}(\hat{x}, \hat{y})$ obtained after the flow is updated, and define the associated reduced costs with respect to the potential function $\hat{\pi}$.

First focus on the arcs in $\hat{A}^{aux}(\hat{x}) \cup \hat{B}^{aux}(\hat{x})$. Following a similar approach to Case 1, it can be shown that the reduced costs of arcs that were also present previously, i.e., in $\hat{A}^{aux}(x) \cup \hat{B}^{aux}(x)$, remain nonnegative. It follows from the construction of the auxiliary network that the new arcs in $\hat{F}^{aux}(\hat{x}, \hat{y})$ (that do not belong to $\hat{F}^{aux}(x, y)$) are those obtained after reversing the arcs in $P \cup \{\hat{a}\}$. Observe that arc $\hat{a}$ is only present with an opposite orientation, since in $\hat{x}$ one unit of flow is sent on arc $\hat{a}$. It can be readily seen that the reduced-cost optimality condition on this arc, hereafter $-\hat{a}$, is now satisfied since it is given by $c_{-\hat{a}}^{\hat{\pi}} = -(c_{\hat{a}}^{\pi} + d(\hat{a}^+)) > 0$, where the inequality follows from the assumption in this case. For each arc a in $P \cap (\hat{A}^{aux}(x) \cup \hat{B}^{aux}(x))$, let $-a$ denote the arc obtained after reversing a, and recall that $-a \in \hat{A}^{aux}(\hat{x}) \cup \hat{B}^{aux}(\hat{x})$. Since arc a is on the shortest path between $\hat{a}^-$ and $\hat{a}^+$, we have $d(a^-) = d(a^+) + c_a^{\pi}$. Consequently, $c_{-a}^{\hat{\pi}} = -c_a^{\hat{\pi}} = -(c_a^{\pi} + d(a^+) - d(a^-)) = 0$. Thus, it follows that all arcs in $\hat{A}^{aux}(\hat{x}) \cup \hat{B}^{aux}(\hat{x})$ have nonnegative reduced cost.

To conclude the proof I establish that the reduced costs of arcs in $\hat{C}^{aux}(\hat{y})$ are positive using the following lemma:

Lemma 3.3.2 (Lemma 3.2 in Moriguchi and Murota (2003))**.** Suppose that $y \in \arg\min_z f(z) - p \cdot z$ and that $(u_1, v_1), (u_2, v_2), \ldots, (u_r, v_r) \in C^{aux}(y)$ have distinct end vertices. If $(u_i, v_i) \in C^{aux}(y) \cap \{a | c_a^p = 0\}$ for $i = 1, \ldots, r$ and $(u_i, v_j) \notin C^{aux}(y) \cap \{a | c_a^p = 0\}$ for any $i < j$, then $\hat{y} = y + \sum_{i=1}^{r} (\chi^{v_i} - \chi^{u_i}) \in \arg\min_z f(z) - p \cdot z$.

Given flows x and $\hat{x}$ in $\hat{F}$, focus on the associated outflow y and $\hat{y}$, and the corresponding sets of arcs $\hat{C}^{aux}(y)$ and $\hat{C}^{aux}(\hat{y})$. The hypotheses of this lemma can be verified by setting $p_v = \hat{\pi}(v)$ for all $v \in V$.

First, observe that under $\hat{\pi}$ the reduced cost of any arc in $F^{aux}(x, y)$ is nonnegative. Hence, under $\hat{\pi}$ any arc $a \neq \hat{a}$ in $\hat{F}^{aux}(x, y)$ has a nonnegative reduced cost, and I conclude that the reduced costs of all arcs in $\hat{C}^{aux}(y)$ are nonnegative. Note that the reduced cost of an arc $(u, v) \in \hat{C}^{aux}(y)$ can be expressed as follows: $c^{\hat{\pi}}_{(u,v)} = \bar{f}(y + \chi^v - \chi^u) - \bar{f}(y) \geq 0$, where $\bar{f}(y) = f(y) - \sum_{v \in V} \hat{\pi}(v) y_v$. Since local optimality implies global optimality for M-convex functions (see Murota (2003)), and $\hat{C}^{aux}(y)$ consists of all (u, v) for which $f(y + \chi^v - \chi^u)$ is bounded, it follows that $y \in \arg\min_{z \in \mathbb{Z}^{|V|}} f(z) - \sum_{v \in V} \hat{\pi}(v) z_v$.

Second, all arcs in $P \cap \hat{C}^{aux}(y)$ have distinct end vertices. To show this, I examine how P was formed in the auxiliary network $F^{aux}(x, y)$ with potential function π. The shortest path P cannot go through a vertex twice, since all reduced costs are nonnegative. In addition, there are no consecutive arcs $(s, u) - (u, t)$ in $P \cap C^{aux}(y)$, since by M-convexity the reduced cost of the shortcut (s, t) is less than the sum of the reduced costs in (s, u) and (u, t), which would have revealed that there exists a shortest path from $\hat{a}^-$ to $\hat{a}^+$ with fewer arcs than P. Formally, let $c^{aux}_{(u',v')}(x, y)$ be the cost of arc (u', v') in $F^{aux}(x, y)$. Using the definition in (2.1) and directly applying M-EXC[$\mathbb{Z}$] on f, it can be seen that the triangle inequality holds for the arc costs of the auxiliary network, i.e.,

$$c^{aux}_{(s,t)}(x, y) \leq c^{aux}_{(s,u)}(x, y) + c^{aux}_{(u,t)}(x, y). \tag{3.4}$$

It follows that the triangle inequality holds for the reduced costs as well:

$$\begin{aligned}
c^{\pi}_{(s,u)} + c^{\pi}_{(u,t)} &= c^{aux}_{(s,u)}(x,y) + \pi(s) - \pi(u) + c^{aux}_{(u,t)} + \pi(u) - \pi(t) \\
&= c^{aux}_{(s,u)} + c^{aux}_{(u,t)} + \pi(s) - \pi(t) \\
&\geq c^{aux}_{(s,t)} + \pi(s) - \pi(t) \\
&= c^{\pi}_{(s,t)}.
\end{aligned}$$

As explained above, we have $c^{\hat{\pi}}_a = 0$ for all $a \in P \cap \hat{C}^{aux}(y)$. Now, suppose that $P \cap \hat{C}^{aux}(y) = \{(u_1, v_1), \ldots, (u_r, v_r)\}$, where the indexing is chosen with respect to the order of arcs in P. It readily follows that there is no shortcut arc $(u_i, v_j) \in \hat{C}^{aux}(y)$ such that $i < j$ and $c^{\hat{\pi}}_{(u_i,v_j)} = 0$, since this would imply that $c^{\pi}_{(u_i,v_j)} = d(v_j) - d(u_i)$, which raises a contradiction by revealing that in $F^{aux}(x,y)$ with potential function π there exists a shortest path from $\hat{a}^-$ to $\hat{a}^+$ with fewer arcs than P.

Thus, Lemma 3.2 in Moriguchi and Murota (2003) guarantees that $\hat{y} \in \arg\min_{z \in \mathbb{Z}^{|V|}} f(z) - \sum\limits_{v \in V} \hat{\pi}(v) z_v$. Global optimality implies local optimality, and I conclude that $c^{\hat{\pi}}_a \geq 0$ for all $a \in \hat{C}^{aux}(\hat{y})$. Since I have also established that the reduced costs of arcs in $\hat{A}^{aux}(\hat{x})$ and $\hat{B}^{aux}(\hat{x})$ are nonnegative, by Theorem 2.2.2 the optimality of $(\hat{x}, \hat{y})$ follows. Theorem 2.2.2 also implies that this optimal solution paired with the potential function $\hat{\pi}$ satisfy the optimality conditions of Theorem 2.2.1, as claimed. □

This lemma implies that if a unit capacity arc is added to an instance of MSFP, the optimal flow, as well as the supporting potential function, change in a predictable way. Specifically, either the initial flow remains optimal, or a new optimal flow is obtained by augmenting the flow on the new arc as well as the path connecting the end points of the aforementioned arc, in the auxiliary network. Recall that in the MSFP formulation of Section 2.3, arcs correspond to possible trades between the agents, and optimal flow provides equilibrium trades. Thus, the change in equilibrium trades as a result of introducing new trades into an economy can be characterized by using Lemma 3.3.1.

3.3.1. Addition of a New Trade

I start by characterizing the change in competitive equilibrium trades with the introduction of a single new trade. I establish that after the introduction of the new trade, the initial competitive equilibrium trades may still constitute an equilibrium. If not, the new competitive equilibrium trades can be obtained by identifying a chain of trades in a slightly modified trading network, and (i) including among equilibrium trades the trades of this chain that do not belong to the initial equilibrium and (ii) removing the remaining trades associated with the arcs in the chain from the set of equilibrium trades.

Theorem 3.3.3. Let X be a set of equilibrium trades in G, and G' be a new trading network obtained after adding a new trade e to G. Denote by G_{res} the trading network obtained from G' after reversing the orientations of the arcs in X. Either X continues to be a set of equilibrium trades in G' or there exists a chain $C \ni e$ of trades in G_{res} such that the new set of equilibrium trades is given by $X' = C_f \cup X \setminus C_r$, where C_r denotes the arcs in G that are present in C with opposite orientations, and C_f denotes the remaining arcs in C.

Proof. Since the value functions are $M^\natural$-convex, the trading network can be transformed to a flow network. I modify the internal network of each agent by introducing an additional vertex for each agent/trade vertex in the flow network, where the agent is the buyer in this trade. Call the new vertices terminal vertices. Each arc in the original flow network corresponding to a trade is replaced by two arcs: one arc from the corresponding vertex of the seller to the terminal vertex, and the other arc from the terminal vertex to the corresponding vertex of the buyer. The former arc is uncapacitated, whereas the latter one has a capacity of either 0 or 1 that encodes its absence or presence in the network. Nonzero net outflow at terminal vertices has an infinite penalty (i.e., flow conservation is imposed at these vertices), whereas the net outflow at the remaining vertices has penalties associated with agents' valuations, as before.

An illustration can be found in Figure 5. The internal network of an agent, say i, appears in (a). The dashed vertices correspond to trades where agent i is a buyer, the solid vertex corresponds to a trade where i is a seller, and the dotted vertex is a special vertex. The network in (b) depicts the transformed network.

Traders incident to e are not allowed to execute trade e in the original network. Thus, in the corresponding flow network the terminal vertex and the seller's vertex associated with trade e are connected, but the buyer of the trade has a zero-capacity arc connecting the terminal vertex associated with trade e to the rest of her internal network. Hence, initially, no flow is sent on the arcs associated with trade e in the flow network, and the efficient set of trades in G does not involve e. Let a_0 denote the arc in the flow network between the seller's vertex associated with trade e and the associated terminal vertex. Similarly, denote by a the arc between the terminal vertex and the vertex corresponding to the buyer.

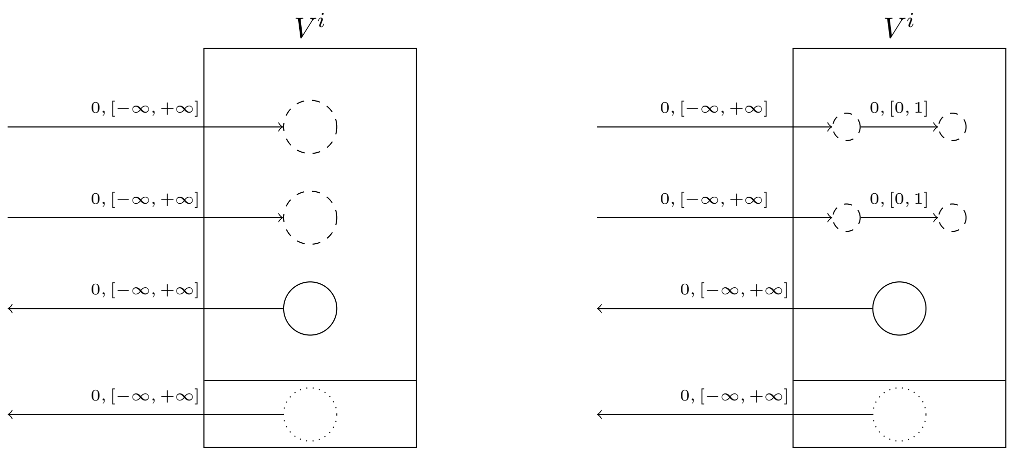

Figure 5: (a) An Agent's Internal Network. (b) The Agent's Modified Network.

Fix a competitive equilibrium (X, p) in G. Denote the corresponding flow/net outflow in the flow network by (x, y), which is obtained by sending one unit of flow on arcs that correspond to trades in X. This is similar to the previous construction, except that we have two arcs associated with each trade (due to the modification of the internal networks) and both carry the same flow. For any trade f in G, set the potential values of the associated vertices equal to p_f, the potential values of $a_0^+, a_0^- = a^+$ equal to $-M$, and the potential value of a^- equal to M, for some $M \gg 0$. Let π denote this potential function. As before, it readily follows from the equilibrium conditions that (x, y) and π satisfy the reduced-cost optimality conditions.

Next, assume that a new trade e is added to the economy. In the corresponding flow network, increase the capacity of arc a from zero to one. Lemma 3.3.1 establishes that a new optimal flow can be obtained from the original flow. There are two cases to consider. In the first case, the original flow continues to be optimal. In the second case, an optimal flow is obtained by sending a unit of flow on arc a, and augmenting the original flow by sending one unit of flow from a^- to a^+ via the shortest path with the least number of arcs between these vertices in the auxiliary network, as described in the lemma. Observe that this cycle traverses at most two special vertices, since these vertices are connected by uncapacitated zero-cost arcs. In other words, the new optimal flow is obtained by augmenting the original flow along an improvement cycle that has at most one arc between the special vertices. In both cases, the potential function updated as in Lemma 3.3.1 satisfies the reduced-cost optimality conditions together with the updated flow. Thus, as before, the corresponding trades/prices constitute a competitive equilibrium.

In the first case, the claim readily follows from these observations by setting $C = \emptyset$. To prove the claim in the second case, first assume that no special vertices are visited. Then, updating the flow along a cycle in the flow network corresponds to updating trades along a cycle C in G_{res}, by executing some previously unexecuted trades, captured by $C_f \ni e$, in the original network, and by dropping some executed trades (in cases where flow is sent in the opposite direction through the auxiliary network) captured by C_r. If a single special vertex is visited, the update structure is the same. If multiple special vertices are visited, the update structure is similar, but the improvement cycle can be viewed as originating and terminating at the special vertices. Hence, updating the flow along this cycle corresponds to updating trades along a chain C in G_{res} that does not constitute a cycle. Moreover, in both cases, the improvement cycle sends flow on arc a, thereby executing trade e. Hence, as claimed, the new set of equilibrium trades can be given by $X' = C_f \cup X \setminus C_r$, for some chain C such that $e \in C$. □

Suppose that a new trading opportunity emerges. A priori it is unclear how the equilibrium trades would change and whether this change would exhibit any meaningful structural properties. Surprisingly, my result implies that the change in equilibrium trades can be captured in terms of chains of trades in the underlying network. This result exploits the MSFP formulation of the problem of finding the efficient allocation and the sensitivity analysis result for MSFP (Lemma 3.3.1). It highlights the power of using the network flow formulation and its properties for obtaining novel comparative statics. In the next section, this result is used as a building block to obtain more detailed comparative statics.

3.3.2. Addition of a New Buyer

Next, a new buyer is added to the economy with a set of possible trades Δ and a value function $v : 2^{\Delta} \to \mathbb{R}$. By sequentially introducing these trades the change in the set of equilibrium trades can be characterized as in Theorem 3.3.3. In this section, I characterize the change in equilibrium prices caused by adding a new buyer. The main result of this section is that when a new *buyer* is introduced into the economy, the prices of *all* prior trades (weakly) increase. I subsequently characterize the payoff implications of this change on the equilibria.

When a new buyer enters the economy there are two effects. First, the traders she is connected to enjoy higher demand, and start offering higher prices for all the trades in which they participate as sellers. On the other hand, the addition of a new buyer increases the competition between the traders serving her. This may lead to a decrease in some prices. Theorem 3.3.4 below shows that the first effect dominates. What about the prices of trades in which these traders participate as buyers? Surprisingly, the price increase is not limited to the traders adjacent to the new buyer, but extends to the rest of the network. This is due to the full substitutability of preferences, which guarantees that when the set of contracts in which an agent participates as a seller expands, the agent starts demanding more contracts in which she participates as a buyer. This generates an increase in the equilibrium prices of the corresponding trades.

Theorem 3.3.4. Assume that (X, p) is a competitive equilibrium in trading network G. Assume that a new agent b joins the trading network, as a buyer of a set of trades Δ whose sellers belong to G. Denote the resulting trading network by G'.

There exists a competitive equilibrium (X', p') in G', such that $p'_e \geq p_e$ for all trades e not incident to b.

Proof. Let v denote the value function of buyer b, X' denote a set of welfare-maximizing trades in G' once b is introduced to the economy, and $S^* = X' \cap \Delta$ denote the set of trades executed by agent b at X'.

To derive competitive equilibrium prices p' supporting X' in the economy with agent b, I first characterize how the valuation and payoff of agent b change due to *unit trade deviations*[6] from S^*, i.e., due to (i) adding a new trade to S^*, (ii) removing an existing trade from S^*, and (iii) executing both steps simultaneously.

The proof has three steps.

1. I construct an alternative economy, $\hat{G}$, where b is replaced with another buyer $\hat{b}$, with the same set of possible trades, and value function $\hat{v} : 2^{\Delta} \to \mathbb{R}$ such that $\hat{v}(S^*) = v(S^*)$. Buyer $\hat{b}$'s value function *is consistent with b in terms of unit trade deviations from S^**. That is, changes in $\hat{v}$ and v due to unit trade deviations coincide.

2. It is shown that X' constitutes an efficient set of trades in economy $\hat{G}$. Moreover, any equilibrium price vector $\hat{p}$ supporting X' in the new economy, is also an equilibrium price vector for G'.

3. It is shown that there exists an equilibrium price vector $\hat{p}$ in $\hat{G}$, where prices of all

[6] For $M^\natural$-concave value functions, local optimality implies global optimality: if such unit trade deviations do not decrease agent b's payoff, then the bundle S^* is payoff-maximizing for b.

trades not incident to agent $\hat{b}$ are weakly higher when compared to an equilibrium price vector p in G.

Note that the last two steps jointly imply that $\hat{p}$ is an equilibrium price vector for economy G', with weakly higher prices for all pre-existing trades.

Step 1: I define the value function $\hat{v}$ of buyer $\hat{b}$ in terms of the cost of a minimum-cost flow in a network $H = (V_H, A_H)$, hereafter referred to as the *valuation network*. Each $e \in \Delta$ gives rise to three vertices v_e^{in}, v_e^1, v_e^0. The entire set of vertices in V_H is given by $V_H = \{v^{out}\} \cup V_{in} \cup V_1 \cup V_0$, where $V_{in} = \{v_e^{in} | e \in \Delta\}, V_1 = \{v_e^1 | e \in \Delta\}$, and $V_0 = \{v_e^0 | e \in \Delta\}$. The vertices in $\{v^{out}\} \cup V_{in}$ correspond to the vertices of the internal network of buyer $\hat{b}$ (see Section 2.3), and are referred to as the terminal vertices of the valuation network. On the other hand, V_1, V_0 are additional vertices whose incident arcs have costs that encode agent $\hat{b}$'s value function.

The set of arcs, A_H, is such that for each trade $e \in \Delta$, we have:

(a) a directed arc (v_e^{in}, v_e^1) with capacity 1 and zero cost,

(b) a directed arc (v_e^1, v_e^0), with capacity 1 and cost $-v(S^*) + v(S^* \setminus \{e\})$ if $e \in S^*$, and cost $-v(S^* \cup \{e\}) + v(S^*)$ otherwise, and

(c) a directed arc (v_e^0, v^{out}) with capacity $k_{(v_e^0, v^{out})} = 1$ and zero cost.

In addition, for each set of trades $e, e' \in \Delta$ such that $e \notin S^*, e' \in S^*$, we have:

(d) a directed arc $(v_e^1, v_{e'}^1)$ with capacity 1 and cost $-v(S^* \cup \{e\} \setminus \{e'\}) + v(S^*)$.

The valuation $\hat{v}(S)$ of agent $\hat{b}$ for a set $S \subset \Delta$ of trades is given by the negative of the minimum cost flow in H where each v_e^{in} for $e \in S$ has a supply of one unit, and the vertex v^{out} demands $|S|$ units (and flow conservation is satisfied). Figure 6 depicts an example of the construction.

In this construction, arc costs of the valuation network, and hence the value function $\hat{v}$ are defined in terms of the value function v of agent b. The construction guarantees that $\hat{b}$ is consistent with b in terms of unit trade deviations from S^*. To see this, first note that any feasible flow on H associated with $S \subset S^*$ (i.e., that imposes one unit of supply at each v_e^{in} for $e \in S$) uses all arcs (v_e^1, v_e^0) for $e \in S$ to capacity. This is because for $e \in S^*$, vertices v_e^1 each have a single outgoing arc with capacity one. Similarly, any feasible flow in H associated with $S \supset S^*$ uses all arcs (v_e^1, v_e^0) for $e \in S$ to capacity, since all vertices v_e^1 for $e \in S \setminus S^*$ have outgoing arcs only to v_e^0 or a vertex $v_{e'}^1$ for $e' \in S^*$ (and the outgoing arcs of the latter are used to capacity). These observations, together with the description of $\hat{v}(S)$, imply that for $S \subset S^*$ and $S \supset S^*$, we have

$$\hat{v}(S) = -\left(\sum_{e \in S \cap S^*} (-v(S^*) + v(S^* \setminus \{e\})) + \sum_{e \in S \setminus S^*} (v(S^*) - v(S^* \cup \{e\})) \right).$$

Hence, for $e \notin S^*$ and $e' \in S^*$ we have $\hat{v}(S^*) - \hat{v}(S^* \cup \{e\}) = v(S^*) - v(S^* \cup \{e\})$ and $\hat{v}(S^*) - \hat{v}(S^* \setminus \{e'\}) = v(S^*) - v(S^* \setminus \{e'\})$; i.e., changes in $\hat{v}$ coincide with changes in v in terms of adding/removing a single trade.

Similarly, consider $S = S^* \cup \{e\} \setminus \{e'\}$ for $e \notin S^*$ and $e' \in S^*$. As before, any feasible flow associated with S in H uses all arcs $(v_{e''}^1, v_{e''}^0)$ for $e'' \in S^* \setminus \{e'\}$. Moreover, a feasible flow routes the supply at v_e^{in} either along the path $(v_e^{in}, v_e^1) - (v_e^1, v_e^0) - (v_e^0, v^{out})$ or along the path $(v_e^{in}, v_e^1) - (v_e^1, v_{e'}^1) - (v_{e'}^1, v^{out})$ (since v_e^1 have outgoing arcs either to v_e^0 or $v_{e'}^1$). These paths respectively have costs $v(S^*) - v(S^* \cup \{e\})$ and $v(S^* \setminus \{e'\}) - v(S^* \cup \{e\} \setminus \{e'\})$. Due to the $M^\natural$-concavity of v, it follows that the latter is smaller, and hence the minimum cost flow under S continues to send flow on arc $(v_e^1, v_{e'}^1)$, while also utilizing arc (v_e^1, v_e^0). Thus, I conclude that $\hat{v}(S) = \hat{v}(S^*) - (-v(S^* \cup \{e\} \setminus \{e'\}) + v(S^*)) = \hat{v}(S^*) + v(S) - v(S^*)$, and that $\hat{v}$ is consistent with v in terms of unit trade deviations.

In order to find the efficient set of trades in the new economy $\hat{G}$, consider the corresponding flow network, denoted hereafter by $\hat{F}$. To obtain $\hat{F}$, construct the flow network as in previous sections, by connecting vertices that belong to the internal network of a trade's buyer/seller via uncapacitated arcs. Similarly connect the special vertices so that between any pair of special vertices there is exactly one uncapacitated directed arc. By convention I assume that all arcs connecting agent $\hat{b}$'s special vertex to other special vertices are outgoing. This construction involves vertices $\{v^{out}\} \cup V_{in}$ for agent $\hat{b}$ in the flow network. For this agent, also add the remaining vertices and arcs in the valuation network (which contains the vertices of the internal network as a subnetwork). As before, for each agent other than $\hat{b}$, I associate a penalty function with the net outflow from the vertices of the internal network. By contrast, for vertices of the valuation network of agent $\hat{b}$, I impose flow conservation. Note that the valuation of agent $\hat{b}$ for a set of incident trades is still captured through the arc costs in her internal network. Thus, the efficient set of trades can be found by solving an M-convex submodular flow problem in $\hat{F}$.

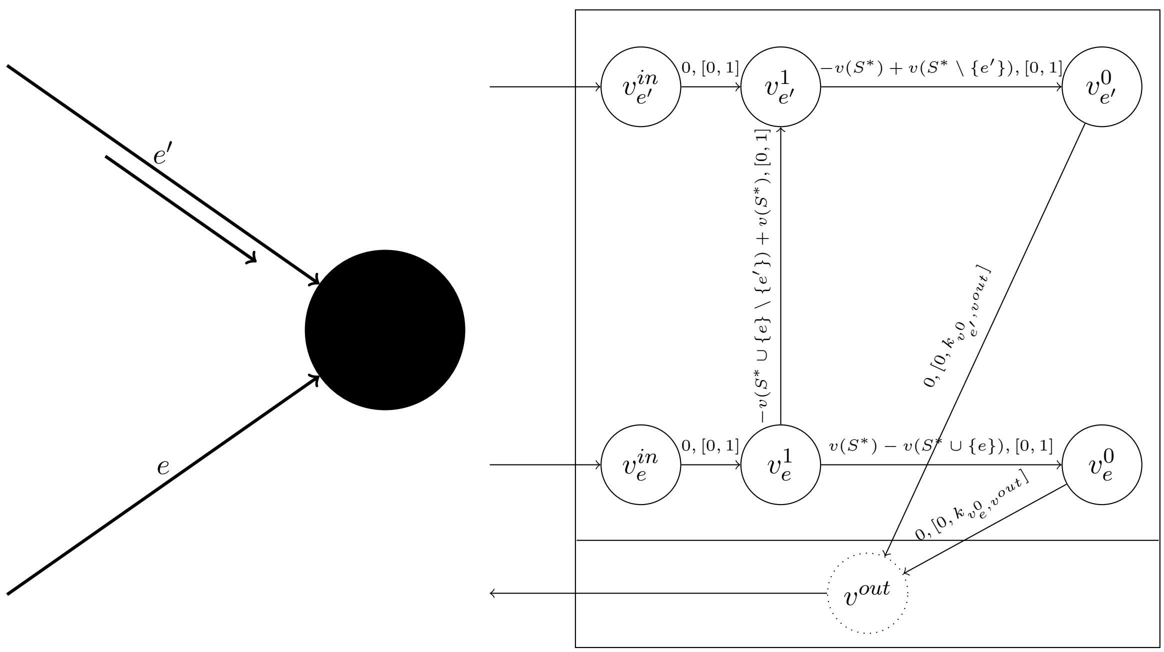

Figure 6: (a) Buyer b with Two Trades $e \notin S^*, e' \in S^*$ (b) Buyer $\hat{b}$'s Valuation Network. The vertices $v_e^{in}, v_{e'}^{in}$ are connected to the sellers' vertices associated with trades e and e' in the underlying trade network.

Step 2: Consider a competitive equilibrium (X', p') in G', and the corresponding optimal flow and potential function (x', π') in the associated flow network, hereafter F'. As before, π' is normalized such that it is zero on special vertices. Observe that F' has the same set of vertices and arcs as $\hat{F}$, other than the arcs and vertices of the valuation network of agent $\hat{b}$. Consider a flow-potential function pair $(\hat{x}, \hat{\pi})$ in $\hat{F}$ such that it coincides with (x', π') on all arcs and vertices common to F'. By construction, the reduced-cost optimality conditions are satisfied by arcs in $\hat{F}$, other than the arcs in the valuation network of $\hat{b}$. I now argue that the flows and potential values on arcs and vertices of $\hat{b}$'s valuation network can be set such that flow conservation and the reduced-cost optimality conditions are satisfied by them as well. The construction will explicitly show that $\hat{x}$ is optimal and corresponds to the same set of trades X' in $\hat{G}$.

Set the flow $\hat{x}$ within $\hat{b}$'s valuation network as follows:

- For each trade $e' \in S^*$, one unit of flow is sent along the path $(v_{e'}^{in}, v_{e'}^1) - (v_{e'}^1, v_{e'}^0) - (v_{e'}^0, v^{out})$.
- For each trade $e \notin S^*$, set flow on path $(v_e^{in}, v_e^1) - (v_e^1, v_e^0) - (v_e^0, v^{out})$ to zero.
- Set the flow on arcs $(v_e^1, v_{e'}^1)$ to zero.

Recall that each vertex $v_{e'}^{in}$ has one unit of incoming flow if $e' \in S^*$ and zero unit of incoming flow if $e' \notin S^*$, since for arcs common to F', the flow in $\hat{F}$ is set consistently with F'. It is easily verified that this construction satisfies flow conservation at the vertices of $\hat{b}$'s valuation network.

Set the potential values within $\hat{b}$'s valuation network as follows:

- For each trade $e \in \Delta$, set $\hat{\pi}(v_e^{in}) = p'_e$, $\hat{\pi}(v_e^1) = p'_e$, and $\hat{\pi}(v_e^0) = 0$.
- Set $\hat{\pi}(v^{out}) = 0$.

Now it can be verified that the constructed flow and potential values satisfy the reduced-cost optimality conditions on the arcs in the valuation network of $\hat{b}$. Consider the auxiliary network associated with $\hat{F}$ under flow $\hat{x}$. Observe that for any $e \in \Delta$, we have $c^{\hat{\pi}}_{(v_e^{in}, v_e^1)} = 0$ since the end points of these arcs have the same potential value by construction, and the underlying arc cost is zero. Similarly, it can be seen that $c^{\hat{\pi}}_{(v^0_e, v^{out})} = 0$. Thus, regardless of the orientation of the arcs (v^{in}_e, v^1_e), (v^1_e, v^{in}_e), (v^0_e, v^{out}), and (v^{out}, v^0_e) in the auxiliary network, the reduced-cost optimality conditions are satisfied. It remains to consider the reduced costs of arcs whose end points belong to $V_0 \cup V_1$:

- $e' \in S^*$: The constructed solution sends one unit of flow from $v^1_{e'}$ to $v^0_{e'}$, and hence the auxiliary network has an arc from $v^0_{e'}$ to $v^1_{e'}$. The associated reduced cost satisfies $c^{\hat{\pi}}_{(v^0_{e'}, v^1_{e'})} = v(S^*) - v(S^* \setminus \{e'\}) - p'_{e'} \geq 0$, since competitive equilibrium conditions imply that $v(S^*) - p'_{e'} \geq v(S^* \setminus \{e'\})$; i.e., in equilibrium (x', p') of G' agent b has no incentive to drop trade $e' \in S^*$.

- $e \notin S^*$: The constructed solution does not utilize the arc from v^1_e to v^0_e, and hence the auxiliary network has an arc from v^1_e to v^0_e. The associated reduced cost satisfies $c^{\hat{\pi}}_{(v^1_e, v^0_e)} = v(S^*) - v(S^* \cup \{e\}) + p'_e \geq 0$, since competitive equilibrium conditions imply that $v(S^*) \geq v(S^* \cup \{e\}) - p'_e$; i.e., in equilibrium (x', p') of G', agent b has no incentive to execute an additional trade $e \notin S^*$.

- $e' \in S^*, e \notin S^*$: Since the constructed solution does not utilize the arc from v^1_e to $v^1_{e'}$, the auxiliary network has an arc from v^1_e to $v^1_{e'}$. The associated reduced cost satisfies $c^{\hat{\pi}}_{(v^1_e, v^1_{e'})} = -v(S^* \cup \{e\} \setminus \{e'\}) + v(S^*) + p'_e - p'_{e'} \geq 0$, since competitive equilibrium conditions imply that $v(S^*) - p'_{e'} \geq v(S^* \cup \{e\} \setminus \{e'\}) - p'_e$; i.e., in equilibrium (X', p') of G', agent b has no incentive to execute an additional trade $e \notin S^*$, while dropping one of the equilibrium trades $e' \in S^*$.

Thus, it follows that the constructed flow-potential function pair $(\hat{x}, \hat{p})$ (together with the associated net outflow vector) satisfies the reduced-cost optimality conditions on all arcs, and yields an optimal solution to the MSFP in $\hat{F}$. Since by convention this flow corresponds to executing trades X', I conclude that X' is an efficient set of trades in $\hat{G}$ as well. This completes the first claim in Step 2.

Consider any other equilibrium price vector $\hat{p}$ in the economy $\hat{G}$ (with agent $\hat{b}$), and observe that $(X', \hat{p})$ is a competitive equilibrium (since any equilibrium price vector supports any efficient set of trades). I show that any such $\hat{p}$ is also a competitive equilibrium price vector supporting X' in the economy G'.

Since $(X', \hat{p})$ is a competitive equilibrium in $\hat{G}$, in flow network $\hat{F}$, the flow $\hat{x}$ corresponding to X' (which is constructed following the same approach as before) satisfies the optimality conditions with the optimal potential function $\hat{\pi}$ associated with $\hat{p}$ as well. Note that this potential function (after normalizing the potential values of the special vertices to $\hat{\pi}(v^{out}) = 0$) should be such that $\hat{\pi}(v^{out}) = 0, \hat{\pi}(v_e^{in}) = \hat{p}_e$ for any trade $e \in \Delta$. Hence, the reduced-cost optimality conditions imply the following:

- For each trade $e' \in S^*$, the path $v^{out} - v^0_{e'} - v^1_{e'} - v^{in}_{e'}$ belongs to the auxiliary network. Since, the reduced costs are nonnegative, we have $0 \leq c^{\hat{\pi}}_{(v^{out}, v^0_{e'})} + c^{\hat{\pi}}_{(v^0_{e'}, v^1_{e'})} + c^{\hat{\pi}}_{(v^1_{e'}, v^{in}_{e'})} = -(-v(S^*) + v(S^* \setminus \{e'\})) - \hat{p}_{e'}$. Hence, $v(S^*) - \hat{p}_{e'} \geq v(S^* \setminus \{e'\})$.

- For each trade $e \notin S^*$, the path $v^{in}_e - v^1_e - v^0_e - v^{out}$ belongs to the auxiliary network. Since, the reduced costs are nonnegative, we have, $0 \leq c^{\hat{\pi}}_{(v^{in}_e, v^1_e)} + c^{\hat{\pi}}_{(v^1_e, v^0_e)} + c^{\hat{\pi}}_{(v^0_e, v^{out})} = v(S^*) - v(S^* \cup \{e\}) + \hat{p}_e$. Hence, $v(S^*) \geq v(S^* \cup \{e\}) - \hat{p}_e$.

- For $e' \in S^*, e \notin S^*$, the path $v^{in}_e - v^1_e - v^1_{e'} - v^{in}_{e'}$ belongs to the auxiliary network. Since, the reduced costs are nonnegative, we have, $0 \leq c^{\hat{\pi}}_{(v^{in}_e, v^1_e)} + c^{\hat{\pi}}_{(v^1_e, v^1_{e'})} + c^{\hat{\pi}}_{(v^1_{e'}, v^{in}_{e'})} = v(S^*) - v(S^* \cup \{e\} \setminus \{e'\}) + \hat{p}_e - \hat{p}_{e'}$. Hence, $v(S^*) \geq v(S^* \cup \{e\} \setminus \{e'\}) - \hat{p}_e + \hat{p}_{e'}$.

These conditions imply that under the price vector $\hat{p}$, agent b (with value function v) cannot deviate from S^* via a single improvement (i.e., dropping a trade, executing a new trade, or both) and improve her payoff. Since v is $M^\natural$-concave, it follows that S^* is demanded by agent b under price vector $\hat{p}$. It readily follows that under this price vector agents other than b also demand their equilibrium trades in X', since their payoffs are identical in economies $\hat{G}$ and G'. Thus, it follows that if $(X', \hat{p})$ is a competitive equilibrium in economy $\hat{G}$, then it is a competitive equilibrium in economy G' as well.

Step 3: I show that there exists an equilibrium price vector $\hat{p}$ in the new economy $\hat{G}$, where the prices of all trades not incident to agent $\hat{b}$ are weakly higher when compared to an equilibrium price vector p in G. I do so by assuming that initially all arcs (v_e^0, v^{out}) in the flow network $\hat{F}$ have zero capacity, i.e., $k_{(v_e^0, v^{out})} = 0$, and by sequentially increasing them to one.

Recall that (X, p) is a competitive equilibrium in the economy G. Consider the associated optimal flow x in F, which is the flow network associated with G. As before, a potential function π that supports x can be derived by setting the potential value of all special vertices to zero, and the potential value of vertices associated with any trade e to p_e.

Next consider $\hat{F}$, but set capacity $k_{(v_e^0, v^{out})} = 0$ for all $e \in \Delta$. Observe that all vertices/arcs in F also belong to $\hat{F}$. Hence, x (after setting flow to zero on all arcs not belonging to F) is still feasible in $\hat{F}$. Set the potential value of vertices of $\hat{F}$ common to F according to π. Next, extend π to define potential values at the remaining vertices of $\hat{F}$.

Let $m = \min_{e' \in S^*, e \notin S^*} v(S^*) - v(S^* \cup \{e\} \setminus \{e'\})$. For $e' \in S^*$, set $\pi(v_{e'}^{in}) = \pi(v_{e'}^1) = -M$ for sufficiently large $M \gg 0$, and for $e \notin S^*$, set $\pi(v_e^{in}) = \pi(v_e^1) = -M - m$. Similarly, for any trade $e \in \Delta$, set the potential value of the vertex associated with the seller of this trade, hereafter s_e, equal to $\pi(s_e) = \pi(v_e^{in})$. Finally set $\pi(v_e^0) = -2M$ for any $e \in \Delta$, and $\pi(v^{out}) = 0$. I next establish that flow x together with the constructed potential function satisfies the reduced-cost optimality conditions on $\hat{F}$ (with $k_{(v_e^0, v^{out})} = 0$ for all $e \in \Delta$), and hence constitutes an optimal flow-potential function pair.

Since (X, p) is a competitive equilibrium in economy G, it can be seen that potential function π satisfies the reduced-cost optimality conditions for the arcs in the network that are not incident to the vertices of agent $\hat{b}$ (or vertices $\{s_{e'} | e' \in \Delta\}$). For sufficiently large M, the fact that $\pi(v_e^0) = -2M$ guarantees that any incoming arc to $\{v_e^0 | e \in \Delta\}$ in the auxiliary network also satisfies the reduced-cost optimality conditions (which are equivalent to the arcs in the flow network since all arcs associated with agent $\hat{b}$ initially have zero flow).

Since $\pi(s_e) = \pi(v_e^{in})$, it also follows that the reduced costs of arcs between these vertices are equal to zero, and hence the optimality conditions are trivially satisfied. The fact that $\pi(s_e) \leq -M \ll 0$ implies that the incoming arcs to s_e satisfy the reduced-cost optimality conditions. Finally, for any $e \in \Delta$, for arc (v_e^{in}, v_e^1) in the auxiliary network we have $c^{\pi}_{(v_e^{in}, v_e^1)} = 0$. Hence, the reduced cost optimality conditions hold for these arcs as well.

Thus, to establish optimality of x, it suffices to verify that any arc $(v_e^1, v_{e'}^1)$ satisfies the reduced-cost optimality conditions for $e \notin S^*, e' \in S^*$. To see this note that $c^{\pi}_{(v_e^1, v_{e'}^1)} = v(S^*) - v(S^* \cup \{e\} \setminus \{e'\}) + \pi(v_e^1) - \pi(v_{e'}^1) = v(S^*) - v(S^* \cup \{e\} \setminus \{e'\}) - m \geq 0$, where the inequality follows from the definition of m. Therefore, it follows that the constructed flow-potential pair satisfies the reduced cost optimality conditions when $k_{(v_e^0, v^{out})} = 0$ for all $e \in \Delta$, and hence is optimal.

I next sequentially increase the capacity of each arc (v_e^0, v^{out}) to $k_{(v_e^0, v^{out})} = 1$, and characterize the change in the optimal flow-potential function pair using Lemma 3.3.1. At each step the optimal flow-potential function pair prior to the update is denoted by (x_{old}, π_{old}) and after the update is denoted by (x_{new}, π_{new}). After finishing the updates for all $e \in \Delta$ an optimal flow-potential function pair $(\hat{x}, \hat{\pi})$ is obtained.

Consider a single step of the above process, and focus on the change in potential values as given by Lemma 3.3.1. The lemma states that the potential value of any vertex v is updated at each step as follows:

$$\pi_{new}(v) = \pi_{old}(v) + d(v),$$

where d are the shortest path distances from v^{out} with respect to the reduced cost of arcs in the auxiliary network corresponding to (x_{old}, π_{old}). Note that $d(v) \geq 0$ since (x_{old}, π_{old}) is optimal prior to capacity increase, and hence satisfies the reduced cost optimality conditions (see Theorem 2.2.2). Also observe that the special vertices still have a potential value of zero, since all special vertices are connected by uncapacitated zero reduced-cost arcs to v^{out}.

Similarly, two vertices of the flow network that correspond to the same trade in the trading network are connected by a zero length arc, since in the flow network the corresponding arc is uncapacitated. Thus, equality between the potential values of the end points of an arc representing a trade that is not incident to $\hat{b}$ is preserved, and can still be interpreted as the price of the trade. I conclude that the potential values of all vertices in the network (weakly) increase, and those of the special vertices remain at zero. In particular, the potential value of a vertex in the flow network associated with a trade e that is not incident to $\hat{b}$ increases, thereby implying that the price for trade e increases.

By updating the capacities sequentially, an optimal flow in the flow network $\hat{F}$ is constructed, where all vertices associated with trades not incident to $\hat{b}$ have (weakly) higher potential values than before. I conclude that there exists an equilibrium in $\hat{G}$, where any trade e not incident to $\hat{b}$ has higher prices when compared to p, i.e., $\hat{p}_e \geq p_e$.

By Step 2, we know that $\hat{p}$ is also a competitive equilibrium price vector in G'. Hence, it follows that $(X', \hat{p})$ constitutes a competitive equilibrium in G'. Since $\hat{p}_e \geq p_e$ for any trade e that is not incident to $\hat{b}$, the claim follows. □

The above theorem makes use of the sensitivity analysis result on MSFP given in Lemma 3.3.1 to characterize how prices change when a new buyer joins the economy. I next examine the impact of such price changes on agents' payoffs.

Corollary 3.3.5. Let (X, p) be a competitive equilibrium in G. Assume that under X, agent a_1 only participates in trades where she is a seller, and agent a_2 only participates in trades where she is a buyer. Let $\sigma_{a_1}, \sigma_{a_2}$ be the payoffs of these agents at (X, p).

Suppose that a new buyer is introduced. Then, there exists a competitive equilibrium (X', p') of the induced trading network G', where the corresponding payoffs $\sigma'_{a_1}, \sigma'_{a_2}$ satisfy:

- $\sigma_{a_1} \leq \sigma'_{a_1}$, and

- $\sigma_{a_2} \geq \sigma'_{a_2}$, unless a_2 starts participating in new trades as a seller under X'.

Proof. Theorem 3.3.4 implies that there exists a competitive equilibrium (X', p') such that all prices (weakly) increase.

Note that if agent a_1 still participates in her optimal trades in X (which are not necessarily optimal under the new prices), her payoff increases (since she participates in those trades as a seller, and the prices have increased). Since under her new equilibrium trades, her payoff is weakly higher, it follows that her equilibrium payoff also increases.

Suppose that under X' agent a_2 only participates in trades S_2, where she is a buyer. Observe that her payoff for S_2 is higher prior to introducing the new buyer, since the prices of these trades were weakly lower. Since her equilibrium payoff (prior to the update) was weakly higher than her payoff for the bundle of trades S_2, it follows that the payoff of a_2 decreases, if she continues to participate in trades only as a buyer. $\square$

Similar results follow when a new seller is introduced into the economy. Mimicking the proof of Theorem 3.3.4, it can be shown that there exists a new competitive equilibrium where all prices decrease and the buyers in the initial economy enjoy higher payoffs.

Qualitatively, these results suggest that when more "downstream" agents (buyers) are added to the economy, the payoffs of "upstream" agents (sellers) improve and vice versa. It is natural to expect the same when the set of agents is fixed but more trades connecting upstream and downstream agents are added to the economy. The next example shows this to be false, even when the new trades correspond to paths that bypass a "middleman" who adds no value. This example highlights the nontrivial behavior of equilibria in trading networks, and the special effect of adding a buyer/seller to the economy. It also implies that while the addition of a new seller improves the payoffs of the buyers in the economy, the change in the payoff need not be monotone in the arcs added to the economy.

Example. In the trading network displayed in Figure 7, agent i has a single good to offer, and incurs a cost of 0 for providing it. Agent l has a value of 10 for consuming the good. Agent j is an intermediary, who has a value of 0 for the good but can facilitate the transfer of the good between i and l at zero cost. Agent k has a value of 1 for consuming the good, and she can also transfer the good between i and l at zero cost. Initially the solid arcs belong to the trading network, and I examine the change in the equilibrium outcome when the dashed arc (i, k) is added to the trading network. An equilibrium in the initial

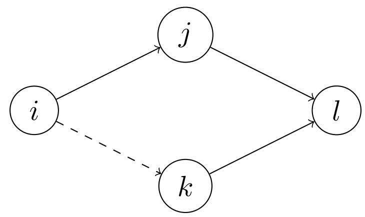

Figure 7: Payoff of buyer l may decrease as more paths to seller i emerge.

network involves trading the good along the $(i, j) - (j, l)$ path at a price of zero. This yields a payoff of 10 units for agent l and zero for the remaining agents. On the other hand, after trade (i, k) is introduced, agent k is willing to pay at least 1 unit to agent i to acquire the good. Consequently, agent i can guarantee at least 1 unit of payoff. Since the total payoff in the efficient set of trades is still 10 units, it follows that the payoff of agent l reduces in *any* equilibrium in the new network. Thus the addition of a trade might raise the prices downstream and reduce the payoff of the final buyer. It is straightforward to construct other examples where the addition of the new trade increases the payoff of the buyer and/or decreases the payoff of the seller.

3.3.3. Addition of Trades between Multiple Buyers and Sellers

The results thus far are restricted to settings where all new trades in the economy are incident to a single agent. Next, they are used as building blocks to characterize how equilibria change with new trades involving multiple buyers/sellers.

The key idea is that a set of agents in a trading network can be replaced with a single representative agent without impacting the equilibrium prices of the trades in the economy for trades non-incident to the aforementioned set of agents. This result is formalized in Lemma 3.3.7. Using it together with the findings of the previous subsections, I characterize the equilibrium impact of adding a set of trades that connect two groups of agents in the underlying trading network (Theorem 3.3.8).

I start by formalizing the notion of a *representative agent*. Suppose that a trading network $G = (N, E)$ is given, and consider a set of agents $\hat{N} \subset N$. Given an arc $e \in E$ as before, let e^+ and e^- denote the tail and head of this arc, respectively. Denote by $E_{\hat{N}} = \{e \in E | e^+, e^- \in \hat{N}\}$ the arcs that are among the members of $\hat{N}$. Let $\delta(\hat{N}) = \cup_{i \in \hat{N}} \delta(i) \setminus E_{\hat{N}}$ denote the set of arcs that have an end point in $\hat{N}$ and another in $N \setminus \hat{N}$. Similarly, let $\delta_+(\hat{N})$ ($\delta_-(\hat{N})$) denote the set of arcs in $\delta(\hat{N})$ that are outgoing from (incoming to) $\hat{N}$.

I construct another trading network $G' = (N', E')$ by replacing $\hat{N}$ with the representative agent $\hat{n}$ such that $N' = (N \setminus \hat{N}) \cup \{\hat{n}\}$. For any $i \in N'$, denote the incident arcs in G' by $\delta'(i)$ and the outgoing and incoming arcs by $\delta'_+(i)$ and $\delta'_-(i)$, respectively. Note that $E' = \cup_{i \in N'} \delta'(i)$. The arcs in G' that are not incident to $\hat{n}$ are precisely the arcs in G that are not incident to $\hat{N}$. On the other hand, each arc that is incident to $\hat{n}$ in G' corresponds to an arc in $\delta(\hat{N})$ in G. Formally, there is a bijective map $\beta : \delta'(\hat{n}) \to \delta(\hat{N})$ such that

1. $\beta(\delta'_+(\hat{n})) = \delta_+(\hat{N})$, $\beta(\delta'_-(\hat{n})) = \delta_-(\hat{N})$,

2. for $\hat{e} \in \delta'_+(\hat{n})$ and $e = \beta(\hat{e})$ (similarly $\hat{f} \in \delta'_-(\hat{n})$ and $f = \beta(\hat{f})$) we have $\hat{e}^- = e^-$ (similarly $\hat{f}^+ = f^+$).

Note that under this construction the arcs incident to each vertex can be given as follows: $\delta'(i) = (\delta(i) \setminus \delta(\hat{N})) \cup \beta^{-1}(\delta(\hat{N}) \cap \delta(i))$ for $i \in N' \setminus \{\hat{n}\}$, and $\delta'(\hat{n}) = \beta^{-1}(\delta(\hat{N}))$.

Define the value function $\hat{w}_i$ of agent $i \in N' \setminus \{\hat{n}\}$ in G' such that for any $S \subset \delta'(i)$, we have:

$$\hat{w}_i(S) = w_i(S \setminus \delta'(\hat{n}) \cup \beta(S \cap \delta'(\hat{n}))).$$

That is, the valuation of i for a set of incident trades S in G' is equivalent to the valuation she has for the corresponding trades in G. Define the value function of $\hat{n}$ such that

$$\hat{w}_{\hat{n}}(S) = \max_{X \subset E_{\hat{N}}} \sum_{i \in \hat{N}} w_i((X \cup \beta(S)) \cap \delta(i)), \tag{3.5}$$

for any $S \subset \delta'(\hat{n})$. Intuitively, each trade between $\hat{N}$ and $N \setminus \hat{N}$ corresponds to a trade between the representative agent $\hat{n}$ and $N \setminus \hat{N}$, and the maximum aggregate value of agents in $\hat{N}$ when a subset of these trades is executed (and the trades among members of $\hat{N}$ are chosen in the best possible way), is equal to the payoff of the representative agent $\hat{n}$ for the same set of trades.

Since for $i \in N' \setminus \{\hat{n}\}$, the valuation for a set of trades is defined as the valuation the agent has for the corresponding trades in the original economy, it readily follows that $\hat{w}_i$ is $M^\natural$-concave. Despite being less obvious, the same conclusion also holds for agent $\hat{n}$, as the next lemma[7] establishes. For this lemma, as well as for subsequent results, I follow the same convention as before and express $\hat{w}_i$ (w_i) as a function on $\mathbb{Z}^{\delta'(i)}$ ($\mathbb{Z}^{\delta(i)}$).

Lemma 3.3.6. $\hat{w}_{\hat{n}} : \mathbb{Z}^{\delta'(\hat{n})} \to \mathbb{R}$ is $M^\natural$-concave.

Proof. Let $h : \times_{i \in \hat{N}} \mathbb{Z}^{\delta(i)} \to \mathbb{R}$ be such that:

$$h(y) = \sum_{i \in \hat{N}} w_i(y^i),$$

where $y = \{y^i\}_{i \in \hat{N}}$ and $y^i \in \mathbb{Z}^{\delta(i)}$. Since h is a sum of $M^\natural$-concave functions with disjoint arguments, it follows that it is $M^\natural$-concave. I define another function $g : \mathbb{Z}^{\delta'(\hat{n})} \times \mathbb{Z}^{E_{\hat{N}}} \to \mathbb{R}$ such that:

$$g(y_{\delta'(\hat{n})}, z_{E_{\hat{N}}}) = \max_{y_{E_{\hat{N}}}} \{h(y_{\delta'(\hat{n})}, y_{E_{\hat{N}}}) \mid y_e^{e^+} + y_e^{e^-} = z_e \text{ for } e \in E_{\hat{N}}\}. \tag{3.6}$$

Here, I split the argument y of h into $y_{\delta'(\hat{n})}$ and $y_{E_{\hat{N}}}$ such that $y_{\delta'(\hat{n})} \in \mathbb{Z}^{\delta'(\hat{n})} = \mathbb{Z}^{\delta(\hat{N})}$ consists of the entries of y that correspond to the arcs between the vertices in $\hat{N}$ and the vertices in $N \setminus \hat{N}$, while $y_{E_{\hat{N}}} = \{y_e^{e+}, y_e^{e-}\}_{e \in E_{\hat{N}}} \in \mathbb{Z}^{E_{\hat{N}}} \times \mathbb{Z}^{E_{\hat{N}}}$ consists of the remaining entries of y, and $z_{E_{\hat{N}}}$ represents a vector in $\mathbb{Z}^{E_{\hat{N}}}$. Observe that if $\{y^i\}_{i \in \hat{N}}$ is consistent

[7] The operation used to define the payoff of the representative agent in (3.5) is mathematically similar to (but different from) the operation used in the definition of contraction of an economy introduced in (3.3). In the proof, it is established that the former operation also preserves $M^\natural$-concavity.

with a set of trades $X \subset E_{\hat{N}}$ (i.e., for each $i \in \hat{N}$, y^i encodes the trades in X in which i participates as a buyer by -1 and those in which i participates as a seller by 1), then we have $y_e^{e^+} + y_e^{e^-} = 0$ for all $e \in E_{\hat{N}}$. Note that $\hat{w}_{\hat{n}}$ is defined in terms of a set of trades $X \subset E_{\hat{N}}$ that solves the optimization problem given in (3.5). Leveraging this observation, I conclude that agent $\hat{n}$'s value function satisfies

$$\hat{w}_{\hat{n}}(y_{\delta'(\hat{n})}) = g(y_{\delta'(\hat{n})}, 0_{E_{\hat{N}}}), \tag{3.7}$$

for $y_{\delta'(\hat{n})} \in \mathbb{Z}^{\delta'(\hat{n})}$, where $0_{E_{\hat{N}}}$ is a vector whose entries are all equal to zero.

The maximization in (3.6) used to define g is referred to as the aggregation operation. In (3.7), $\hat{w}_{\hat{n}}$ is expressed in terms of g by setting some of the entries equal to zero – an operation referred to as restriction. Both aggregation and restriction preserve $M^\natural$-concavity (Theorem 6.15 in Murota (2003)). These observations imply that $\hat{w}_{\hat{n}}$ is $M^\natural$-concave, and the claim follows. $\square$

Let (X, p) denote a competitive equilibrium in $G = (N, E)$. I refer to the tuple (X', p') as the projection of this equilibrium onto G' if $X' = X \setminus (E_{\hat{N}} \cup \delta(\hat{N})) \cup \beta^{-1}(X \cap \delta(\hat{N})) \subset E'$ and $p' \in \mathbb{R}^{|E'|}$ is such that $p'_e = p_e$ for $e \in E' \setminus \delta'(\hat{n})$ and $p'_f = p_{\beta(f)}$ for $f \in \delta'(\hat{n})$. Intuitively, (X', p') is a projection of (X, p), if (i) X' consists of the trades in G' that correspond to the trades $X \setminus E_{\hat{N}}$ in G and (ii) prices p' for trades in G' match the prices of the corresponding trades in G. The next lemma establishes that (X, p) and (X', p') are closely related.

Lemma 3.3.7. Consider trading network $G = (N, E)$, and the network $G' = (N', E')$ obtained after replacing $\hat{N} \subset N$ by the representative agent $\hat{n}$.

(i) If (X, p) is a competitive equilibrium in G, then its projection (X', p') is a competitive equilibrium in G'.

(ii) Conversely, if (X', p') is a competitive equilibrium in G', then there exists a competitive equilibrium (X, p) in G, whose projection yields (X', p').

Proof.

Proof of (i). I first prove that the projection (X', p') is an equilibrium in G'. To establish this, it suffices to show that for any $i \in N'$ the payoff-maximizing trades are given by $X' \cap \delta'(i)$ under price vector p'. Recall that by the definition of the projection, for any $i \in N \setminus \hat{N} \subset N'$, the adjacent trades in X' correspond to those in X. Furthermore, the valuation of agent $i \in N \setminus \hat{N}$ for any subset of adjacent trades in G is the same as her valuation for the corresponding trades in G', and the prices of these trades are identical under p and p'. Finally, since (X, p) is a competitive equilibrium, in G under price vector p the set of payoff-maximizing trades for agent $i \in N \setminus \hat{N}$ is given by $X \cap \delta(i)$. These observations imply that in G', for $i \in N \setminus \hat{N}$ the set of payoff-maximizing trades under price vector p' is given by $X' \cap \delta'(i)$.

It remains to prove that in network G' under price vector p' the payoff-maximizing trades for $\hat{n}$ are given by $X' \cap \delta'(\hat{n})$ I claim that the following inequalities hold:

$$
\begin{aligned}
\hat{w}_{\hat{n}}(X' \cap \delta'(\hat{n})) + \sum_{e \in X' \cap \delta'_+(\hat{n})} p'_e - \sum_{e \in X' \cap \delta'_-(\hat{n})} p'_e &\geq \sum_{i \in \hat{N}} [w_i(X \cap \delta(i))] + \sum_{e \in X \cap \delta_+(\hat{N})} p_e - \sum_{e \in X \cap \delta_-(\hat{N})} p_e \\
&= \sum_{i \in \hat{N}} [w_i(X \cap \delta(i)) + \sum_{e \in X \cap \delta_+(i)} p_e - \sum_{e \in X \cap \delta_-(i)} p_e] \\
&\geq \sum_{i \in \hat{N}} \max_{Y^i \subset \delta(i)} [w_i(Y^i) + \sum_{e \in Y^i \cap \delta_+(i)} p_e - \sum_{e \in Y^i \cap \delta_-(i)} p_e] \\
&\geq \max_{Y \subset E} \sum_{i \in \hat{N}} [w_i(Y \cap \delta(i)) + \sum_{e \in Y \cap \delta_+(i)} p_e - \sum_{e \in Y \cap \delta_-(i)} p_e] \\
&\geq \max_{Y \subset E} \sum_{i \in \hat{N}} [w_i(Y \cap \delta(i))] + \sum_{e \in Y \cap \delta_+(\hat{N})} p_e - \sum_{e \in Y \cap \delta_-(\hat{N})} p_e \\
&= \max_{S \subset \delta'(\hat{n})} [\hat{w}_{\hat{n}}(S) + \sum_{e \in S \cap \delta'_+(\hat{n})} p'_e - \sum_{e \in S \cap \delta'_-(\hat{n})} p'_e].
\end{aligned}
$$

Here, the first inequality follows since $X \cap E_{\hat{N}}$ is a feasible solution to the maximization problem (3.5), and by the definition of projection we have $\beta(X' \cap \delta'(\hat{n})) = X \cap \delta(\hat{N})$, as well as $p'_e = p_e$ for $e \in E' \setminus \delta'(\hat{n})$ and $p'_f = p_{\beta(f)}$ for $f \in \delta'(\hat{n})$. The second line follows by noting that for any $e \in X \cap E_{\hat{N}}$ there exists one agent in $\hat{N}$ for whom the corresponding price p_e appears with a positive sign in the summations in the first line and another agent in $\hat{N}$ for whom the same term appears with a negative sign. Thus, they cancel out. The third line follows since (X, p) is a competitive equilibrium, and hence $X \cap \delta(i)$ is a payoff-maximizing set of trades for agent i in G under price vector p. The fourth line follows by switching the order of max operators and the summations. The fifth line follows once again by canceling out common price terms. By the definition of the projection and (3.5), it can be seen that the expression in this line is equivalent to the payoff-maximization problem of agent $\hat{n}$ in G' under price vector p'. Therefore, the last line follows. On the other hand, these expressions imply that for agent $\hat{n}$ the payoff-maximizing trades are indeed given by $X' \cap \delta'(\hat{n})$. Hence, I conclude that (X', p') is a competitive equilibrium, as claimed.

Proof of (ii). I next prove the second part of the claim. Suppose that (X', p') is a competitive equilibrium in G'. I construct an equilibrium (X, p) in G such that (X', p') is a projection of this equilibrium. To that end, set $p_e = p'_e$ for $e \in E \setminus (\delta(\hat{N}) \cup E_{\hat{N}})$ and $p_f = p'_{\beta^{-1}(f)}$ for $f \in \delta(\hat{N})$. Similarly, set $X \setminus E_{\hat{N}} = (X' \setminus \delta'(\hat{n})) \cup \beta(X' \cap \delta'(\hat{n}))$. I will construct the prices $\{p_e\}_{e \in E_{\hat{N}}}$ and the equilibrium trades in $E_{\hat{N}}$ (i.e., $X \cap E_{\hat{N}}$) in the rest of the proof.

To construct the remaining equilibrium quantities, I first consider another trading network $\tilde{G} = (\hat{N}, E_{\hat{N}})$, which is obtained by restricting attention to the subnetwork of G involving agents in $\hat{N}$. In $\tilde{G}$, let the value function $\tilde{w}_i : 2^{E_{\hat{N}} \cap \delta(i)} \to \mathbb{R}$ of $i \in \hat{N}$ be such that

$$\tilde{w}_i(S) = \max_{T \in \delta(\hat{N})} [w_i((T \cup S) \cap \delta(i)) + \sum_{e \in T \cap \delta_+(i)} p_e - \sum_{e \in T \cap \delta_-(i)} p_e], \tag{3.8}$$

for $S \subset E_{\hat{N}} \cap \delta(i)$. Note that we have

$$\begin{aligned}
\max_{\tilde{X} \subset E_{\hat{N}}} \sum_{i \in \hat{N}} \tilde{w}_i(\tilde{X} \cap \delta(i)) &= \max_{\tilde{X} \subset E_{\hat{N}}} \sum_{i \in \hat{N}} \max_{T \in \delta(\hat{N})} [w_i((T \cup \tilde{X}) \cap \delta(i)) + \sum_{e \in T \cap \delta_+(i)} p_e - \sum_{e \in T \cap \delta_-(i)} p_e] \\
&= \max_{T \in \delta(\hat{N})} \max_{\tilde{X} \subset E_{\hat{N}}} \sum_{i \in \hat{N}} [w_i((T \cup \tilde{X}) \cap \delta(i)) + \sum_{e \in T \cap \delta_+(i)} p_e - \sum_{e \in T \cap \delta_-(i)} p_e] \\
&= \max_{T \in \delta'(\hat{n})} \hat{w}_{\hat{n}}(T) + \sum_{e \in T \cap \delta'_+(\hat{n})} p'_e - \sum_{e \in T \cap \delta'_-(\hat{n})} p'_e \\
&= \hat{w}_{\hat{n}}(X' \cap \delta'(\hat{n})) + \sum_{e \in X' \cap \delta'_+(\hat{n})} p'_e - \sum_{e \in X' \cap \delta'_-(\hat{n})} p'_e.
\end{aligned}$$

Here, the first equality is by the definition of $\{\tilde{w}_i\}_{i \in \hat{N}}$. Note that the max operation can be pushed out of the summation, since the arguments related to each agent are disjoint; i.e., the collection $\{\delta(\hat{N}) \cap \delta(i)\}_{i \in \hat{N}}$ consists of disjoint sets. Hence, the second equality follows. The third one follows from (3.5) and the definition of prices $\{p_e\}_{e \in \delta(\hat{N})}$. The last equality follows since (X', p') constitutes an equilibrium in G'. Observe that an efficient set of trades $\tilde{X}^\star$ in $\tilde{G}$ solves the optimization problems in the first line. Let $(\tilde{X}^\star, \tilde{p})$ be an equilibrium in $\tilde{G}$. Note that the last equality also implies that $T = \beta(X' \cap \delta'(\hat{n}))$ solves the optimization problems on the right-hand side of the first line where T appears. Equivalently, it implies that for any $i \in \hat{N}$ and $S = \tilde{X}^\star \cap \delta(i)$ an optimal solution of the optimization problem in (3.8) is given by $T = \beta(X' \cap \delta'(\hat{n}))$.

Let $X \cap E_{\hat{N}} = \tilde{X}^\star$. Together with the construction of $X \setminus E_{\hat{N}}$, this implies that $X = \tilde{X}^\star \cup (X' \setminus \delta'(\hat{n})) \cup \beta(X' \cap \delta'(\hat{n}))$. Setting $p_e = \tilde{p}_e$ for $e \in E_{\hat{N}}$ I also finalize the construction of the price vector p. I claim that (X, p) is an equilibrium in G. To establish this, I will show that in G, the trades $X \cap \delta(i)$ are payoff-maximizing for all $i \in N$ under price vector p.

To see this first consider $i \in N \setminus \hat{N}$. Note that, by construction, trades $X \cap \delta(i)$ in G correspond to trades $X' \cap \delta'(i)$ in G'. Furthermore, as before, the valuation of agent $i \in N \setminus \hat{N}$ for any subset of adjacent trades in G is the same as her valuation for the corresponding trades in G', and the prices of these trades are identical under p and p'. Since (X', p') is an equilibrium in G', it follows that $X' \cap \delta'(i)$ is payoff-maximizing in G' under price vector p'. On the other hand, since these trades correspond to $X \cap \delta(i)$ in G, and the prices of trades adjacent to i are identical in both networks, it follows that $X \cap \delta(i)$ maximizes the payoff of $i \in N \setminus \hat{N}$ under price vector p.

Consider next $i \in \hat{N}$. Since $(\tilde{X}^\star, \tilde{p})$ is an equilibrium in $\tilde{G}$, it follows that trades $\tilde{X}^\star \cap \delta(i)$ are payoff-maximizing for i in $\tilde{G}$ under price vector $\tilde{p}$, i.e.,

$$\tilde{w}_i(\tilde{X}^\star \cap \delta(i)) + \sum_{e \in \tilde{X}^\star \cap \delta_+(i)} \tilde{p}_e - \sum_{e \in \tilde{X}^\star \cap \delta_-(i)} \tilde{p}_e \geq \max_{S \subset E_{\hat{N}} \cap \delta(i)} \tilde{w}_i(S) + \sum_{e \in S} \tilde{p}_e - \sum_{e \in S} \tilde{p}_e. \tag{3.9}$$

Recall that for $S = \tilde{X}^\star \cap \delta(i)$, an optimal solution of the optimization problem in (3.8) is given by $T = \beta(X' \cap \delta'(\hat{n}))$. Using this observation and the definition of $\{p_e\}_{e \in E_{\hat{N}}}$, the left-hand side of (3.9) is given by:

$$w_i(Y \cap \delta(i)) + \sum_{e \in Y \cap \delta_+(i)} p_e - \sum_{e \in Y \cap \delta_-(i)} p_e = w_i(X \cap \delta(i)) + \sum_{e \in X \cap \delta_+(i)} p_e - \sum_{e \in X \cap \delta_-(i)} p_e, \tag{3.10}$$

where $Y = (\beta(X' \cap \delta'(\hat{n})) \cup \tilde{X}^\star)$. Here, the equality follows since the definition of X implies that for $i \in N$ we have $Y \cap \delta(i) = X \cap \delta(i)$. Similarly, using (3.8), the right-hand side of (3.9) is given by

$$\max_{S \subset E_{\hat{N}} \cap \delta(i)} \max_{T \in \delta(\hat{N})} [w_i((T \cup S) \cap \delta(i)) + \sum_{e \in (T \cup S) \cap \delta_+(i)} p_e - \sum_{e \in (T \cup S) \cap \delta_-(i)} p_e]. \tag{3.11}$$

It follows from (3.9)–(3.11) that $X \cap \delta(i)$ is payoff maximizing for agent i in network G under price vector p. Finally, it readily follows from the construction of (X, p) that its projection yields (X', p'). Hence, the claim follows. □

I proceed with establishing the first comparative static of this section. In particular, suppose that the initial trading network has two components G^s and G^b that possibly consist of multiple traders and trades among them (see Figure 8). Suppose that new trades from G^s

to G^b that are incident to multiple traders from G^s and G^b are added to the economy. What is the impact of these trades on the competitive equilibrium?

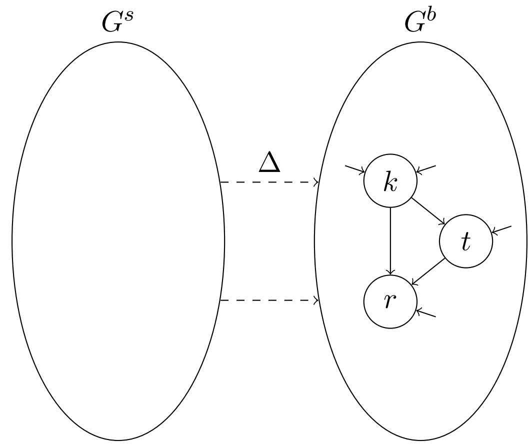

Figure 8: Comparative statics involving multiple buyers/sellers. Initial economy has components G^s/G^b that are connected by the new trades (dashed arcs).

Theorem 3.3.8. Suppose that trading network G consists of two components $G^s = (N^s, E^s)$ and $G^b = (N^b, E^b)$, and let (X, p) denote an equilibrium in this network. Suppose that a set Δ of trades whose sellers belong to N^s and buyers belong to N^b are added to the economy. Then,

- there exists an equilibrium (X', p') in the new economy such that $p'_e \geq p_e$ for $e \in E^s$.
- there exists an equilibrium (X'', p'') in the new economy such that $p''_e \leq p_e$ for $e \in E^b$.

Proof. Theorem 3.3.8 is proven by leveraging Lemma 3.3.7 together with the earlier comparative statics results. In particular, replace agents in G^b (similarly G^s) with a single representative buyer (seller). Then, analyzing the impact of the addition of new arcs reduces to analyzing the addition of a (representative) buyer (similarly seller) to the economy. The impact of the addition of a new buyer to the economy was characterized in Section 3.3.2. The theorem follows by leveraging this characterization.

Denote the trading network obtained after adding trades Δ to G by G'. Denote by $\hat{G}$ the trading network obtained from G after replacing G^b with a representative agent b. Observe that adding trades Δ to G is equivalent to adding the same trades between b and the vertices that belong to G^s in $\hat{G}$. Denote the trading network induced by adding the aforementioned trades to $\hat{G}$ by $\hat{G}'$. In networks $\hat{G}$ and $\hat{G}'$ the value function of the representative agent b is defined as in (3.5), in terms of the value functions of the agents in G^b (which capture the valuations of these agents for trades in Δ). Note that $\hat{G}'$ can equivalently be obtained from G' by replacing G^b with the representative agent b.

Let (X, p) denote an equilibrium in G. By Lemma 3.3.7 there exists an equilibrium $(\hat{X}, \hat{p})$ in $\hat{G}$ that is a projection of (X, p). Theorem 3.3.4 implies that $\hat{G}'$ has an equilibrium $(\hat{X}', \hat{p}')$, where the prices of trades in E^s weakly increase relative to the corresponding prices in $\hat{p}$. By Lemma 3.3.7 there is an equilibrium (X', p') in G', where the prices of trades outside G^b are the same as those in $\hat{p}'$. These observations imply that $p'_e = \hat{p}'_e \geq \hat{p}_e = p_e$ for all $e \in E^s$. Thus, the first part of the theorem follows.

The proof of the second part is similar and is omitted. The key difference is that the proof now involves replacing G^s with a representative agent s (as opposed to replacing G^b with b), and using an analogous result to Theorem 3.3.4 that shows that the addition of a new seller (as opposed to a buyer) decreases (as opposed to increases) the prices in the rest of the economy. □

Note that Corollary 3.3.5 generalizes to the setting of Figure 8. Suppose that there exists an equilibrium of the original economy where some agent in G^b participates only as a buyer (e.g., agent r in Figure 8). Theorem 3.3.8 implies that there exists an equilibrium in the new economy where the payoff of this agent weakly increases. Similar characterizations hold for agents who participate only as sellers in G^b, as well as for such agents in G^s. Perhaps more interestingly, it is possible to reason about the impact of new trading opportunities on the payoffs of a set of agents. For instance, if there is a set of agents $\hat{N}^b$ in $G^b = (N^b, E^b)$ who in the initial equilibrium participate as buyers in trades between $\hat{N}^b$ and $N^b \setminus \hat{N}^b$, the aggregate equilibrium payoff of these agents increases from the addition of the new trades. For instance, the total payoff of agents k, r, t increases as a result of the addition of the dashed arcs in Figure 8. Since these observations on payoff changes follow from Theorem 3.3.8 in a straightforward way by mimicking the approach of Corollary 3.3.5 I do not state them formally.

The characterization in this subsection so far focuses on settings where the initial network has two disconnected components. I next generalize this characterization, by allowing for arcs between G^b and G^s.

Theorem 3.3.9. Let $G^s = (N^s, E^s)$ and $G^b = (N^b, E^b)$ denote two induced subnetworks of the trading network $G = (N, E)$. Suppose that $E = E^b \cup E^s \cup L \cup R$, where trades in L (R) have sellers in N^s (N^b) and buyers in N^b (N^s).

Let (X, p) denote an equilibrium in G where all trades in R are executed and no trade in L is executed. Assume that a set Δ of trades whose sellers belong to N^s and buyers belong to N^b are added to the economy. Then, the results of Theorem 3.3.8 continue to hold.

Proof. To establish the result, I consider a sequence of transformations to the underlying network, as illustrated in Figure 9, and study how the competitive equilibria change as a result of these transformations. All these networks share the same set of vertices, N. $\bar{G}$ is obtained from G by replacing each arc (trade) in R with an arc between the same vertices but with opposite orientation. The set of arcs that replace those in R is denoted by $\bar{R}$. $\hat{G}$ is obtained from $\bar{G}$ by removing the arcs in $L \cup \bar{R}$. $\tilde{G}$ is obtained from $\hat{G}$ by adding $L \cup \bar{R} \cup \Delta$. Finally, G' is obtained from $\tilde{G}$ by replacing the arcs in $\bar{R}$ with those in R. Note that G' can also be directly obtained from G by adding arcs Δ to G. However, considering the sequence of transformations in Figure 9 allows us to reason about how competitive equilibria change in a more straightforward way. In particular, I will show that after each transformation it is possible to obtain a competitive equilibrium where the prices of trades in E^s weakly increase. Note that this readily implies that the first result of Theorem 3.3.8 continues to hold in the setting described in Theorem 3.3.9, as claimed.

I first describe the transformation $G \to \bar{G}$ where all arcs in R are replaced with arcs in $\bar{R}$

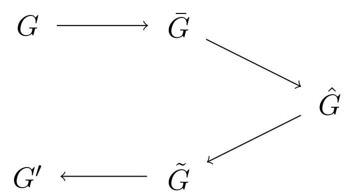

Figure 9: A sequence of transformations. The networks on the left (G, G') involve R, whereas the ones in the middle $(\bar{G}, \tilde{G})$ involve $\bar{R}$. The lower networks $(\tilde{G}, G')$ involve Δ and the ones above $(G, \bar{G})$ do not. $\hat{G}$ does not involve any of the arcs in Δ, L, R, or $\bar{R}$.

that have the opposite orientation (see Figure 10). Let $\bar{E}$ denote the set of arcs in $\bar{G}$, and

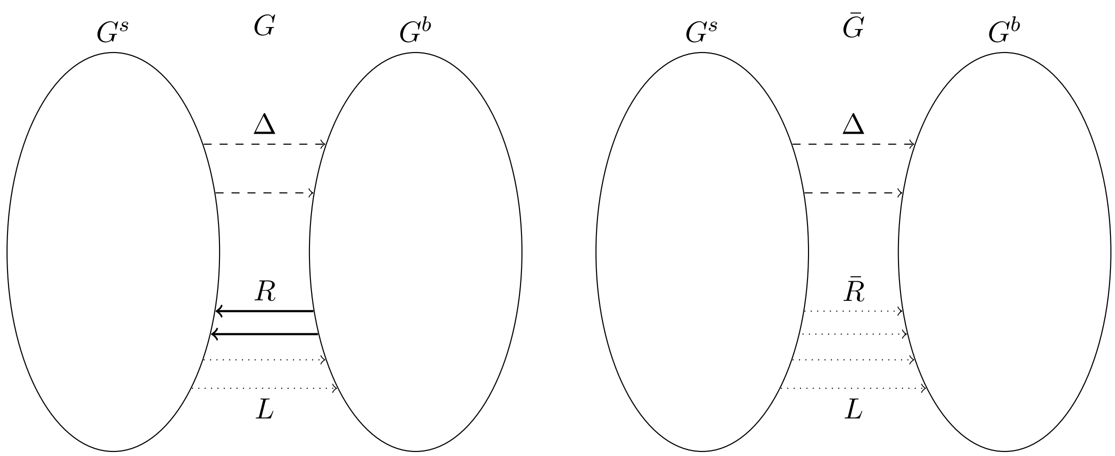

Figure 10: Δ denotes the set of new arcs (trades) in the trading network. In the initial equilibrium in G, the trades associated with the arcs in R are executed. $\bar{R}$ is obtained from R by reversing the orientation of each arc in R.

$\bar{\delta}(i) \subset \bar{E}$ denote the set of (directed) arcs incident to i in this network. Define a bijective map $\gamma : E \to \bar{E}$ such that:

1. $\gamma(e)^+ = e^+$ and $\gamma(e)^- = e^-$ for all $e \in E \setminus R$,

2. $\gamma(e)^+ = e^-$ and $\gamma(e)^- = e^+$ for all $e \in R$.

Similarly, define a bijective map $\zeta : 2^E \to 2^{\bar{E}}$, such that for any $Y \subset E$ we have

1. $e \in Y \Leftrightarrow \gamma(e) \in \zeta(Y)$ for all $e \in E \setminus R$,

2. $e \in Y \Leftrightarrow \gamma(e) \notin \zeta(Y)$ for all $e \in R$.

Next, the equilibrium (X, p) in G is used to construct the valuations of the agents in $\bar{G}$. In particular, let

$$\bar{w}_i(\bar{Y}) = w_i(\zeta^{-1}(\bar{Y}) \cap \delta(i)) - \sum_{e \in R \cap \delta_-(i)} p_e + \sum_{e \in R \cap \delta_+(i)} p_e,$$

for all $\bar{Y} \in \bar{\delta}(i)$ and $i \in N$. Consider the representation of $\bar{w}_i$ as a function on $\mathbb{Z}^{\bar{\delta}(i)} = \mathbb{Z}^{\delta(i)}$. It readily follows from the definition of $\bar{w}_i$ and ζ that:

$$\bar{w}_i(z) = w_i(z_{E \setminus R}, z_{\bar{R}_+} - 1_{\bar{R}_+}, z_{\bar{R}_-} + 1_{\bar{R}_-}) - \sum_{e \in R \cap \delta_-(i)} p_e + \sum_{e \in R \cap \delta_+(i)} p_e,$$

for any $z \in \mathbb{Z}^{\bar{\delta}(i)}$. Here, $z_{E \setminus R}$ denotes the entries of z that correspond to the arcs in $E \setminus R = \bar{E} \setminus \bar{R}$. I use the shorthand notation $\bar{R}_+$ (similarly $\bar{R}_-$) to denote $\bar{R} \cap \bar{\delta}_+(i)$ ($\bar{R} \cap \bar{\delta}_-(i)$), and define $z_{\bar{R}_+}$ and $z_{\bar{R}_-}$ similarly to $z_{E \setminus R}$. 1_S is used to denote the vector of ones in $\mathbb{Z}^S$ for $S \in \{\bar{R}_+, \bar{R}_-\}$. Also, when I write $w_i(z_{E \setminus R}, z_{\bar{R}_+} - 1_{\bar{R}_+}, z_{\bar{R}_-} + 1_{\bar{R}_-})$, I follow the convention that $z_{\bar{R}_+} - 1_{\bar{R}_+}$ ($z_{\bar{R}_-} + 1_{\bar{R}_-}$) is associated with the part of the domain of w_i that corresponds to $\delta_-(i)$ ($\delta_+(i)$). Note that this convention is consistent with the construction, since the arcs $\bar{\delta}_+(i)$ ($\bar{\delta}_-$(i)) in $\bar{G}$ correspond to the arcs $\delta_-(i)$ ($\delta_+(i)$) of G. Since $-\sum_{e \in R \cap \delta_-(i)} p_e + \sum_{e \in R \cap \delta_+(i)} p_e$ is a constant and w_i is $M^\natural$-concave, it readily follows from Theorem 6.15 in Murota (2003) that $\bar{w}_i$ is $M^\natural$-concave.

Next, I construct a competitive equilibrium $(\bar{X}, \bar{p})$ in $\bar{G}$. Set $\bar{X} = \zeta(X)$, and let the prices remain the same for all arcs, i.e., $\bar{p}_{\gamma(e)} = p_e$ for all $e \in E$. To establish that $(\bar{X}, \bar{p})$ is a competitive equilibrium, I next argue that for any agent i, in $\bar{G}$ under price vector $\bar{p}$ the set of payoff-maximizing trades is given by $\bar{\delta}(i) \cap \bar{X}$.

First, consider an agent $i \in N$ such that $\delta(i) \cap R = \emptyset$; i.e., i is not adjacent to the trades R in G. It readily follows that the adjacent trades and their prices are identical for agent i in network G (under price vector p) and in network $\bar{G}$ (under price vector $\bar{p}$). Since (X, p) is a competitive equilibrium, we know that $X \cap \delta(i)$ is a set of payoff-maximizing trades for agent i in G. These observations immediately imply that in $\bar{G}$, under price vector $\bar{p}$, the payoff-maximizing trades for i are given by $\bar{\delta}(i) \cap \bar{X}$.

Next consider $i \in N$ such that $\delta(i) \cap R \neq \emptyset$. In $\bar{G}$, under price vector $\bar{p}$, for any $\bar{Y} \subset \bar{\delta}(i)$ the payoff of agent i can be written as follows:

$$\begin{aligned}
\bar{u}_i(\bar{Y}, \bar{p}) &= \bar{w}_i(\bar{Y}) + \sum_{e \in \bar{Y} \cap \bar{\delta}_+(i)} \bar{p}_e - \sum_{e \in \bar{Y} \cap \bar{\delta}_-(i)} \bar{p}_e \\
&= w_i(\zeta^{-1}(\bar{Y}) \cap \delta(i)) - \sum_{e \in R \cap \delta_-(i)} p_e + \sum_{e \in R \cap \delta_+(i)} p_e + \sum_{e \in \bar{Y} \cap \bar{\delta}_+(i)} \bar{p}_e - \sum_{e \in \bar{Y} \cap \bar{\delta}_-(i)} \bar{p}_e \\
&= w_i(\zeta^{-1}(\bar{Y}) \cap \delta(i)) - \sum_{e \in R \cap \delta_-(i)} p_e + \sum_{e \in R \cap \delta_+(i)} p_e + \sum_{e \in (\bar{Y} \setminus \bar{R}) \cap \bar{\delta}_+(i)} \bar{p}_e - \sum_{e \in (\bar{Y} \setminus \bar{R}) \cap \bar{\delta}_-(i)} \bar{p}_e \\
&\quad + \sum_{e \in \bar{Y} \cap \bar{R} \cap \bar{\delta}_+(i)} \bar{p}_e - \sum_{e \in \bar{Y} \cap \bar{R} \cap \bar{\delta}_-(i)} \bar{p}_e,
\end{aligned} \tag{3.12}$$

where $\bar{u}_i$ denotes the payoff function of agent i in $\bar{G}$. Here, the first equality follows from the definition of the payoff function. The second one follows from the definition of $\bar{w}_i$. The third equality is obtained by rewriting the summations that involve the prices after splitting $\bar{Y}$ into $\bar{Y} \cap \bar{R}$ and $\bar{Y} \setminus \bar{R}$.

Recall that $\bar{p}_{\gamma(e)} = p_e$ for all $e \in E$. In addition, the definition of ζ implies that $\zeta^{-1}(\bar{Y} \setminus \bar{R}) = \zeta^{-1}(\bar{Y}) \setminus R$ and

$$\bar{Y} \cap \bar{R} = \{\gamma(e) | e \in R \setminus \zeta^{-1}(\bar{Y})\}.$$

Using these observations together with (3.12), I obtain:

$$\begin{aligned}
\bar{u}_i(\bar{Y}, \bar{p}) &= w_i(\zeta^{-1}(\bar{Y}) \cap \delta(i)) - \sum_{e \in R \cap \delta_-(i)} p_e + \sum_{e \in R \cap \delta_+(i)} p_e + \sum_{e \in (\zeta^{-1}(\bar{Y}) \setminus R) \cap \delta_+(i)} p_e - \sum_{e \in (\zeta^{-1}(\bar{Y}) \setminus R) \cap \delta_-(i)} p_e \\
&\quad + \sum_{e \in (R \setminus \zeta^{-1}(\bar{Y})) \cap \delta_-(i)} p_e - \sum_{e \in (R \setminus \zeta^{-1}(\bar{Y})) \cap \delta_+(i)} p_e \\
&= w_i(\zeta^{-1}(\bar{Y}) \cap \delta(i)) + \sum_{e \in \zeta^{-1}(\bar{Y}) \cap \delta_+(i)} p_e - \sum_{e \in \zeta^{-1}(\bar{Y}) \cap \delta_-(i)} p_e \\
&= u_i(\zeta^{-1}(\bar{Y}) \cap \delta(i), p).
\end{aligned}$$

Here, the second equality follows by collecting common terms together, and the last equality follows from the definition of the payoff function. Note that by setting $\bar{Y} = \bar{X} \cap \bar{\delta}(i)$, we have

$$\bar{u}_i(\bar{X}\cap\bar{\delta}(i), \bar{p}) = u_i(\zeta^{-1}(\bar{X}\cap\bar{\delta}(i))\cap\delta(i), p) = u_i(X\cap\delta(i), p) \geq u_i(\zeta^{-1}(Y')\cap\delta(i), p) = \bar{u}_i(Y', \bar{p}), \tag{3.13}$$

for any $Y' \subset \bar{\delta}(i)$. Here the second equality follows from the definition of $\bar{X}$, and the inequality follows since (X, p) is an equilibrium in G. On the other hand, since (3.13) holds for any $Y' \subset \bar{\delta}(i)$, I conclude that $\bar{X} \cap \bar{\delta}(i)$ is a payoff-maximizing set of trades in $\bar{G}$ under price vector $\bar{p}$. Thus, it follows that $(\bar{X}, \bar{p})$ is an equilibrium in $\bar{G}$.

Observe that, by construction, $\bar{X} \cap (L \cup \bar{R}) = \emptyset$, and hence the trades in $L \cup \bar{R}$ are not demanded in the competitive equilibrium $(\bar{X}, \bar{p})$ in $\bar{G}$. Thus, removing these trades without changing the prices of the remaining trades yields another equilibrium. Note that $\hat{G}$ is obtained by excluding trades $L \cup \bar{R}$ from $\bar{G}$. Denote the set of arcs in $\hat{G}$ by $\hat{E}$, and note that $\hat{E} \cup L \cup \bar{R} = \bar{E}$. Let $\hat{X} = \bar{X}$, and denote by $\hat{p}$ the price vector obtained by restricting $\bar{p}$ to $\hat{E}$, i.e., $\hat{p}_e = \bar{p}_e$ for $e \in \hat{E}$. The above observation implies that $(\hat{X}, \hat{p})$ is an equilibrium in $\hat{G}$.

Next, add $L \cup \bar{R} \cup \Delta$ to $\hat{G}$ to obtain $\tilde{G}$. It follows from Theorem 3.3.8 that there exists a new equilibrium $(\tilde{X}, \tilde{p})$ in $\tilde{G}$ for which $\tilde{p}(e) \geq \hat{p}(e)$ for all trades in E^s.

Finally, by replacing the trades in $\bar{R}$ with those in R, and replicating the approach I used when I transformed G to $\bar{G}$, it follows that there exists an equilibrium (X', p') in G' such that the prices of trades in $E^s \cup E^b$ are the same as the corresponding prices in equilibrium $(\tilde{X}, \tilde{p})$ in $\hat{G}$.

In sum, the equilibria that were constructed for the sequence of trading networks presented in Figure 9 are such that

$$p_e = \bar{p}_e = \hat{p}_e \leq \tilde{p}_e = p'_e$$

for any $e \in E^s$. Thus, I conclude that addition of the trades in Δ to G yields an equilibrium where the prices of trades in E^s weakly increase. Hence, it follows that the first result of Theorem 3.3.8 continues to hold in the setting described in the theorem statement.

Under an almost identical approach (but this time using the second finding of Theorem 3.3.8 in the transformation from $\hat{G}$ to $\tilde{G}$), it follows that the second result of Theorem 3.3.8 also carries over to the setting described in the theorem statement. Hence, the claim follows. $\square$

It can be readily seen that Theorem 3.3.8 is a special case of Theorem 3.3.9 with $L = R = \emptyset$. Qualitatively, this result suggests that the presence of unused trades (L) that have the same orientation as the trades added to the economy and the executed trades (R) in the opposite orientation do not impact the comparative statics in Theorem 3.3.8. In fact, this claim is formalized in the proof of Theorem 3.3.9. More precisely, I first show that the equilibrium in the initial trading network is identical to the equilibrium in a trading network where trades in R are replaced with trades with the opposite orientation (after a valuation transformation). In the equilibrium in the new network, the aforementioned trades are not executed. Leveraging this observation, I reduce the analysis of the impact of the trades in Δ on the competitive equilibrium of the initial network, to the setting of Theorem 3.3.8.

My findings suggest that leveraging the MSFP formulation is valuable for deriving nontrivial comparative statics and certain monotonicity properties of equilibrium prices. In addition to the change in the equilibrium prices, it is possible to characterize the change in the equilibrium payoff of an agent (or a group of agents). The latter is omitted here for brevity.

3.3.4. Discussion: Other Comparative Statics and Applications

I have illustrated how a competitive equilibrium changes with the addition of new trades/agents to the trading network. I close this section by outlining another set of comparative statics that the MSFP formulation can generate, as well as possible applications of my results.

In the classic network flow problem, it is possible to characterize the impact of the changes in (i) the arc capacities and (ii) the arc costs, on the optimal flow and vertex potential values (see, e.g., Ahuja et al. (1993)). Above I furnish results analogous to (i) in the context of MSFP (e.g., Lemma 3.3.1). The same techniques allow one to provide results analogous to (ii) as well. Following the proof of Lemma 3.3.1, one can show that in MSFP as the arc cost c_a for arc $a \in E$ decreases, either the optimal flow does not change (e.g., if the flow on a is equal to the capacity of the arc), or the flow on this arc increases by one unit, and the new optimal flow can be found by complementing the initial one with the shortest path with the least number of arcs from a^- to a^+ in the underlying auxiliary network. In the latter case, the new optimal vertex potential values can also be obtained by computing the length of the shortest path from a^- to the remaining arcs, similar to Lemma 3.3.1.

When I constructed the flow network associated with the underlying trading network, I set all arc costs of the flow network equal to zero (see Section 2.3). However, allowing positive arc costs enables one to model a variety of trade frictions, e.g., transportation costs or excise[8] taxes. In the presence of such trade frictions, a competitive equilibrium can be obtained by solving a version of the MSFP formulation with nonzero arc costs. Moreover, all of the results on the existence of a competitive equilibrium and its equivalence to (chain) stability immediately generalize to this setting. This equivalence fails in the non-quasilinear setting; see Fleiner et al. (2018).

Perhaps more interestingly, by combining the MSFP characterization of competitive equilibria and the sensitivity analyses outlined in the previous paragraph, it is possible to reason about the change in the equilibrium prices/payoffs as the aforementioned trade frictions increase/decrease. For instance, the impact of targeting some sectors of an economy with carbon taxes, on the market prices of the output of different sectors can be studied by combining the MSFP formulation and these sensitivity results (similar to the analysis of marginal tax changes in King et al. (2019)).

The comparative statics provided in this section also have immediate application to trade

[8]These taxes are independent of the transaction prices, and are imposed on oil, tobacco, alcohol, and certain chemicals like carbon; see, e.g., Wasserman et al. (1991) and Barthold (1994).

quotas. For instance, suppose that there is no trade between two countries (i.e., quotas are set equal to zero for all goods). Suppose that quotas are raised to facilitate trade of certain goods; what is the impact on the prices of different goods/services in the two countries/economies? The setting studied in the previous section – after interpreting G^s and G^b as the trading networks internal to different countries – precisely answers this question. For instance, Theorem 3.3.8 implies that when economy G^b raises quotas (for incoming goods), the buyers in this economy enjoy lower prices. Moreover, if these quotas are fully used and economy G^s in turn raises quotas, Theorem 3.3.9 implies that this results in an increase in the payoff of the buyers in economy G^s. I view the analysis of the impact of more realistic changes in quotas on the competitive equilibrium, using the machinery developed in this section, as an interesting future direction.

3.4. Concluding Remarks and Future Applications

In this work, I show that the problem of finding efficient sets of trades in trading networks can be formulated as an MSFP in a suitable network. The network flow formulation and the associated optimality conditions readily imply the known results on competitive equilibria in trading networks, such as their existence and equivalence to (chain) stability. This formulation also leads to algorithms for finding competitive equilibria and identifying blocking chains, when they exist. Moreover, by leveraging sensitivity analysis ideas from network flows I provide new comparative statics for trading networks.

In this work, the focus has been on quasilinear utilities/payoffs. For more general settings without transferable utilities, full substitutability of preferences is not sufficient for the existence of (chain) stable outcomes and its equivalence to the competitive equilibrium outcome (Hatfield et al., 2019b). For such settings, alternative solution concepts have been proposed in the literature; see, e.g., Fleiner et al. (2018). Exploring possible applications of network flow formulations in such more general settings is an interesting future direction.

CHAPTER 4 : Analytical Tools for Prophet Inequalities

Prophet Inequalities are a way to quantify the value of perfect information in optimal stopping problems that are distribution-free. The first prophet inequality was derived in the classic paper of Krengel and Sucheston (1977) in the context that I will call the basic stopping problem. Consider a sequence of n random variables that are interpreted as rewards. The random variables are sampled sequentially, and at each step, a *decision-maker* has to decide whether to stop and receive the reward or continue. The question of maximizing the expected reward constitutes the on-line problem. On the other hand, a *prophet* possesses complete clairvoyance of the rewards, i.e., has perfect information. The off-line problem is straightforward: select the largest realized reward. A prophet inequality bounds the ratio of the expected value of the on-line problem to the off-line problem from below. The higher this ratio, the less important perfect information is. Over time a huge literature has developed devoted to sharpening the original prophet inequality or deriving similar inequalities for other stopping problems. An interesting survey of results and techniques can be found in Hill and Kertz (1992) and later developments are presented by Lucier (2017).

In this chapter, a variety of different techniques that have been used to derive prophet inequalities are summarized. This will serve as a contrast to two new linear programming approaches that I have developed. The first is a primal approach, based on scaling the reduced form of the prophet's strategy. The second is a dual approach inspired by Davis and Karatzas (1994).

4.1. Basic Stopping: A Linear Programming Approach

In the basic stoppping problem, a sequence of n random variables, indexed by $I = \{1, \ldots, n\}$ and interpreted as rewards, are sampled sequentially, i.e., they are not known in advance. Reward r_i is revealed on step i. The realized rewards are independent. For most of this dissertation, I will assume that the rewards can take values from a finite set R_i, where the probability that reward r_i is realized is denoted by $f_i(r_i)$. [1] r refers to a full profile of rewards $r = (r_1, \dots, r_n)$. Denote by $r_{\leq i}$ the profile of rewards up to and including step i. $r_{<i}$ denotes the profile of rewards up to but not including step i.

A decision consists of a stopping rule based on the realized rewards up to the current step. A stopping rule specifies the probability $z_i(r_i)$ that the decision-maker stops at step i conditional on the event that she has not stopped in previous steps. Specifically, $z_i(t_i) = Pr[\text{select } r_i | 1, \dots, i-1 \text{ not stopped}]$. This fully captures the set of stopping rules, since independence implies that there is no need to condition the decision to stop on i upon $r_{<i}$.

A linear programming representation can be given by considering path-dependent decisions. Let $q_i(r_{\leq i})$ be the probability of stopping on $i \in I$ given the profile of rewards $r_{\leq i}$ was realized. These are called ex-post variables. The set of feasible stopping rules can be represented by a set of nonnegative values q such that the sum of the variables relevant to a full profile of rewards adds up to one. In more detail, the optimal stopping rule can be found as a solution to a linear program (LP), defined below.

$$\begin{aligned} V = \max \ & \mathbb{E}_r[\sum_{i \in I} r_i q_i(r_{\leq i})] \\ s.t. \ & \sum_{i \in I} q_i(r_{\leq i}) \leq 1 \quad \forall r \\ & q \geq 0. \end{aligned} \tag{LP}$$

The optimal stopping rule can be implemented by dynamic programming. It selects a reward r_i if it greater than the expected reward in future rounds.

Definition 4.1.1. Let v_i be the expected reward of the optimal stopping rule starting from i. A dynamic program will proceed backwards, computing the values $v_1, \dots, v_n$ recursively as follows,

$$v_i = \mathbb{E}_{r_i} \max\{r_i, v_{i+1}\}, \tag{4.1}$$

[1] However, it is sometimes convenient to work with continuous distributions in that certain quantities can be given succinct expressions.

with initial condition $v_{n+1} = 0$. With these values in hand, the optimal stopping rule can be easily implemented. The optimal stopping rule selects a reward r_i in step i if and only if $r_i \geq v_{i+1}$.

In this chapter, I will also consider the prophet's strategy. The prophet, possessed of complete clairvoyance on the rewards, selects the maximum reward for each full profile of rewards, therefore, given a profile of rewards r, the prophet gains $\max_i r_i$. The prophet's expected reward is

$$M = \mathbb{E}_r[\max_i r_i] \tag{4.2}$$

In a similar fashion to (LP), the prophet's strategy can be represented by a set of variables $\{w_i(r)\}_{i \in I, r \in R}$, such that for all profiles $\sum_i w_i(r) \leq 1$. Let w^* be the prophet's strategy, such that for each profile of rewards r all variables are set to zero but the one with the largest reward.

In the following sections, I study the optimal stopping rule, along with alternative strategies. A number of strategies that return a reward comparable to the prophet's expected reward will be presented.

4.2. Balanced Thresholds

I first consider rules based on setting a fixed threshold. The decision-maker selects a realized reward of r_i whenever it exceeds the fixed threshold. Several thresholds can be used to derive the prophet inequality. For simplicity, in this section, I assume that the rewards are sampled from a continuous distribution. The results hold for all types of distributions, subject to a few technicalities.

For the basic stopping problem, the total reward can be split between the value of the threshold and the decision-maker's surplus. Let $\tau(r)$ be the stopping time when profile r is realized. Let $p = \mathbb{P}_r[\max_i r_i \geq T]$. Then, the total reward of the above selection rule is bounded from below as follows,

$$\begin{aligned} V &= pT + \mathbb{E}_r[(r_{\tau(r)} - T)^+] = pT + \mathbb{E}_r[\sum_i (r_i - T)^+ \mathbb{I}(\tau(r) = i)] \\ &= pT + \mathbb{E}_r[\sum_i (r_i - T)^+ \mathbb{I}(\tau(r) > i-1)] = pT + \mathbb{E}_r[\sum_i (r_i - T)^+] \mathbb{P}[\tau(r) > i-1]] \\ &\geq pT + (1-p) \sum_i \mathbb{E}_{r_i}[(r_i - T)^+]. \end{aligned} \tag{4.3}$$

I summarize different thresholds that have appeared in the literature, and attempt to match (4.3) with the properties of each threshold. The thresholds appear in Table 4.2.

T_H	$T_H = \sum_i \mathbb{E}_{r_i}[(r_i - T_H)]^+$
T_L	$T_L = \mathbb{E}_r[\max_i r_i - T_L]^+$
T_{KW}	$\frac{1}{2}\mathbb{E}_r[\max_i r_i]$
T_{SK}	$\mathbb{P}_r[\max_i r_i \leq T_{mLOS}] \sim \frac{1}{2}$

Table 1: Balanced Thresholds

The basic stopping problem has an interpretation in terms of dynamic posted prices. Agents arrive sequentially. Agent i has a reservation value r_i, drawn from a distribution independently of the other agents. The threshold can be interpreted as a take-it-or-leave-it price. If the current agent's r_i is below the threshold, no sale takes place, and the seller moves to the next agent. If it is above, a sale is made, and the process stops. The expected payoff of optimal stopping or a threshold-based strategy is the expected payoff of the seller posting the prices plus the buyers' surplus. The expected payoff of the prophet corresponds to the expected payoff of a seller who knows the reservation values of each buyer in advance and can, therefore, charge them their full value.

4.2.1. Median of Largest Order Statistic and an Interval of Prices (Samuel-Cahn, 1984)

Samuel-Cahn (1984) provides a single threshold T_{SK}, and examines the reward of a decision-maker who stops when she observes a reward $r_i \geq T_{SK}$. Set the threshold such that

$$\mathbb{P}_r[\max_i r_i < T_{SK}] \leq \frac{1}{2} \text{ and } \mathbb{P}_r[\max_i r_i > T_{SK}] \leq \frac{1}{2}.$$

From the assumption of continuous distributions, it follows that, $p = 1 - p = \frac{1}{2}$. Let $s = \sum_i \mathbb{E}_{r_i}[r_i - T_{SK}]^+$. The proof follows,

$$
\begin{aligned}
V &\geq pT_{SK} + (1-p)\sum_i \mathbb{E}_{r_i}[(r_i - T_{SK})^+] = pT_{SK} + (1-p)s \\
&= \frac{1}{2}(T_{SK} + s) \geq \frac{1}{2}(T_{SK} + \mathbb{E}_r[\max_i r_i - T_{SK}]^+) \\
&= \frac{1}{2}\mathbb{E}_r[\max_i r_i] = \frac{1}{2}M.
\end{aligned}
$$

Interestingly, Samuel-Cahn (1984) also provides a separate interval of numbers $[T_L, T_H]$ such that all the numbers in the interval can serve as thresholds to derive the prophet inequality. Consider two thresholds T_L, T_H which can be given as fixed points of two different functions ϕ_L, ϕ_H, respectively, i.e., $T_L = \phi_L(T_L)$ and $T_H = \phi_H(T_H)$. The two functions are

- $\phi_L(T) = \mathbb{E}_r[(\max_i r_i - T)^+]$ and
- $\phi_H(T) = \mathbb{E}_r[\sum_i (r_i - T)^+]$.

Both of the functions admit fixed points, since the continuity of the distribution functions implies the continuity of the functions ϕ_L, ϕ_H.

The two thresholds are related and bounded from below as shown in Lemma 4.4.

Lemma 4.2.1. T_L is always less than T_H and bounded from below by $\frac{1}{2}M$, i.e.,

$$T_H \geq T_L \geq \frac{1}{2}M \tag{4.4}$$

Proof. Suppose that $T_L > T_H$. This implies that

$$T_L > T_H = \mathbb{E}_r[\sum_i (r_i - T_H)^+] \geq \mathbb{E}_r[\max_i r_i - T_H]^+ \geq \mathbb{E}_r[(\max_i r_i - T_L)^+] = T_L,$$

which is a contradiction.

A similar proof can be given for the lower bound. Suppose that $T_L < \frac{1}{2}M$. Then,

$$\frac{1}{2}M > T_L = \mathbb{E}_r[\max_i r_i - T_L]^+ \geq \mathbb{E}_r[\max_i r_i - \frac{1}{2}M]^+ \geq \mathbb{E}_r[\max_i r_i - \frac{1}{2}M] = \frac{1}{2}M,$$

which is a contradiction. □

I combine 4.4 with inequality 4.3 to show that, for any threshold $T \in [T_L, T_H]$, the expected reward is at least half of the prophet's rewards,

$$\begin{aligned} V &\geq pT + (1-p)\sum_i \mathbb{E}_{r_i}[(r_i - T)^+] \geq pT_L + (1-p)\sum_i \mathbb{E}_{r_i}[(r_i - T)^+] \\ &\geq pT_L + (1-p)T_H \geq T_L \geq \frac{1}{2}M. \end{aligned}$$

4.2.2. Half the Largest Order Statistic (Kleinberg and Weinberg, 2019)

Kleinberg and Weinberg (2019) also provide a (different) threshold in order to derive the classic prophet inequality. The strategy of the decision-maker is as before. The threshold is set to

$$T_{KW} = \frac{1}{2}\mathbb{E}_r[\max_i r_i].$$

Observe that $\mathbb{E}_r[(\max_i r_i - T_{KW})^+] \geq \mathbb{E}_r[\max_i r_i - T_{KW}] = T_{KW}$, where the equality follows from the linearity of expectations and the definition of T_{KW}. Thus, I conclude that

$$\begin{aligned} V &\geq pT_{KW} + (1-p)\mathbb{E}_{r_i}[(r_i - T_{KW})^+] \geq pT_{KW} + (1-p)T_{KW} \\ &= \frac{1}{2}\mathbb{E}_r[\max_i r_i] = \frac{1}{2}M. \end{aligned}$$

Remark. Both proofs discussed are threshold-based. Given that a threshold is a price, both proofs initially separate the agents' surplus and the payoff generated by the decision-maker. Then each quantity can be bounded below, which gives the prophet inequality. Given that the thresholds differ, the two proofs differ in the way they lower bound the two quantities. However, in all cases, a balance is attained, which is more evident in the second proof. On the other hand, the threshold in the first case is more robust to outlier values.

4.3. Duality Theory

I provide an alternative proof of the classic prophet inequality, leveraging duality theory. The analysis directly extends to continuous distributions and linear functional programming. First, a linear program describing the optimal stopping rule is presented, along with its dual and auxiliary representations of them. Then, strong duality and properties of the optimal dual variables are utilized to derive the classic prophet inequality.

Consider an alternative representation of the optimal on-line resource allocation problem, denoted by (ALP).

$$\begin{aligned} V = \max\ & \mathbb{E}_r[\sum_i r_i w_i(r)] \\ s.t.\ & \sum_i w_i(r) \leq 1 \quad \forall r \\ & w_i(r) = q_i(r_1, \ldots, r_i) \quad \forall i \quad \forall r \\ & w, q \geq 0 \end{aligned} \tag{ALP}$$

The formulation is due to Davis and Karatzas (1994). Here the set of variables is expanded by including a set of variables w that depend on future rewards too and a set of new constraints restricting them to have the same value when the history is the same.

Consider a partial dual function $\mathcal{L}(\lambda)$, denoted by (LLP).

$$\begin{aligned} \mathcal{L}(\lambda) = \max\ & \mathbb{E}_r[\sum_i (r_i + \lambda_i(r)) w_i(r)] - \sum_i \mathbb{E}_{r_1,\ldots,r_i}[\mathbb{E}_{r_{i+1},\ldots,r_n}[\lambda_i(r)] q_i(r_1, \ldots, r_i)] \\ s.t.\ & \sum_i w_i(r) \leq 1 \quad \forall r \\ & w, q \geq 0. \end{aligned} \tag{LLP}$$

Weak duality implies that the dual function serves as an upper bound to the optimal solution V. However, strong duality implies that for optimal dual variables the upper bound matches the optimal value, i.e., there exists λ^* such that $\mathcal{L}(\lambda^*) = V$.

Optimal dual variables can be found utilizing the dual problem denoted by (DP).

$$\begin{aligned}
&\min \mathbb{E}_r[\pi(r)] \\
&s.t.\ \pi(r) \geq \lambda_i(r) + r_i \\
&\qquad \mathbb{E}_{r_{i+1},\ldots,r_n}[\lambda_i(r)] \geq 0 \\
&\qquad \pi \geq 0
\end{aligned} \tag{DP}$$

By examining (DP) it is clear that solving it reduces to computing optimal variables π. (DP) can be projected to the space of π and the dual variables λ can be chosen maximally to support them. In more detail, given an optimal solution π^*, optimal dual variables λ^* can be chosen as follows,

$$\lambda_i^*(r) = \pi^*(r) - r_i. \tag{4.5}$$

The projection of the dual problem to the space of π variables is denoted by (PDP).

$$\begin{aligned}
&\min \mathbb{E}_r[\pi(r)] \\
&s.t.\ \mathbb{E}_{r_{i+1},\ldots,r_n}[\pi(r)] \geq r_i \quad \forall i \quad \forall r_1, \ldots, r_i \\
&\qquad \pi \geq 0
\end{aligned} \tag{PDP}$$

Finally, consider the function

$$g(\lambda, w) = \mathbb{E}_r[\sum_i (r_i + \lambda_i(r)) w_i(r)]. \tag{4.6}$$

The above primal-dual definitions are used to provide a proof of the prophet inequality. First, a few helpful lemmas are presented.

Lemma 4.3.1. Given an optimal solution λ^*, the optimal value V can be given by

$$\begin{aligned} V = \max\ & \mathbb{E}_r[\sum_i (r_i + \lambda_i^*(r)) w_i(r)] \\ s.t.\ & \sum_i w_i(r) \le b \quad \forall r \\ & w \ge 0. \end{aligned} \tag{ALLP}$$

Proof. The proof is short. Since λ^* is optimal, it is clear from (DP) that $\mathbb{E}_{r_{i+1},\ldots,r_n} \lambda_i^*(r) \ge 0$ for all i and $r_1, \ldots, r_i$. Hence, the optimal q in (LLP) is zero. Hence, the second term in the objective cancels out, and (LLP) reduces to the form stated in the lemma. □

Lemma 4.3.2. An optimal solution to (DP) is given by

$$\begin{aligned} \pi^*(r) &= \sum_i [r_i - v_{i+1}]^+, \\ \lambda_i^*(r) &= \pi(r) - r_i = \sum_i [r_i - v_{i+1}]^+ - r_i. \end{aligned} \tag{4.7}$$

Proof. The above values can be constructed inductively by iteratively adding increments to the variables π to satisfy the constraints. However, I prove their optimality by showing that they are feasible and the objective becomes equal to V.

First, it is shown that the proposed π^* is a feasible solution to (PDP).

$$\begin{aligned} \mathbb{E}_{r_{i+1},\ldots,r_n}[\pi(r)] = & \sum_{j \le i} [r_j - v_{j+1}]^+ + \sum_{j > i} \mathbb{E}_{r_j} [r_j - v_{j+1}]^+ \\ & \ge [r_i - v_{i+1}]^+ + \sum_{j>i} \mathbb{E}_{r_j} [r_j - v_{j+1}]^+ \\ & = [r_i - v_{i+1}]^+ + \sum_{j>i} (v_j - v_{j+1}) \\ & = [r_i - v_{i+1}]^+ + v_{i+1} \\ & \ge r_i. \end{aligned}$$

The first equality follows by linearity of expectations and the first inequality follows by omitting positive terms before i. The second equality comes from the definition of the values $v_{i+1}, \dots, v_n$ and the third equality by canceling terms in the summation. The last inequality is straightforward. Furthermore, λ^* is feasible too, by definition.

Last, the objective is equal to

$$\begin{aligned}
\mathbb{E}_r[\pi(r)] &= \mathbb{E}_r[\sum_i [r_i - v_{i+1}]^+] \\
&= \sum_i \mathbb{E}_{r_i}[r_i - v_{i+1}]^+ \\
&= V.
\end{aligned}$$

The last equality holds because a dynamic program will choose i if the current reward is larger than the expected reward in future rounds. This completes the proof. □

Lemma 4.3.3. Consider λ^*, as proposed in (4.7), and let w^* be the prophet's decision. Then

$$\mathbb{E}_r[\sum_i \lambda_i(r) w_i^*(r)] \geq -V. \tag{4.8}$$

Proof. $\lambda_i^*(r)$ can be lower bounded for all i and r. Specifically,

$$\begin{aligned}
\lambda_i^*(r) = & \sum_j [r_j - v_{j+1}]^+ - r_i \\
& \geq [r_i - v_{i+1}]^+ - r_i + \sum_{j>i} [r_j - v_{j+1}]^+ \\
& = [r_i - v_{i+1}]^+ - r_i + \sum_{j>i} (v_j - v_{j+1}) \\
& = [r_i - v_{i+1}]^+ - r_i + v_{i+1} \\
& \geq\ - v_{i+1} \geq -V.
\end{aligned}$$

To complete the proof,

$$\mathbb{E}_r[\sum_r \lambda_i(r) w_i^*(r)] \geq -V \mathbb{E}_r[\sum_i w_i^*(r)] \geq -V.$$

The first inequality holds since w^* is nonnegative and the second inequality holds because $\sum_i w_i^*(r) \leq 1$ for all r. □

By leveraging the above lemmas the prophet inequality can be derived.

Theorem 4.3.4. The reward of the optimal stopping rule is at least half of the prophet's reward in expectation, i.e., $V \geq \frac{M}{2}$.

Proof. A few more steps of analysis will suffice to bound the optimal expected reward from below.

$$\begin{aligned} V \geq g(\lambda^*, w^*) &= \mathbb{E}_r[\sum_i r_i w_i^*(r) + \sum_i \lambda_i^*(r) w_i^*(r)] \\ &= M + \mathbb{E}_r[\sum_i \lambda_i^*(r) w_i^*(r)] \geq M - V \end{aligned}$$

The first inequality comes from Lemma 4.3.1 and the feasibility of w^*, the second equality comes form the definition of the prophet's reward, and the last inequality comes from Lemma 4.3.3. The above inequality gives an approximation guarantee of $\frac{1}{2}$. □

The proof is closely related to the one by Davis and Karatzas (1994). They first show the equivalence of (ALP) and (ALLP), by explicitly constructing the dual variables and the associated stopping rules. I sidestep this part by showing that the equivalence is a simple byproduct of strong duality. For the remainder, both of the proofs are based on bounding the dual variables from below. Davis and Karatzas (1994) manages to do so for a suitable choice of dual variables. Here a different set of dual values is provided, which is chosen maximally. My hope is that these additional insights will be useful to address the problem in more general settings.

Remark. Notice that most of the steps along the proof can be extended to the case of multiple resource constraints. The crucial step in the above analysis is pinning down the optimal dual variables π^*. This is the step that requires careful thought when generalizing to multiple constraints. The application of strong duality as well as determining optimal dual variables λ^* carry on.

4.4. The Reduced-Form Representation

I describe a more compact primal formulation using a new set of variables $\{Q_j(r_j)\}_{i \in I, r_i \in R_i}$. The new set of variables are related to the ex-post variables q as follows,

$$Q_j(r_j) = \mathbb{E}_{r_1,\ldots,r_{j-1}}[q_j(r_1,\ldots,r_j)].$$

The variables $Q_j(r_j)$ are sometimes called interim variables. A formulation of the stopping problem in terms of the interim variables is called *the reduced-form representation.* This is the first work to express the reduced-form representation of the basic stopping problem. Similarly, a more compact formulation of the prophet's problem can be given using variables $\{W_j(r_j)\}_{i \in I, r_i \in R_i}$, related to the prophet's ex-post variables as follows,

$$W_j(r_j) = \mathbb{E}_{r_{-j}}[w_j(r)].$$

The above set of variables provide a tractable description of the decision-maker's and prophet's problems, in the form of a polytope in polynomially many variables.

Given the above set of reduced variables, I examine the reduced-form representation of the decision-maker's and prophet's problems. When it comes to the prophet's problem, the answer has already been given in (C Border, 1991), where a set of constraints is formed using Hall's theorem to match interim allocations with ex-post allocations. The formulation is given in PRF.

$$\begin{aligned}
\max \ & \sum_i \mathbb{E}_{r_i}[r_i W_i(r_i)] \\
s.t. \ & \sum_i \sum_{r_i \in S_i} f_i(r_i) W_i(r_i) \le 1 - \prod_i (1 - \sum_{r_i \in S_i} f_i(r_i)) \quad \forall S \subseteq R \\
& W \ge 0.
\end{aligned} \tag{PRF}$$

I now examine the projection of (LP) to the interim space. The use of interim variables is possible because of the independence of the rewards. The objective can be written in terms of the interim variables quite easily as the aggregate weighted average of the rewards in each round scaled by the interim variables. The projection of (LP) to the space of interim variables can be formulated as a set of constraints parameterized by the reward in each round.

Lemma 4.4.1. A set of interim values Q is feasible if and only if for all i and r_i,

$$Q_i(r_i) + \sum_{j>i} \mathbb{E}_{r_j}[Q_j(r_j)] \le 1. \tag{4.9}$$

Proof. Consider a set of decisions z supporting Q. For each i and r_i the interim value is given by $Q_i(r_i) = z_i(r_i)\mathbb{E}_{r_{<i}}[\prod_{j<i}(1 - z_j(r_j))]$. Hence,

$$z_i(r_i) = \frac{Q_i(r_i)}{\mathbb{E}_{r_{<i}}[\prod_{j<i}(1 - z_j(r_j))]}.$$

I will prove by induction that

$$\mathbb{E}_{r_{\le i}}[\prod_{j \le i}(1 - z_j(r_j))] = 1 - \sum_{j \le i} \mathbb{E}_{r_j}[Q_j(r_j)].$$

For $i = 1$, the equality reduces to $\mathbb{E}_{r_1}[1 - z_1(r_1)] = 1 - \mathbb{E}_{r_1}[Q_1(r_1)]$, which holds since

$Q_1(r_1) = z_1(r_1)$. Given that the equality holds for i, I prove the equality for $i+1$,

$$\begin{aligned}
\mathbb{E}_{r_{\leq i+1}}[\prod_{j\leq i+1}(1-z_j(r_j))] &= \mathbb{E}_{r_{i+1}}[(1-z_{i+1}(r_{i+1}))]\mathbb{E}_{r_{\leq i}}[\prod_{j\leq i}(1-z_j(r_j))] \\
&= (1-\mathbb{E}_{r_{i+1}}[z_{i+1}(r_{i+1})])\mathbb{E}_{r_{\leq i}}[\prod_{j\leq i}(1-z_j(r_j))] \\
&= \mathbb{E}_{r_{\leq i}}[\prod_{j\leq i}(1-z_j(r_j))] - \mathbb{E}_{r_{i+1}}[z_{i+1}(r_{i+1})]\mathbb{E}_{r_{\leq i}}[\prod_{j\leq i}(1-z_j(r_j))] \\
&= 1-\sum_{j\leq i}\mathbb{E}_{r_j}[Q_j(r_j)] - \mathbb{E}_{r_{i+1}}[Q_i(r_{i+1})] \\
&= 1-\sum_{j\leq i+1}\mathbb{E}_{t_j}[Q_j(t_j)].
\end{aligned}$$

The first equality follows from independence, the second equality follows from linearity of expectations, and the fourth equality follows from the inductive step and the definition of the interim variables. Hence, a set of interim values points out to a strategy z such that for all i and r_i,

$$z_i(r_i) = \frac{Q_i(r_i)}{1-\sum_{j<i}\mathbb{E}_{r_j}[Q_j(r_j)]}.$$

The only constraint tying z is $z_i(r_i) \leq 1$. After substituting with the expression of interim values, the desired inequality is revealed. □

The reduced form representation of (LP) is described as follows,

$$\begin{aligned}
&\max \sum_i \mathbb{E}_{r_i}[r_i Q_i(r_i)] \\
&s.t.\ Q_i(r_i) + \sum_{j>i}\mathbb{E}_{r_j}[Q_j(r_j)] \quad \forall i \quad \forall r_i \\
&\quad Q \geq 0.
\end{aligned} \tag{RF}$$

4.4.1. Weakly Coupled LP Relaxations

I describe a proof of the classic prophet inequality, attributed to Guha and Munagala (2007), which utilizes a weakly coupled linear programming relaxation of the prophet's problem (PRF). Threshold T_H will be chosen, as in Samuel-Cahn (1984), such that

$$T_H = \sum_i \mathbb{E}_{r_i}[(r_i - T_H)^+].$$

The threshold might be the same as in the previous proof, but the proof differs. I will assume that the distribution functions are continuous to guarantee that T_H exists.

The proof of an approximation guarantee for the above rule makes use of an linear programming relaxation of (PRF). Let $W_i(r_i)$ be the probability that i is selected given that r_i is realized. The linear programming relaxation is denoted by (RPRF).

$$\begin{aligned} \max \ & \sum_i \mathbb{E}_{r_i} r_i W_i(r_i) \\ s.t. \ & \sum_i \mathbb{E}_{r_i} W_i(r_i) \leq 1 \\ & 0 \leq W_i(r_i) \leq 1 \quad \forall i \quad \forall r_i \end{aligned} \tag{RPRF}$$

Let T be the dual variable corresponding to the first constraint. A partial dual function of RPRF is given by

$$\begin{aligned} \mathcal{L}(T) = \max T + \sum_i \mathbb{E}_{r_i}[(r_i - T) W_i(r_i)] \\ s.t. \ 0 \leq W_i(r_i) \leq 1 \quad \forall i \quad \forall r_i \end{aligned}$$

Solving the maximization problem implies that $\mathcal{L}(T) = T + \sum_i \mathbb{E}_{r_i}[(r_i - T)^+]$. Weak duality implies that $\mathcal{L}(T) \geq M$. Recall from Lemma 4.4 that $T_H = \sum_i \mathbb{E}_{r_i}[(r_i - T_H)^+] \geq \frac{1}{2} M$.

The approximation guarantee comes from an amortized analysis of the final reward. The expected reward is bounded below,

$$V \geq pT_H + (1-p)\sum_i \mathbb{E}_{r_i}[(r_i - T_H)^+] =$$
$$= p\frac{1}{2}M + (1-p)\frac{1}{2}M \geq \frac{1}{2}M.$$

4.4.2. New Algorithms utilizing the Reduced Form

I now describe a novel approach utilizing the reduced-form representation of the prophet's strategy to devise approximately efficient strategies. I will devise three new algorithms that perform well in three different settings, the basic stopping problem, a case where the rewards are sampled from the same distribution, and a case where the decision-maker can first order the agents appropriately and then start selecting in that order.

Basic Setting

For the classic setting, consider the strategy $Q^{\mathcal{A}}$ which scales the prophet's strategy by half, i.e., for all i and rewards r_i,

$$Q_i^{\mathcal{A}}(r_i) = \frac{1}{2}W_i^*(r_i).$$

The above algorithm is crude but achieves approximate efficiency. The approximation guarantee achieved carries on to the optimal stopping rule, as a consequence.

Theorem 4.4.2. Algorithm $\mathcal{A}$ achieves at least $1/2$ of the prophet's reward in expectation.

Proof. It is clear that the objective function with respect to the reduced form for both problems is linear and coincides. Thus, a simple scaling approximates the optimal objective,

$$\sum_{i\in I}\mathbb{E}_{r_i}[Q_i^{\mathcal{A}}(r_i)r_i] = \frac{1}{2}\sum_{i\in I}\mathbb{E}_{r_i}[W_i^*(r_i)r_i].$$

The proposed solution is feasible for the stopping problem. In detail, $Q_i^{\mathcal{A}}(r_i)+\sum_{j<i}\mathbb{E}_{r_j}[Q_j^{\mathcal{A}}(r_j)] = \frac{1}{2}W_i^*(r_i) + \frac{1}{2}\sum_{j<i}\mathbb{E}_{r_j}[W_j^*(r_j)] \leq 1$. The last inequality holds since $W_i^*(r_i) \leq 1$ and the prob-

ability of stopping in the first $i-1$ steps is also less than 1. $\square$

IID Setting

If one imposes restrictions on the distribution over rewards one might expect improved prophet inequalities. For the case when the distributions are identical, I show how using the optimal interim variables of the off-line problem can lead to informative proofs of the prophet inequality. For the IID setting, consider a rule $\mathcal{B}$ based on stopping on i with reward r_i with probability $z_i^{\mathcal{B}}(r_i) = W_i^*(r_i)$. The algorithm achieves approximate efficiency when compared to the prophet's reward, but it is not the best bound found in the literature. A first reference on the problem can be found in Hill and Kertz (1982), where the prophet inequality presented matches the one presented here. For the latest developments on the basic stopping problem with identical distributions, see Abolhassani et al. (2017).

Theorem 4.4.3. Algorithm $\mathcal{B}$ achieves at least $1-\frac{1}{e}$ of the prophet's reward in expectation.

Proof. Consider a symmetric version of the prophet's strategy, such that $W_i^*(r) = W_j^*(r) = W^*(r)$ for all i and r_i. The prophet's expected reward is equal to $M = n\mathbb{E}_r[W^*(r)]$. The reward gained by $\mathcal{B}$ can be bounded from below as follows,

$$\begin{aligned}
V^{\mathcal{B}} &= \sum_i \mathbb{E}_{r_i}[r_i W^*(r_i)] \prod_{j<i} (1 - \mathbb{E}_{r_j}[W^*(r_j)]) \\
&= \sum_i \mathbb{E}_r[rW^*(r)](1 - \mathbb{E}_r[W^*(r)])^{i-1} \\
&= \mathbb{E}_r[rW^*(r)] \sum_i (1 - \mathbb{E}_r[W^*(r)])^{i-1} \\
&= \frac{M}{n} \sum_i (1 - \frac{1}{n})^{i-1} = (1 - (1 - \frac{1}{n})^n)M \\
&\geq (1 - \frac{1}{e})M.
\end{aligned}$$

$\square$

Variable Arrival Order Setting

The basic stopping problem where the decision-maker can choose the arrival order of the rewards was presented by Yan (2011). I show an alternative proof of the prophet inequality for this case, which matches the best possible. Consider a strategy $\mathcal{C}$ that selects the order of the rewards processed and then proceeds with examining the sequence of rewards and deciding which to select. The decision-maker orders the agents in decreasing order of $k_i = \frac{\mathbb{E}_{r_i}[r_i W_i^*(r_i)]}{\mathbb{E}_{r_i}[W_i^*(r_i)]}$. Conditionally on the event of reaching i, the decision-maker selects r_i with probability $z_i(r_i) = W_i^*(r_i)$. The expected reward is bounded below by a fraction of the prophet's expected reward.

Theorem 4.4.4. Algorithm $\mathcal{C}$ achieves at least $1 - \frac{1}{e}$ of the prophet's reward in expectation.

Proof. I assume that the rewards are ordered such that $k_i \geq k_{i+1}$ for all i. As before I analyze the interim values under $\mathcal{C}$,

$$Q_i^{\mathcal{C}}(r_i) = W_i^*(r_i) \prod_{j<i} (1 - \mathbb{E}_{r_j} W_j^*(r_j)).$$

Let $a_i = \mathbb{E}_{r_i}[r_i W_i^*(r_i)], b_i = \mathbb{E}_{r_i}[W_i^*(r_i)]$, and $c_i = \prod_{j<i}(1 - \mathbb{E}_{r_j}[W_j^*(r_j)]) = \prod_{j<i}(1 - b_j)$. The prophet's reward can be written as $M = \sum_i a_i$. The expected reward of $\mathcal{C}$ is given by

$$\begin{aligned} V^{\mathcal{C}} &= \sum_i \mathbb{E}_{r_i}[W_i^*(r_i)] \prod_{j<i}(1 - \mathbb{E}_{r_j}[W_j^*(r_j)]) = \sum_i a_i c_i \\ &\geq \sum_i a_i \sum_i b_i c_i = M \sum_i b_i c_i = M \sum_i b_i c_i \\ &= M \sum_i b_i \prod_{j<i}(1 - b_j) = M(1 - \prod_i (1 - b_i)) \\ &\geq M(1 - \prod_i e^{-b_i}) = M(1 - e^{-\sum_i b_i}) \geq (1 - \frac{1}{e})M. \end{aligned}$$

The first two equalities come from the definition of $\mathcal{C}$ and the definition of a, c. I claim that the first inequality holds, which I will prove shortly. The third equality follows from the definition of M in terms of a and the forth equality follows since by optimality $\sum_i \mathbb{E}_{r_i}[W_i^*(r_i)] = 1$. The fifth equality follows by the definition of c in terms of b and the sixth equality by simplifying the expression. The second inequality follows by the standard trick, and the last inequality again follows since the prophet always utilizes the resource.

It remains to prove that $\sum_i a_i c_i \geq \sum_i a_i \sum_i b_i c_i$. I will use the fact that $k_i = \frac{a_i}{b_i}$ and c_i are both decreasing sequences. The inequality is rewritten in a better form,

$$\sum_i b_i \frac{a_i}{b_i}[c_i - \sum_j b_j c_j] \geq 0.$$

Set $x_i = c_i$ and $y_i = \frac{a_i}{b_i}$ for all i. Let h be an increasing function satisfying $y_i = h(x_i)$ for all i. This is possible because both sequences x and y are increasing. The inequality holds, as follows,

$$\begin{aligned}\sum_i b_i \frac{a_i}{b_i}[c_i - \sum_j b_j c_j] &= \sum_i b_i y_i [x_i - \sum_j b_j x_j] = \sum_i b_i h(x_i)[x_i - \sum_j b_j x_j] \\ &= \sum_i b_i (h(x_i) - h(\sum_j x_j)))(x_i - \sum_j b_j x_j) \geq 0.\end{aligned}$$

The first equality follows from the definition of x and y, and the second equality follows from the definition of h. The third equality holds because $\sum_i b_i(x_i - \sum_j b_j x_j) = 0$. The inequality holds because h is increasing, i.e., $(x_i - \sum_j b_j x_j)(h(x_i) - h(\sum_j b_j x_j)) \geq 0$ for all i. $\square$

CHAPTER 5 : Optimal On-line Verification Rules

In many large organizations, scarce resources must be allocated internally without the benefit of prices. Examples include the headquarters of a firm that must choose between multiple investment proposals from each of its division managers and funding agencies allocating a grant among researchers. In these settings, the private information needed to determine the right allocation resides with the agents, and the principal must rely on verification of agents' claims, which can be costly. I interpret verification as acquiring information (e.g., requesting documentation, interviewing an agent, or monitoring an agent at work), which can be costly. The headquarters of the diversified firm can hire an external firm to conduct an assessment of any division manager's claims, for example. The funding agency must allocate time to evaluate the claims of the researcher applying for a grant. Furthermore, in these settings, the principal can punish an agent if his claim is found to be false. For example, the head of personnel can reject an applicant, fire an employee, or deny promotion. Funding agencies can cut off funding.

Prior work considered an off-line version of this problem. Specifically, there is a principal who has to allocate one indivisible object among a finite number of agents, all of whom are present. The value to the principal of assigning the object to a particular agent is the private information of the agent. Each agent prefers to possess the object than not. The principal would like to give the object to the agent who has the highest value to her. Ben-Porath et al. (2014), the first to pose the question, assumes punishment is unlimited in the sense that an agent can be rejected and not receive the resource. Punishment can be limited because verification is imperfect or information arrives only after an agent has been hired for a while. In Mylovanov and Zapechelnyuk (2017), verification is free, but punishment is limited. Li (2020) generalizes both papers by incorporating costly verification *and* limited punishment.

This section introduces and analyzes an on-line version of this problem in which the agents arrive and depart one at a time, and the decision to allocate the object to an agent must be made upon the arrival of an agent. If the principal declines to allocate the object to an agent, the agent departs and cannot be recalled. If the principal allocates the object to an agent, the decision is irreversible. The problem is analogous to the problem of choosing a selling mechanism when facing a stream of buyers who arrive over time (see, for example, Gershkov and Moldovanu (2014)), except we do not have access to monetary transfers.

If each agent were to truthfully report the value to the principal, the principal faces the stopping problem. The goal is to select a single element with maximum value. An element of the sequence must be selected or discarded upon its arrival, and this decision is irrevocable. Recall that the solution involves a sequence of thresholds, indexed by the agent, and the principal allocates the object to the first agent whose reported value exceeds their corresponding threshold.

If the principal were to adopt such a policy in this setting, it would encourage all agents to exaggerate their values. To discourage this, the principal can ration at the top of the distribution of values or verify an agent's claim and punish him if his claim is found to be false. The first reduces allocative efficiency while the second is costly. This work aims to find the optimal way to provide incentives via these two devices in an on-line setting. The contributions of this work are as follows:

1. A reformulation of the on-line problem as a compact linear program that may be useful in other applications.

2. This reformulation allows us to derive a prophet inequality for the on-line version of the verification problem.

This setting is related to three lines of work. The first is on costly verification that begins with Townsend (1979). This work and others that followed such as Gale and Hellwig (1985), and Mookherjee and Png (1989), analyze off-line settings with transfers, which I rule out.

The second is on partial but costless verification, see for example Caragiannis et al. (2012) or Ball and Kattwinkel (2019), for example. In these models, verification is costless but imperfect. In my model, verification is perfect but costly. At a high level, the two are related because one can think of partial verification as being costly, but the cost is endogenous, depending on the nature of the realized allocation. In my case, the cost is exogenous.

Finally, it is related to the extensive literature on on-line selection problems. The absence of money in my setting means that the results from these papers do not apply.

In Section 5.1, I introduce my setting and the linear programming formulation. In Section 5.2, I characterize the form of the optimal mechanism and provide a corresponding prophet inequality. In Section 5.3, I study the variation of the problem with limited punishment.

5.1. Model

There is a single indivisible good to allocate among a set of agents denoted by $I = \{1, \ldots, n\}$. The type of agent $i \in I$ is t_i which is the value to the principal of allocating the object to agent i. I assume that the agents' types are independently distributed. The distribution of agent's i type has strictly positive density f_i over the interval $T_i = [\underline{t}_i, \bar{t}_i]$. The preferences of the agents are simple: each prefers to possess the object to not. The actual private benefit enjoyed by an agent from receiving the object does not need to be specified.

Agents arrive one after the other and report their type, not necessarily truthfully. The principal can verify the reported type of agent i at cost $c > 0$ and determine perfectly if the agent has lied. In the event an agent is discovered to have lied, the object is withheld from them. This is the case of unlimited punishment. The case of limited punishment is considered later.

By the revelation principle we can restrict attention to direct mechanisms. Denote by $t_{\leq i}$ the profile of reported types made by all agents up to and including agent i. I write $t_{<i}$ to denote the profile of reported types made by all agents up to but not including

i. A direct mechanism specifies for each profile of type reports, an allocation rule and an verification rule for each agent i. The allocation rule specifies the probability $z_i(t_i)$ he is allocated the good conditional on the event that the good is not already allocated. Specifically, $z_i(t_i) = Pr[\text{choose } t_i | 1, \ldots, i-1 \text{ not allocated}]$. This fully captures the set of on-line allocation rules, since independence means there is no need to condition the decision to allocate the good to agent i upon $t_{<i}$. The verification rule is the probability that agent i is assigned the good *and* inspected conditional on the event that the good is not already allocated and denoted $a_i(t_i)$. Therefore:

$$0 \leq a_i(t_i) \leq z_i(t_i) \leq 1 \quad \forall i \in I \quad \forall t_i \in T_i. \tag{5.1}$$

Definition 5.1.1. A direct mechanism $\mathcal{M} = (T_1, \ldots, T_{|I|}, \{z_i(\cdot), a_i(\cdot)\}_{i \in I})$ restricts the strategy set of each agent i to T_i, and returns an allocation rule $q_i : T_i \rightarrow [0, 1]$ and a verification rule $a_i : T_i \rightarrow [0, 1]$ for each agent $i \in I$.

Definition 5.1.2. A direct mechanism $\mathcal{M} = (T_1, \ldots, T_{|I|}, \{z_i(\cdot), a_i(\cdot)\}_{i \in I})$ is incentive compatible if each agent i has an incentive to truthfully report her type, i.e.

$$z_i(t_i) \geq z_i(t_i') - a_i(t_i') \quad \forall i \in I \quad \forall t_i, t_i' \in T_i. \tag{5.2}$$

The left-hand side of (5.2) is the probability of receiving the good with a truthful report. The right-hand side is the probability of receiving the good with a misreport adjusted downwards for the possibility of being inspected and punished for misreporting.

The principal would like to choose the allocation and verification probabilities z and a satisfying (5.1) and (5.2) to maximize:

$$\sum_{i \in I} \mathbb{E}_{t_{<i}}[\prod_{j<i}(1 - z_j(t_j))]\mathbb{E}_{t_i}[t_i z_i(t_i) - c a_i(t_i)].$$

5.1.1. Reduced-Form Representation

I work with a reduced-form representation of the allocation and verification rules (see for example C Border (1991); Vohra (2012); Li (2020)). Given an allocation and verification rule, (z, a), let $Q_i(t_i) = z_i(t_i)\mathbb{E}_{t_{<i}}[\prod_{j<i}(1 - z_j(t_j))]$ and $A_i(t_i) = a_i(t_i)\mathbb{E}_{t_{<i}}[\prod_{j<i}(1 - z_j(t_j))]$ be the interim allocation and verification probabilities respectively. The interim allocation and verification probabilities are related to the allocation and verification probabilities as follows:

Lemma 5.1.1. Let Q, A, z, a be the interim as well as actual allocation and verification rules of a direct mechanism. Then the interim and actual rules are related as follows:

$$z_i(t_i) = \frac{Q_i(t_i)}{1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} \tag{5.3}$$

$$a_i(t_i) = \frac{A_i(t_i)}{1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} \tag{5.4}$$

Proof. As shown in the proof of Lemma 4.4.1 for the stopping problem,

$$\mathbb{E}_{r \leq i}[\prod_{j \leq i}(1 - z_j(r_j))] = 1 - \sum_{j \leq i} \mathbb{E}_{r_j}[Q_j(r_j)].$$

It is now easy to relate the allocation and verification rules. By the definition of Q, A,

- $Q_i(t_i) = z_i(t_i)\mathbb{E}_{t_{<i}}[\prod_{j<i}(1 - z_j(t_j))] \Rightarrow z_i(t_i) = \frac{Q_i(t_i)}{1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]}$,
- $A_i(t_i) = a_i(t_i)\mathbb{E}_{t_{<i}}[\prod_{j<i}(1 - z_j(t_j))] \Rightarrow a_i(t_i) = \frac{A_i(t_i)}{1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]}$.

□

It follows from Lemma 5.1.1 that the set of constraints (5.1) can be reduced to

$$Q_i(t_i) + \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] \leq 1 \quad \forall i \in I \quad \forall t_i \in T_i$$

$$0 \leq A_i(t_i) \leq Q_i(t_i) \quad \forall i \in I \quad \forall t_i \in T_i$$

Using the reduced form I can formulate the principal's problem as the following linear program (denoted (VLP)):

$$\begin{aligned}
\max_{Q,A} \; & \sum_{i \in I} \mathbb{E}_{t_i}[t_i Q_i(t_i) - cA_i(t_i)] \\
s.t. \; & Q_i(t_i) + \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] \leq 1 \quad \forall i \quad \forall t_i \in T_i \\
& Q_i(t_i) \geq Q_i(t_i') - A_i(t_i') \quad \forall i \in I \quad \forall t_i, t_i' \in T_i \\
& 0 \leq A_i(t_i) \leq Q_i(t_i) \quad \forall i \in I \quad \forall t_i \in T_i
\end{aligned} \tag{VLP}$$

5.2. The Optimal Mechanism

In this section, I derive the optimal interim allocation and verification rules. The interim verification rule will be derived as a function of the optimal interim allocation rule. The optimal interim allocation rule will be given as a solution to a linear program. The actual allocation and verification rules can be obtained from the interim ones via Lemma 5.1.1.

Given the optimal interim allocation rule, the optimal interim verification rule can be deduced from the incentive constraints in (LP). They can be reduced to the following:

$$\min_{t_i} Q_i(t_i) \geq Q_i(t_i') - A_i(t_i') \quad \forall i \in I \quad \forall t_i' \in T_i \tag{5.5}$$

Therefore, at optimality,

$$A_i(t_i) = Q_i(t_i) - \min_{t_i'} Q_i(t_i'). \tag{5.6}$$

(5.6) is used to eliminate the verification variables from (LP). I also introduce a new set of variables $\{\phi_i\}_{i\in I}$ accounting for the minimum interim allocation per agent. For a given $\{\phi_i\}_{i\in I}$, the optimal interim allocation rule is given by the following linear program denoted LP(ϕ):

$$
\begin{aligned}
V(\phi) = \max_{Q} \sum_{i\in I} \mathbb{E}_{t_i}[(t_i - c)Q_i(t_i)] + c\phi_i \\
s.t.\ Q_i(t_i) + \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] \leq 1 \quad \forall i \quad \forall t_i \in T_i \\
Q_i(t_i) \geq \phi_i \geq 0 \quad \forall i \in I \quad \forall t_i \in T_i
\end{aligned}
\tag{5.7}
$$

Whenever $\sum\limits_i \phi_i \leq 1$, $V(\phi)$ is well defined, otherwise there is no feasible solution. This is because $1 \geq \sum_i \mathbb{E}_{t_i}[Q_i(t_i)] \geq \sum_i \phi_i$ should hold. Hence, the problem of finding the optimal mechanism reduces to

$$\max_{\phi:\sum\limits_{i\in I}\phi_i\leq 1} V(\phi),$$

which is also a linear program. I now characterize the optimal interim allocation and verification rules given ϕ.

Lemma 5.2.1. The optimal solution of LP(ϕ) is monotonic in type, i.e.

$$Q_i(t_i) \leq Q_i(t'_i) \quad \forall i \in I \quad \forall t_i \leq t'_i$$

Proof. Suppose not. Then, there is an i and pair (t_i, t'_i) such that $Q_i(t_i) > Q_i(t'_i)$. I pick an $\epsilon > 0$ such that

- $Q_i(t_i) - \frac{\epsilon}{f_i(t_i)} \geq Q_i(t'_i)$,
- $Q_i(t'_i) + \frac{\epsilon}{f_i(t'_i)} \leq Q_i(t_i)$.

If $Q_i(t_i)$ is reduced by $\frac{\epsilon}{f_i(t_i)}$ and $Q_i(t'_i)$ is increased by $\frac{\epsilon}{f_i(t'_i)}$, feasibility is preserved. The objective function value increases by $\epsilon(t'_i - t_i) > 0$, which is a contradiction. □

Hence, there exists a threshold $\hat{t}_i$ for all i such that $Q_i(t_i) = \phi_i$ for $t_i \leq \hat{t}_i$ and $Q_i(t_i) \geq \phi_i$ otherwise.

I show that the optimal strategy is a threshold strategy in each round. A transformation of variables will prove convenient:

$$Q_i(t_i) = \phi_i + x_i(t_i) \tag{5.8}$$

Given ϕ, the optimal strategy can be found by identifying the solution to the following linear program:

$$\begin{aligned} \max_x \ & \sum_{i \in I} \mathbb{E}_{t_i}[x_i(t_i)(t_i - c)] \\ s.t.\ & x_i(t_i) + \sum_{j<i} \mathbb{E}_{t_j}[x_j(t_j)] \leq 1 - \sum_{j \leq i} \phi_j \quad \forall i \in I \quad \forall t_i \in T_i \\ & x_i(t_i) \geq 0 \quad \forall i \in I \quad \forall t_i \in T_i \end{aligned} \tag{XP}$$

(XP) is a simplified version of LP(ϕ) given by the transformation defined in (5.8).

Lemma 5.2.2. Suppose that Q is the optimal solution to LP(ϕ). Then, for each agent i, there exists a threshold $\hat{t}_i$, such that

$$Q_i(t_i) = \begin{cases} 1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] & \text{if } t_i \geq \hat{t}_i \\ \phi_i & \text{otherwise} \end{cases} \tag{5.9}$$

Proof. Suppose we are interested in the allocation and verification rules when we reach agent i. Fix all other variables. We are interested in solving the following linear program

$$\begin{aligned}
&\max_{x_i} \mathbb{E}_{t_i}[x_i(t_i)(t_i - c)] \\
&s.t.\ x_i(t_i) \leq 1 - \sum_{j \leq i} \phi_j - \sum_{j<i} \mathbb{E}_{t_j}[x_j(t_j)] \quad \forall i \in I \quad \forall t_i \in T_i \\
&\quad \mathbb{E}_{t_i}[x_i(t_i)] \leq 1 - \sum_{j \leq k} \phi_j - x_k(t_k) - \sum_{j<k, j \neq i} \mathbb{E}_{t_j}[x_j(t_j)] \quad \forall k > i \quad \forall t_k \in T_k \\
&\quad x_i(t_i) \geq 0 \quad \forall i \in I \quad \forall t_i \in T_i
\end{aligned}$$

Now, it is clear that the optimal solution can actually be characterized by a threshold. All high types will be assigned their upper limit till the constraint on the aggregate allocation binds. Thus, the optimal solution x is given by

$$x_i(t_i) = \begin{cases} 1 - \sum\limits_{j \leq i} \phi_j - \sum\limits_{j<i} \mathbb{E}_{t_j}[x_j(t_j)] & \text{if } t_i \geq \hat{t}_i \\ 0 & \text{otherwise} \end{cases}$$

Returning back to Q variables completes the proof. □

Lemma 1 allows us to derive the actual allocation and verification rules given the interim ones. I also provide the form for the actual allocations, given the characterization of the optimal interim allocation in terms of parameters $\phi, \hat{t}$,

Corollary 5.2.3. For each agent i there exists a threshold $\hat{t}_i$ and constant α_i, such that the optimal actual allocation can be written as follows:

$$z_i(t_i) = \begin{cases} 1 & \text{if } t_i \geq \hat{t}_i \\ \alpha_i & \text{otherwise} \end{cases} \qquad a_i(t_i) = \begin{cases} 1 - \alpha_i & \text{if } t_i \geq \hat{t}_i \\ 0 & \text{otherwise} \end{cases}$$

Proof. Lemma 5.1.1 is used to derive the form of the actual allocation:

$$z_i(t_i) = \frac{Q_i(t_i)}{1 - \sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} = \begin{cases} \frac{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]}{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} & \text{if } t_i \geq \hat{t}_i \\ \frac{\phi_i}{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} & \text{otherwise} \end{cases} = \begin{cases} 1 & \text{if } t_i \geq \hat{t}_i \\ \alpha_i & \text{otherwise} \end{cases}$$

where $\alpha_i = \frac{\phi_i}{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]}$.

The form for the actual verification rule follows by (5.6). ☐

Before continuing, I summarize the roadmap for determining the optimal allocation and verification rules:

1. Solve the linear program $\max\limits_{\phi:\sum\limits_{i\in I}\phi_i \leq 1} V(\phi)$ to find the optimal interim allocation rule Q.

2. Derive the optimal interim verification rule A from equation (5.6).

3. Derive the optimal actual allocation and verification rules q, a from the interim ones Q, A, via Lemma 5.1.1.

5.2.1. Prophet Inequality

A prophet inequality is derived for the setting with verification using the reduced form.[1] It scales the optimal off-line solution so as to make it a feasible solution for the on-line setting. This technique can also be used in the standard setting.

Theorem 5.2.4. The optimal on-line algorithm achieves at least 1/2 of the performance of the optimal off-line algorithm in expectation.

Proof. Let $Q_i^*(t_i)$ be the interim expected probability with which agent i with type t_i receives the item in the optimal off-line solution. Let $\phi_i^* = \inf\limits_{t_i} Q_i^*(t_i)$ as proposed in Ben-Porath

[1]This result does not assume that the distribution of types is IID.

et al. (2014). The expected total value to the principal is given by

$$\sum_{i \in I} [\mathbb{E}_{t_i}[Q_i^*(t_i)(t_i - c)] + \phi_i^* c].$$

Pick on-line values $Q_i(t_i) = \frac{1}{2} Q_i^*(t_i)$ and $\phi_i = \frac{1}{2}\phi_i^*$. It is clear that the objective function with respect to the reduced form for both problems is linear and coincides. Thus, a simple scaling approximates the optimal objective:

$$\sum_{i \in I} [\mathbb{E}_{t_i}[Q_i(t_i)(t_i - c)] + \phi_i c] = \frac{1}{2} \sum_{i \in I} [\mathbb{E}_{t_i}[Q_i^*(t_i)(t_i - c)] + \phi_i^* c]$$

It suffices to prove that the proposed solution is feasible for the on-line problem.

- $Q_i(t_i) + \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] = \frac{1}{2} Q_i^*(t_i) + \frac{1}{2} \sum_{j<i} \mathbb{E}_{t_j}[Q_j^*(t_j)] \leq 1$: This holds since $Q_i^*(t_i) \leq 1$ and the expected off-line allocation for the first $i-1$ agents is also less than 1.

- $Q_i(t) \geq \phi_i$: The constraint coincides with the off-line constraint. Nothing changes by scaling both sides of the inequality.

□

5.3. Limited Punishment

The punishment is limited if the principal cannot reduce an agent's payoff to his outside option by punishing him. If we interpret verification as acquiring information, then punishment can be limited because information is imperfect.[2] I assume that punishment is proportional to the private benefit enjoyed by the agent from receiving the object. If v_i is the private benefit enjoyed by agent i, punishment is $k_i v_i$, where each $k_i \in [0, 1]$. These are the same assumptions as in Li (2020). As I show below, limited punishment will cause the principal to 'ration at the top' as well. All types above some threshold face the same

[2]Verification cost and punishment level are taken as exogenous, but it is possible that the principal can get more precise information by incurring a higher information acquisition cost, which, in turn, leads to a more severe expected punishment. The results in this work readily extend to the case where the principal can jointly optimize over verification cost and punishment level.

probability of receiving the good.

By the Revelation Principle, we can focus on direct mechanisms. In this case, if an agent is inspected, it is optimal to penalize him if and only if he is found to have lied. After the allocation is made, the planner will observe the agent's type and destroy a fraction k_i of the agent's payoff. A direct mechanism specifies for each profile of type reports the probability $z_i(t_i)$ that the good is assigned to agent i conditional on the event that it is not already assigned. These variables must satisfy the following feasibility conditions:

$$0 \leq z_i(t_i) \leq 1 \quad \forall i \in I \quad \forall t_i \in T_i \tag{5.10}$$

The incentive compatibility constraints are as follows:

$$v_i z_i(t_i) \geq (v_i - k_i v_i) z_i(t_i') \Rightarrow$$

$$z_i(t_i) \geq (1 - k_i) z_i(t_i') \quad \forall i \in I \quad \forall t_i, t_i' \in T_i \tag{5.11}$$

The principal would like to choose the allocation probabilities q to maximize:

$$\sum \mathbb{E}_{t_{<i}}[\prod_{j<i}(1 - z_j(t_j))\mathbb{E}_{t_i}[t_i z_i(t_i)].$$

As before I work with a reduced-form representation. This allows us to formulate the optimal mechanism as the following linear program :

$$\begin{aligned}
\max_Q \; & \sum_{i \in I} \mathbb{E}_{t_i}[t_i Q_i(t_i)] \\
s.t. \; & Q_i(t_i) + \sum_{j<i} \mathbb{E}_{t_i}[Q_i(t_i)] \leq 1 \quad \forall i \quad \forall t_i \in T_i \\
& Q_i(t_i) \geq (1 - k_i) Q_i(t_i') \quad \forall i \quad \forall t_i \in T_i \quad \forall t_i' \in T_i \\
& Q_i(t_i) \geq 0 \quad \forall i \quad \forall t_i \in T_i
\end{aligned}$$

5.3.1. The Optimal Mechanism

The incentive constraint is simplified, as in Mylovanov and Zapechelnyuk (2017). The proof is included for completeness.

Lemma 5.3.1. An allocation rule satisfies incentive compatibility if and only if for all i there exists χ_i such that

$$(1-k_i)\chi_i \leq Q_i(t_i) \leq \chi_i \quad \forall t_i \in T_i \tag{5.12}$$

Proof. If incentive compatibility holds then (5.12) holds with $\chi_i = \sup_{t_i} Q_i(t_i)$. Conversely, if (5.12) holds for some χ_i, then it also holds with $\chi_i' = \sup_{t_i} Q_i(t_i)$, which implies incentive compatibility. $\square$

I now write down a linear program which finds the optimal strategy. We know that for optimal χ this linear program is going to return the optimal strategy.

$$\begin{aligned}
&\max_{Q,\chi} \sum_{i\in I} \mathbb{E}_{t_i}[t_i Q_i(t_i)] \\
&s.t.\ Q_i(t_i) + \sum_{j<i} \mathbb{E}_{t_i}[Q_i(t_i)] \leq 1 \quad \forall i \in I \quad \forall t_i \in T_i \\
&\quad (1-k_i)\chi_i \leq Q_i(t_i) \leq \chi_i \quad \forall i \in I \quad \forall t_i \in T_i \\
&\quad Q_i(t_i) \geq 0 \quad \forall i \in I \quad \forall t_i \in T_i
\end{aligned}$$

The optimal strategy is now described.

Lemma 5.3.2. Suppose that Q is the optimal on-line solution. Let $\chi_i = \sup_{t_i \in T_i} Q_i(t_i)$. Then for each agent i, there exists a threshold $\hat{t}_i$ such that

$$Q_i(t_i) = \begin{cases} \chi_i & \text{if } t_i \geq \hat{t}_i \\ (1-k_i)\chi_i & \text{otherwise} \end{cases} \tag{5.13}$$

Proof. Suppose we are interested in the allocation rule when we reach agent i. Fix all other variables at their optimal value. We are interested in solving the following linear program:

$$\begin{aligned}
&\max_{Q_i} \mathbb{E}_{t_i}[t_i Q_i(t_i)] \\
&s.t.\ Q_i(t_i) \leq 1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] \quad \forall i \in I \quad \forall t_i \in T_i \\
&\qquad \mathbb{E}_{t_i}[Q_i(t_i)] \leq 1 - Q_k(t_k) - \sum_{j<k, j\neq i} \mathbb{E}_{t_j}[Q_j(t_j)] \quad \forall k > i \quad \forall t_k \in T_k \\
&\qquad (1-k_i)\chi_i \leq Q_i(t_i) \leq \chi_i \quad \forall t_i \in T_i \\
&\qquad Q_i(t_i) \geq 0 \quad \forall t_i \in T_i
\end{aligned}$$

Now, it is clear that the optimal solution can be characterized by a threshold policy. All high types will be assigned their upper limit till a constraint for the aggregate allocation binds. The optimal on-line solution has the following form:

$$Q_i(t_i) = \begin{cases} \min\{\chi_i, 1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]\} & \text{if } t_i \geq \hat{t}_i \\ (1-k_i)\chi_i & \text{otherwise} \end{cases} \tag{5.14}$$

The upper limit can be simplified. I prove that

$$\chi_i \leq 1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)] \quad \forall i \in I.$$

Suppose otherwise. Pick $\chi' = 1 - \sum_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]$. This makes the constraints less strict since the upper bound remains the same, but the lower bound reduces. Thus, the allocation for lower types can be reduced and the allocation of higher types can be increased while holding the aggregate allocation steady. This is a contradiction since such a change will increase total welfare. □

In the limited penalties case the actual allocation will have a slightly different form.

Corollary 5.3.3. For each agent i there exists a threshold $\hat{t}_i$, and constant β_i, such that the optimal actual allocation rule can be written as follows:

$$z_i(t_i) = \begin{cases} \beta_i & \text{if } t_i \geq \hat{t}_i \\ (1-k_i)\beta_i & \text{otherwise} \end{cases}$$

Proof. Lemma 5.1.1 is used to get the form of the actual allocation rule:

$$z_i(t_i) = \frac{Q_i(t_i)}{1 - \sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} = \begin{cases} \frac{\chi_i}{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} & \text{if } t_i \geq \hat{t}_i \\ \frac{(1-k_i)\chi_i}{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]} & \text{otherwise} \end{cases} = \begin{cases} \beta_i & \text{if } t_i \geq \hat{t}_i \\ (1-k_i)\beta_i & \text{otherwise} \end{cases}$$

where $\beta_i = \frac{\chi_i}{1-\sum\limits_{j<i} \mathbb{E}_{t_j}[Q_j(t_j)]}$. □

5.3.2. Prophet Inequality

The same machinery as before is used to further illustrate that extra constraints that restrict the optimal solution in both off-line and on-line cases, do not have an effect on the prophet inequality.

Theorem 5.3.4. The optimal on-line algorithm achieves at least 1/2 of the performance of the optimal off-line algorithm on expectation.

Proof. Let $Q_i^*(t_i)$ be the interim probability with which agent i with type t_i receives the item in the optimal off-line solution. Let $\chi_i^* = \sup\limits_{t_i \in T_i} Q_i^*(t_i)$ as proposed in Mylovanov and Zapechelnyuk (2017). The expected total value to the principal is given by

$$\sum_{i \in I} \mathbb{E}_{t_i}[t_i Q_i^*(t_i)]$$

Pick on-line values $Q_i(t_i) = \frac{1}{2} Q_i^*(t_i)$ and $\chi_i = \frac{1}{2}\chi_i^*$ for all $i \in I$. It is clear that the objective function with respect to the reduced form for both problems is linear and coincides. Thus, a simple scaling approximates the optimal objective:

$$\sum_{i\in I}\mathbb{E}_{t_i}[t_iQ_i(t_i)] = \frac{1}{2}\sum_{i\in I}\mathbb{E}_{t_i}[t_iQ_i^*(t_i)]$$

it suffices to prove that the proposed solution is feasible for the on-line problem.

- $Q_i(t_i) + \sum_{j<i}\mathbb{E}_{t_j}[Q_j(t_j)] = \frac{1}{2}Q_i^*(t_i) + \frac{1}{2}\sum_{j<i}\mathbb{E}_{t_j}[Q_j^*(t_j)] \leq 1$: This holds since $Q_i^*(t_i) \leq 1$ and the expected off-line allocation for the first $i-1$ agents is also less than 1.

- $(1-k_i)\chi_i \leq Q_i(t) \leq \chi_i$: The constraint coincides with the off-line constraint. Nothing changes by scaling both sides of the inequalities.

□

CHAPTER 6 : Prophet Inequalities with Complex Constraints

In this chapter, an extension of the basic stopping problem is introduced, which involves selecting a point in a polyhedron where the value of each coordinate must be chosen sequentially. The prescribed polyhedra can be described by a collection of inequalities, e.g., knapsack or polymatroid.

A version of this problem was proposed by Feldman et al. (2016). In that paper, there is one unit of a divisible resource that is interpreted as a service level. Each agent can be allocated a level of service in $[0, 1]$. For example, this could represent the duration an agent's advertisement is displayed. The draw r_i denotes the marginal value of the agent i for service. The goal is to allocate service levels so as to maximize the expected total reward enjoyed. A polyhedral selection problem, where the constraint set is characterized by a polymatroid, was examined by Dütting and Kleinberg (2015). A significant portion of the existing literature has shown interest in a combinatorial setting, where additional integrality restrictions are imposed on the coordinates. Such problems are called *on-line selection problems*.

In the next section, a general version of the *on-line polyhedral selection problem* is introduced. A few special cases of interest are described. I will emphasize the *on-line fractional knapsack selection problem* and provide a possible application in computational sprinting. Then, a reduced-form representation of it is proposed in order to derive a new prophet inequality. Unfortunately, the technique cannot be generalized to an arbitrary set of constraints. Last, I argue that the prophet inequality derived for on-line matroid selection problems in the integer and fractional cases might have an intriguing connection. It is shown that the linear programming formulations for the simple settings of a uniform and cardinal matroid carry through to the combinatorial setting. This establishes that the possible prophet inequalities to derive in each case are the same. Previous results in the literature observe ways to transform algorithms from one case to the other. To the best of my knowledge, this is the first formal relation.

6.1. On-line Polyhedral Selection

Let $q_i(r_{\leq i})$ be the level of service offered to agent $i \in I$ given the profile of rewards $r_{\leq i}$ was realized, where $I = \{1, \ldots, n\}$. These are called ex-post allocation variables. The agents arrive sequentially and the decision-maker decides at the spot for the level of service awarded to each agent. For all $i \in I$, let

$$\mathbf{q}(r) = [q_1(r_1), \ldots, q_i(r_{\leq i}), \ldots, q_n(r)].$$

To describe the set of feasible ex-post allocations let A be a non-negative $m \times n$ matrix and b a $m \times 1$ non-negative vector. Then, $\mathbf{q}$ is feasible if for all profiles of rewards r,

$$A\mathbf{q}(r) \leq \mathbf{b}.$$

Formally the best selection is given as a solution to (PSLP),

$$\begin{aligned} &\max \mathbb{E}_r[r_i q_i(r_{\leq i})] \\ &s.t. \sum_{i \in I} a_{ki} q_i(r_{\leq i}) \leq b_k \quad \forall k \quad \forall r \\ &\mathbf{q}(r) \geq 0 \quad \forall r \end{aligned} \tag{PSLP}$$

Now a linear programming formulation for the prophet's problem is given, i.e., the off-line version. For reasons that will become clear later, notation similar to the above will be used. In the prophet's problem, the level of service offered to agent i is based on the *entire* profile of rewards. Denote by $w_i(r)$, the ex-post level of service offered to agent i at reward profile r. Let $\mathbf{w}(r) = [w_1(r), \ldots, w_n(r)]$. Here r_{-i} denotes the profile $(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_n)$.

The prophet's problem can be expressed as follows:

$$\begin{aligned}
&\max \mathbb{E}_r[\sum_{i \in I} r_i w_i(r)] \\
&s.t. \sum_i a_{ki} w_i(r) \le b_k \quad \forall k \quad \forall r \\
&\mathbf{w} \ge 0 \quad \forall r
\end{aligned} \tag{PPSLP}$$

Denote the optimal solution to the prophet's problem by $\mathbf{w}^*$.

Consider the interim allocation variables $\{Q_i(r_i)\}_{i \in I, r_i \in R_i}$ defined as follows,

$$Q_i(r_i) = \mathbb{E}_{r_{<j}}[q_j(r_{\le j})].$$

A formulation of the on-line polyhedral selection problem in terms of the interim allocations is called the reduced-form representation. Similarly, the interim allocation variables are defined,

$$W_i(r_i) = \mathbb{E}_{r_{-i}}[w_i(r_i, r_{-i})].$$

An interim allocation Q is implementable if there exists an ex-post allocation $\mathbf{q}$ satisfying the set of constraints (IMP[Q]), defined below.

$$\begin{aligned}
&\sum_{i=1}^{n} a_{ki} q_i(r_{\le i}) \le b_k \quad \forall k \quad \forall r \\
&\mathbb{E}_{r_{<i}}[q_i(r_{\le i})] \ge Q_i(r_i) \quad \forall i \in I \quad \forall r_i \\
&\mathbf{q}(r) \ge 0 \quad \forall r
\end{aligned} \tag{IMP[Q]}$$

The objective function of the stopping problem in terms of an interim allocation is $\sum_{i \in I} \mathbb{E}_{r_i}[r_i Q_i(r_i)]$. Denote the optimal solution to the reduced-form representation of the on-line selection problem by Q^*. Denote the optimal solution to the reduced-form representation of the prophet's problem by W^*.

6.1.1. Normalization

In fact, the parameterization of the problem can be slightly changed. The LHS on each of the constraints can be scaled by the RHS, such that all new right hand sides are equal to one. The prophet's feasible polytope can now be redefined as follows,

$$\sum_i a_{ki} w_i(r) \leq 1 \quad \forall k \quad \forall r$$

$$w_i(r) \geq 0 \quad \forall r$$

Similarly, the implementation problem (IMP[Q]) can be rewritten as follows,

$$\sum_i a_{ki} q_i(r_{\leq i}) \leq 1 \quad \forall k \quad \forall r$$

$$\mathbb{E}_{r_{<i}}[q_i(r_{\leq i})] \geq Q_i(r_i) \quad \forall i \in I \quad \forall r_i$$

$$\mathbf{q} \geq 0 \quad \forall r$$

6.1.2. Examples

The *on-line polyhedral selection problem* describes a quite general setting with complex constraints. Several special cases are worth noting, with the hope that their special structure could lead to different types of results. I go through a few examples.

Example. The *basic stopping problem* described in Chapter 4 is a special case. A single constraint is applied to the variables for each profile. The relevant constraints will be

$$\sum_i q_i(r_{\leq i}) \leq 1 \quad \forall r. \tag{6.1}$$

The relevant constraints will be

$$\sum_i w_i(r) \leq 1 \quad \forall r. \tag{6.2}$$

Note that here, in comparison to the basic stopping problem, integrality constraints on the variables are not imposed. As we will see in the end of this chapter, this is without loss of generality, since the optimal solution will be integer. Intuitively, a decision-maker selects either the current reward or the future expected reward, i.e., fractional values are not reward-maximizing.

Example. The *on-line fractional knapsack selection problem*, first introduced by Feldman et al. (2016), relates to a knapsack with one unit of capacity and n items with weights $a_1, \ldots, a_n$ and one unit of supply. The knapsack constraint and items' supply constraints for the on-line problems follow:

$$\begin{aligned} &\sum_i a_i q_i(r_{\leq i}) \leq 1 \quad \forall r \\ &q_i(r_{\leq i}) \leq 1 \quad \forall i \quad \forall r_{\leq i}. \end{aligned} \tag{6.3}$$

Similarly, the prophet's problem can be summarized by a quite similar set of constraints,

$$\begin{aligned} &\sum_i a_i w_i(r) \leq 1 \quad \forall r \\ &w_i(r) \leq 1 \quad \forall i \quad \forall r. \end{aligned} \tag{6.4}$$

Note that this settings subsumes another interesting setting where each item has the same weight.

Example. The polymatroid setting involves polymatroid constraints. The relevant constraints for the on-line setting follow:

$$\sum_{i \in S} q_i(r_{\leq i}) \leq f(S) \quad \forall S \quad \forall r, \tag{6.5}$$

where f is a submodular function. The prophet's problem can be summarized by a similar set of constraints,

$$\sum_{i \in S} w_i(r) \leq f(S) \quad \forall S \quad \forall r, \tag{6.6}$$

This is the setting examined by Dütting and Kleinberg (2015).

The literature contains some stylized applications of these selection problems, such as spatially distributed markets or position auctions. Below, the first concrete application of the on-line fractional knapsack selection problem, that I am aware of, is given.

Computational Sprinting. Recent chip microarchitecture developments make it possible to expedite processes running in the processor. The technique is called sprinting, and it is a class of mechanisms that provides a short but significant performance boost while temporarily exceeding the thermal design point of the processors. Huang et al. (2019) propose a software runtime that manages sprints by dynamically predicting utility and modeling thermal headroom. The authors also compare their set of mechanisms experimentally against an "oracular policy", which matches the notion of a prophet in our setting.

Computational sprinting can be modeled as an on-line fractional knapsack selection problem. Each epoch consists of a large number of instructions and can be categorized accordingly. When the computation reaches an epoch, the mechanism can predict the gains from sprinting accurately. The mechanism has to decide the extent of increasing the temperature beyond the thermal design point in each epoch. Apart from the thermal headroom available in each period, there is a limit on the number of periods that the processor can perform over the thermal design point. The intertemporal limitation on heat increase is modeled as a knapsack constraint. An allocation for the fractional knapsack constraint could serve as a guide to pace sprints to maximize long-run performance under thermal constraints.

6.2. A Prophet Inequality for the On-line Fractional Knapsack Selection Problem

In this section, I will leverage algorithm $\mathcal{A}$ to derive a new prophet inequality for the on-line fractional knapsack selection problem. Algorithm $\mathcal{A}$ first computes the optimal prophet's interim allocations, scales them by a factor, and then constructs an on-line allocation that supports the scaled interim allocations. Let W^* be the prophet's interim allocation and α the scaling factor. It will be shown that this strategy using a factor $\alpha = \frac{1}{2}$ is implementable, providing a prophet inequality with an approximation guarantee of $\frac{1}{2}$. When the setting was first introduced, Feldman et al. (2016) derived a prophet inequality with an approximation guarantee of $\frac{1}{11.657}$. Later, Duetting et al. (2017) described a new algorithm that outputs an on-line allocation with precision ϵ and delivers a prophet inequality with an approximation guarantee of $\frac{1}{2+\epsilon}$.

Lemma 6.2.1. Scaling the prophet's optimal interim allocation W^* by a factor of $\frac{1}{2}$ is implementable.

Proof. The implementation problem can be stated as a linear feasibility problem as in (IMP[Q]). Let (FP(i)) be the implementation problem with respect to $\frac{1}{2}W_1^*, \dots, \frac{1}{2}W_i^*$. It is shown inductively that given feasibility of (FP(i)) a solution to FP(i+1) can be constructed. Formally, FP(i) is given by

$$\begin{aligned}
&\mathbb{E}_{r_{<j}}[q_j(r_{\le j})] \ge \frac{1}{2}W_j^*(r_j) \quad \forall r_j \quad \forall j \le i \\
&\sum_{j \le i} a_j q_j(r_{\le j}) \le 1 \quad \forall r_{\le i} \\
&0 \le q_j(r_{\le j}) \le 1 \quad \forall j \le i \quad \forall r_{\le j}
\end{aligned} \tag{FP(i)}$$

A function is defined which can be interpreted as the maximum available allocation for round $i+1$ after implementing the interim allocations $\frac{1}{2}W_1^*, \dots, \frac{1}{2}W_i^*$. The function will be parameterized by a constant α as a cap for the supply in round $i+1$. The parameterized function is given in (6.7).

$$\begin{aligned}
h_i(\alpha) = \max\ &\mathbb{E}_{r_{\le i}}[z(r_{\le i})] \\
s.t.\ &\mathbb{E}_{r_{<j}}[q_j(r_{\le j})] \ge \frac{1}{2}W_j^*(r_j) \quad \forall r_j \quad \forall j \le i \\
&z(r_{\le i}) \le \alpha \quad \forall r_{\le i} \\
&a_{i+1} z(r_{\le i}) + \sum_{j \le i} a_j q_j(r_{\le j}) \le 1 \quad \forall r_{\le i} \\
&0 \le q_j(r_{\le j}) \le 1 \quad \forall r_{\le j} \quad \forall j \le i \\
&z(r_{\le i}) \ge 0 \quad \forall r_{\le i}
\end{aligned} \tag{6.7}$$

Note that $h_i(1)$ produces a feasible solution for FP(i+1) if $h_i(1) \geq \frac{1}{2} W^*_{i+1}(r_{i+1})$ for all r_{i+1}.

Consider two separate cases for $a_{i+1} \geq 1$ and $a_{i+1} < 1$. In the first case, the unit supply constraint for period $i+1$ is redundant, since the knapsack constraint is more restrictive towards variables z. As a consequence, when it comes to the prophet's allocation $W^*_{i+1}(r_{i+1})$, it is bounded above by $\frac{1}{a_{i+1}}$. Furthermore, for the optimal solution in (6.7) for $\alpha = 1$, it follows that the variables z will be set such that they fill out the knapsack constraint, i.e., $z(r_{\leq i}) = \frac{1-\sum_{j\leq i} a_j q_j(r_{\leq j})}{a_{i+1}}$ for all profiles $r_{\leq i}$. It readily follows that $h(1)$ can be bounded from below,

$$
\begin{aligned}
h_i(1) = \mathbb{E}_{r_{\leq i}}\Big[\frac{1-\sum\limits_{j\leq i} a_j q_j(r_{\leq j})}{a_{i+1}}\Big] &= \frac{1-\sum\limits_{j\leq i} a_j \mathbb{E}_{r_{\leq j}}[q_j(r_{\leq j})]}{a_{i+1}} \\
&= \frac{1-\frac{1}{2}\sum\limits_{j\leq i} a_j \mathbb{E}_{r_j}[W^*_j(r_j)]}{a_{i+1}} \geq \frac{1}{2a_{i+1}}.
\end{aligned}
$$

Thus, it follows that $h_i(1) \geq \frac{1}{2a_{i+1}} \geq \frac{1}{2} W^*_{i+1}(r_{i+1})$ for all r_{i+1}.

On the other hand, when $a_{i+1} < 1$, the knapsack constraint allows for larger z which must be restricted by the unit supply. In this case, the prophet's allocation $W^*_{i+1}(r_{i+1})$ is bounded above by 1. Consider an optimal solution $(\hat{z}, \hat{\mathbf{q}})$ to (6.7) for $\alpha = \frac{1}{a_{i+1}}$. The supply constraint (parameterized by $\alpha = \frac{1}{a_{i+1}}$) is redundant since it is less restrictive than the knapsack constraint. As before, the optimal value can be bounded from below as follows,

$$
\begin{aligned}
h_i\Big(\frac{1}{a_{i+1}}\Big) = \mathbb{E}_{r_{\leq i}}[\hat{z}(r_{\leq i})] &= \mathbb{E}_{r_{\leq i}}\Big[\frac{1-\sum\limits_{j\leq i} a_j \hat{q}_j(r_{\leq j})}{a_{i+1}}\Big] \\
&= \frac{1-\sum\limits_{j\leq i} a_j \mathbb{E}_{r_{\leq j}}[\hat{q}_j(r_{\leq j})]}{a_{i+1}} = \frac{1-\frac{1}{2}\sum\limits_{j\leq i} a_j \mathbb{E}_{r_j}[W^*_j(r_j)]}{a_{i+1}} \geq \frac{1}{2a_{i+1}}.
\end{aligned}
$$

Set $\mathbf{q} = \hat{\mathbf{q}}$ and $z = a_{i+1}\hat{z}$. $(z, \mathbf{q})$ is a feasible solution to (6.7) for $\alpha = 1$, since $z(r_{\leq i}) = a_{i+1}\hat{z}(r_{\leq i}) \leq a_{i+1}\frac{1}{a_{i+1}} = 1$ for all profiles $r_{\leq i}$ and all other constraints trivially continue to hold. Furthermore, the objective function can be bounded below as follows,

$$
\mathbb{E}_{r_{\leq i}}[z(r_{\leq i})] = a_{i+1}\mathbb{E}_{r_{\leq i}}[\hat{z}(r_{\leq i})] \geq a_{i+1}\frac{1}{2a_{i+1}} = \frac{1}{2}.
$$

By feasibility of $(z, \mathbf{q})$, it follows that $h_i(1)$ is bounded from below by the solution's objective, i.e., $h_i(1) \geq \mathbb{E}_{r_{\leq i}}[z(r_{\leq i})] \geq \frac{1}{2}$. This implies that $h_i(1) \geq \frac{1}{2} \geq \frac{1}{2} W^*_{i+1}(r_{i+1})$ for all r_{i+1}, which completes the proof for this case too. $\square$

The prophet inequality for the fractional knapsack setting holds as a simple corollary of Lemma 6.2.1.

Theorem 6.2.2. Let $V_{\mathcal{A}}$ be the reward gained by $\mathcal{A}$ in the fractional knapsack selection setting. $V_{\mathcal{A}}$ accounts to half the reward gained by the prophet,

$$V_{\mathcal{A}} = \frac{1}{2} M.$$

The applicability of algorithm $\mathcal{A}$ in a more general setting, as well as that of other algorithms presented in Section 4.4, are left for future research. Furthermore, the task of understanding prophet inequalities in more general polyhedral selection problems is left for further research.

6.3. Continuous vs Combinatorial Settings

On-line polyhedral selection problems with integrality constraints are referred to in the literature as *on-line selection problems*. I revisit the basic stopping problem and cardinal matroid selection. I point out that the linear programming formulations of the optimal stopping rule and the prophet's problem in a continuous domain coincide with the ones for the combinatorial domain. As a consequence, the approximation factor driving the prophet inequalities in these two domains must coincide. It is of interest and left for future research to generalize the question for polymatroids, where the prophet's problem is guaranteed to follow the same pattern (Schrijver, 2003), i.e., the greedy algorithm optimizing a linear function over a polymatroid with integer values is optimal and always returns an integer solution.

Unit Supply

First, a linear program for the fractional case is defined without using the interim variables. The optimal on-line allocation for the fractional case is given by (LP).

$$\begin{aligned} \max \sum_i & \mathbb{E}_{r_i}[r_i \mathbb{E}_{r_1,\ldots,r_{i-1}}[q_i(r_{\leq i})]] \\ s.t. \sum_i & q_i(r_{\leq i}) \leq 1 \quad \forall r \\ & q_i(r_{\leq i}) \geq 0 \quad \forall i \quad \forall r_{\leq i} \end{aligned} \tag{LP}$$

There exists an optimal solution to (LP) which is integral. To prove the statement an integer optimal solution is constructed given an optimal solution q^*.

Theorem 6.3.1. There exists a solution $\bar{q}$ to (LP) which is integral, i.e., for all i and all profiles $r_{\leq i}$, $\bar{q}_i(r_{\leq i}) \in \{0, 1\}$.

Proof. A process that constructs an optimal intger solution given an optimal solution q^* will be provided.

Start with each r_1, step by step. Construct two solutions q^1, q^2 as follows: For all profiles starting with r_1:

1. $q_1^1(r_1) \coloneqq 1$ and $q_i^1(r_{\leq i}) \coloneqq 0$ for all $i \geq 2$ and $r_2, \ldots, r_i$,

2. $q_1^2(r_1) \coloneqq 0$ and $q_i^2(r_{\leq i}) \coloneqq \frac{1}{1-q_1^*(r_1)}$ for all $i \geq 2$ and $r_2, \ldots, r_i$.

q^1, q^2 are both feasible. By construction, it follows $q^* = q_1^*(r_1) \times q^1 + (1 - q^*(r_1)) \times q^2$. By linearity of expectations the above relation carries on to their values,

$$V^* = V(q^*) = q_1^*(r_1) \times V(q^1) + (1 - q^*(r_1)) \times V(q^2).$$

This completes the proof because either q^1 or q^2 with value V^*. The new solution will have a smaller number of non-integer values. By repeatedly applying the above process, an integer solution with value V^* is created. $\square$

The above theorem reduces the prophet inequality discovery to the one for the fractional problem. The optimal integer on-line solution is better than the optimal on-line fractional solution, which is better than $\frac{1}{2}$ of the optimal off-line fractional solution, which is better than the optimal off-line integer solution.

Multi-Unit Supply

I first define the problem for the fractional case without using the interim variables. The optimal on-line allocation for the fractional case is given by (MLP).

$$
\begin{aligned}
\max \sum_i & \mathbb{E}_{r_i}[r_i \mathbb{E}_{r_{<i}}[q_i(r_{\leq i})]] \\
s.t. \sum_i & q_i(r_{\leq i}) \leq k \quad \forall r \\
0 \leq & q_i(r_{\leq i}) \leq 1 \quad \forall i \in I \quad \forall r_{\leq i}
\end{aligned}
\tag{MLP}
$$

There exists an optimal solution to (MLP) which is integral. In order to prove this, one will be constructed given an optimal solution q^*.

Theorem 6.3.2. There exists a solution $\bar{q}$ to (MLP) which is integral, i.e. for all i and all profiles $r_{\leq i}$, $\bar{q}_i(r_{\leq i}) \in \{0, 1\}$.

Proof. The above procedure is slightly rearranged in order to construct q^1, q^2. Fix r_1 such that $q_1(r_1) \in (0, 1)$. For all profiles r starting with r_1 there exists minimum index $i(r)$ such that $q_{i(r)}(r_{\leq i(r)}) \in (0, 1)$, because in a non-generic optimal solution all cardinality constraints are binding (otherwise for a profile that it is not true, moving backwards we could find a variable to raise). Set

$$\epsilon := \min\{q_1(r_1), 1 - q_1(r_1), \min_{r_2,\ldots,r_n}\{q_{i(r)}(r_{\leq i(r)})\}, \min_{r_2,\ldots,r_n}\{1 - q_{i(r)}(r_{\leq i(r)})\}\}.$$

For all profiles starting with r_1:

1. $q^1_1(r_1) \coloneqq q^*_1(r_1) + \epsilon$ and for all r starting with r_1, $q^1_{i(r)}(r_{\leq i(r)}) = q^*_{i(r)}(r_{\leq i(r)}) - \epsilon$,

2. $q^2_1(r_1) \coloneqq q^*_1(r_1) - \epsilon$ and for all r starting with r_1, $q^2_{i(r)}(r_{\leq i(r)}) = q^*_{i(r)}(r_{\leq i(r)}) + \epsilon$.

q^1, q^2 are both feasible. By construction, it follows that $q^* = \frac{1}{2}q^1 + \frac{1}{2}q^2$. By linearity of expectations the above relation carries on on their values,

$$V^* = V(q^*) = \frac{1}{2}V(q^1) + \frac{1}{2}V(q^2).$$

Since q^* is optimal q^1 and q^2 must also be optimal. One of them, say for index $k \in \{1, 2\}$, will have at least one less non-integer value. Set $q^* \coloneqq q^k$ and repeat. We end up with an integer solution with value V^*. □

The above integrality property has a straightforward consequence for the relation of prophet inequalities in the continuous and combinatorial domains. It implies that the best approximation factor achieved is the same for both domains. It is an open question whether the integrality property holds for more general sets of constraints, like in the case of polymatroid constraints. From an algorithmic perspective, such a characterization might be valuable for devising an optimal and tractable selection rule for the matroid selection problem, whose existence remains an open question. In more detail, it is of interest to understand whether optimal interim allocations can be computed for the case of polymatroid constraints and whether a tractable selection rule can be devised from them.

BIBLIOGRAPHY

M. Abolhassani, S. Ehsani, H. Esfandiari, M. HajiAghayi, R. Kleinberg, and B. Lucier. Beating 1-1/e for ordered prophets. In *49th ACM Symposium on Theory of Computing*, page 61–71, 2017.

R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, 1993.

S. Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. In *52nd IEEE Symposium on Foundations of Computer Science*, page 512–521, 2011.

L. M. Ausubel. An efficient dynamic auction for heterogeneous commodities. *American Economic Review*, 96(3):602–629, 2006.

I. Ball and D. Kattwinkel. Probabilistic verification in mechanism design. In *20th ACM Conference on Economics and Computation*, pages 389–390, 2019.

T. A. Barthold. Issues in the design of environmental excise taxes. *Journal of Economic Perspectives*, 8(1):133–151, 1994.

E. Ben-Porath, E. Dekel, and B. L. Lipman. Optimal allocation with costly verification. *American Economic Review*, 104(12):3779–3813, 2014.

Y. Blum, A. E. Roth, and U. G. Rothblum. Vacancy chains and equilibration in senior-level labor markets. *Journal of Economic Theory*, 76(2):362–411, 1997.

K. C Border. Implementation of reduced form auctions: A geometric approach. *Econometrica*, 59:1175–87, 1991.

I. Caragiannis, E. Elkind, M. Szegedy, and L. Yu. Mechanism design: From partial to probabilistic verification. In *13th ACM Conference on Electronic Commerce*, pages 266–283, 2012.

S. Chawla, J. D. Hartline, D. L. Malec, and B. Sivan. Multi-parameter mechanism design and sequential posted pricing. In *42nd ACM Symposium on Theory of Computing*, page 311–320, 2010.

V. Danilov, G. Koshevoy, and K. Murota. Discrete convexity and equilibria in economies with indivisible goods and money. *Mathematical Social Sciences*, 41(3):251–273, 2001.

V. Danilov, G. A. Koshevoy, and C. Lang. Gross substitution, discrete convexity, and submodularity. *Discrete Applied Mathematics*, 131(2):283–298, 2003.

M. Davis and I. Karatzas. A deterministic approach to optimal stopping. *Probability, Statistics and Optimization*, pages 455–466, 1994.

P. Duetting, M. Feldman, T. Kesselheim, and B. Lucier. Prophet inequalities made easy: Stochastic optimization by pricing non-stochastic inputs. In *58th IEEE Symposium on Foundations of Computer Science*, pages 540–551, 2017.

P. Dütting and R. Kleinberg. Polymatroid prophet inequalities. In *Algorithms - ESA 2015*, pages 437–449, 2015.

M. Feldman, O. Svensson, and R. Zenklusen. Online contention resolution schemes. In *27th ACM-SIAM Symposium on Discrete Algorithms*, pages 1014–1033, 2016.

T. Fleiner, Z. Jankó, A. Tamura, and A. Teytelboym. Trading networks with bilateral contracts. *Working Paper*, 2018.

S. Fujishige and Z. Yang. A note on Kelso and Crawford's gross substitutes condition. *Mathematics of Operations Research*, 28(3):463–469, 2003.

D. Gale and M. Hellwig. Incentive-compatible debt contracts: The one-period problem. *The Review of Economic Studies*, 52(4):647–663, 1985.

A. Gershkov and B. Moldovanu. *Dynamic Allocation and Pricing: A Mechanism Design Approach*. MIT Press, 2014.

F. Granot and A. F. Veinott. Substitutes, complements and ripples in network flows. *Mathematics of Operations Research*, 10(3):471–497, 1985.

S. Guha and K. Munagala. Approximation algorithms for budgeted learning problems. In *39th ACM Symposium on Theory of Computing*, page 104–113, 2007.

F. Gul and E. Stacchetti. Walrasian equilibrium with gross substitutes. *Journal of Economic Theory*, 87(1):95–124, 1999.

F. Gul and E. Stacchetti. The English auction with differentiated commodities. *Journal of Economic Theory*, 92(1):66–95, 2000.

J. D. Hartline. Approximation in mechanism design. *American Economic Review*, 102(3): 330–36, 2012.

J. W. Hatfield and P. Milgrom. Matching with contracts. *American Economic Review*, 95 (4):913–935, 2005.

J. W. Hatfield, S. D. Kominers, A. Nichifor, M. Ostrovsky, and A. Westkamp. Stability and competitive equilibrium in trading networks. *Journal of Political Economy*, 121(5): 966–1005, 2013.

J. W. Hatfield, S. D. Kominers, A. Nichifor, M. Ostrovsky, and A. Westkamp. Full substitutability. *Theoretical Economics*, 14(4):1535–1590, 2019a.

J. W. Hatfield, S. D. Kominers, A. Nichifor, M. Ostrovsky, and A. Westkamp. Chain stability in trading networks. *Working Paper*, 2019b.

T. Hill and R. Kertz. A survey of prophet inequalities in optimal stopping theory. 1992.

T. P. Hill and R. P. Kertz. Comparisons of stop rule and supremum expectations of i.i.d. random variables. *The Annals of Probability*, 10(2):336–345, 1982.

Z. Huang, J. A. Joao, A. Rico, A. D. Hilton, and B. C. Lee. Dynasprint: Microarchitectural sprints with dynamic utility and thermal management. In *52nd IEEE/ACM International Symposium on Microarchitecture*, page 426–439, 2019.

Y. T. Ikebe and A. Tamura. Stability in supply chain networks: An approach by discrete convex analysis. *Journal of the Operations Research Society of Japan*, 58(3):271–290, 2015.

Y. T. Ikebe, Y. Sekiguchi, A. Shioura, and A. Tamura. Stability and competitive equilibria in multi-unit trading networks with discrete concave utility functions. *Japan Journal of Industrial and Applied Mathematics*, 32(2):373–410, 2015.

S. Iwata, S. Moriguchi, and K. Murota. A capacity scaling algorithm for M-convex submodular flow. *Mathematical Programming*, 103(1):181–202, 2005.

A. S. Kelso and V. Crawford. Job matching, coalition formation, and gross substitutes. *Econometrica*, 50(6):1483–1504, 1982.

M. King, B. Tarbush, and A. Teytelboym. Targeted carbon tax reforms. *European Economic Review*, 119:526 – 547, 2019.

R. Kleinberg and S. M. Weinberg. Matroid prophet inequalities and applications to multi-dimensional mechanism design. *Games and Economic Behavior*, 113(C):97–115, 2019.

U. Krengel and L. Sucheston. Semiamarts and finite values. *Bulletin of the American Mathematical Society*, 83(4):745–747, 1977.

R. P. Leme. Gross substitutability: An algorithmic survey. *Games and Economic Behavior*, 106:294–316, 2017.

Y. Li. Mechanism design with costly verification and limited punishments. *Journal of Economic Theory*, 186, 2020.

B. Lucier. An economic view of prophet inequalities. *ACM SIGecom Exchanges*, 16(1): 24–47, 2017.

D. Mookherjee and I. Png. Optimal Auditing, Insurance, and Redistribution. *The Quarterly Journal of Economics*, 104(2):399–415, 1989.

S. Moriguchi and K. Murota. Capacity scaling algorithm for scalable M-convex submodular flow problems. *Optimization Methods and Software*, 18(2):207–218, 2003.

K. Murota. Submodular flow problem with a nonseparable cost function. *Combinatorica*, 19(1):87–109, 1999.

K. Murota. *Discrete Convex Analysis: Monographs on Discrete Mathematics and Applications 10*. Society for Industrial and Applied Mathematics, 2003.

K. Murota. Discrete convex analysis: A tool for economics and game theory. *Journal of Mechanism and Institution Design*, 1(1):151–273, 2016.

K. Murota and A. Tamura. New characterizations of M-convex functions and their applications to economic equilibrium models with indivisibilities. *Discrete Applied Mathematics*, 131(2):495–512, 2003a.

K. Murota and A. Tamura. Application of M-convex submodular flow problem to mathematical economics. *Japan Journal of Industrial and Applied Mathematics*, 20(3):257–277, 2003b.

T. Mylovanov and A. Zapechelnyuk. Optimal allocation with ex post verification and limited penalties. *American Economic Review*, 107(9):2666–94, 2017.

M. Ostrovsky. Stability in supply chain networks. *American Economic Review*, 98(3): 897–923, 2008.

E. Samuel-Cahn. Comparison of threshold stop rules and maximum for independent nonnegative random variables. *The Annals of Probability*, 12(4):1213–1216, 1984.

A. Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer Science & Business Media, 2003.

L. S. Shapley and M. Shubik. The assignment game I: The core. *International Journal of Game Theory*, 1(1):111–130, 1971.

A. Shioura. Fast scaling algorithms for M-convex function minimization with application to the resource allocation problem. *Discrete Applied Mathematics*, 134(1–3):303–316, 2004.

A. Shioura and A. Tamura. Gross substitutes condition and discrete concavity for multi-unit valuations: A survey. *Journal of the Operations Research Society of Japan*, 58(1): 61–103, 2015.

N. Sun and Z. Yang. Equilibria and indivisibilities: Gross substitutes and complements. *Econometrica*, 74(5):1385–1402, 2006.

N. Sun and Z. Yang. A double-track adjustment process for discrete markets with substitutes and complements. *Econometrica*, 77(3):933–952, 2009.

N. Sun and Z. Yang. An efficient and incentive compatible dynamic auction for multiple complements. *Journal of Political Economy*, 122(2):422–466, 2014.

R. M. Townsend. Optimal contracts and competitive markets with costly state verification. *Journal of Economic Theory*, 21(2):265 – 293, 1979.

R. V. Vohra. Dynamic mechanism design. *Surveys in Operations Research and Management Science*, 17(1):60–68, 2012.

J. Wasserman, W. G. Manning, J. P. Newhouse, and J. D. Winkler. The effects of excise taxes and regulations on cigarette smoking. *Journal of Health Economics*, 10(1):43–64, 1991.

Q. Yan. Mechanism design via correlation gap. In *22nd ACM-SIAM Symposium on Discrete Algorithms*, page 710–719, 2011.