Generalized Top Trading Cycles: An Iterative Approach for Exchange Economies With Money

Acknowledgements

I would like to express my deep thanks to Prof. David Parkes for his patience and good humor in advising me. The opportunity to work with and learn from him has been a unique privilege, and I would not have reached this point without his guidance to keep me on track.

I owe a great deal to soon-to-be Prof. Nisarg Shah for encouraging me, pointing me to useful resources, spending time with me at the whiteboards, and looking over my work. I wish him a successful start to his professorship.

Thank you to Prof. Michael Mitzenmacher and Prof. Yaron Singer for generously agreeing to act as readers for this thesis. I would also like to thank Prof. Singer for arming me, through AM221, with the tools of optimization and a (perhaps unjustified) mathematical fearlessness.

Thank you to Emma Rose Kantor for preventing me from puddling and for reminding me that the Earth is unlikely to stop rotating. Thank you to Gabbi Merz for her invaluable proofreading and her steady hands at the helm of CS51, and to Tim McNamara for burning the midnight oil with me. Finally, thank you to my parents, whose support made this possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Imagine that you own a car and decide to sell it. How would you start? Perhaps you would put a sign in the car window and wait for a buyer to call. You might even visit a dealership to haggle over a fair price. Online marketplaces have replicated both of these phenomena: Craigslist allows buyers and sellers to contact each other directly, while CarMax, an online dealer, buys and sells cars through its website and claims to offer "one fair price."

A market for used cars is an example of a *discrete exchange economy with money*. In a discrete exchange economy with money, there are many individual agents, each of whom owns at most one indivisible item. Each agent has *unit demand*, meaning that she has use for at most one item. Some agents have no item and wish to buy one. Other agents own an item and wish to sell for cash. A third class of agents might wish both to sell and to buy, exchanging their current item for a different one and making or receiving a payment at the same time. In the used car example, the third type of agent wishes to trade in her old car at the dealership while she simultaneously buys a new one. No online marketplace has replicated this swap-and-pay phenomenon: we simply negotiate a trade at the dealership or make two separate transactions on Craigslist. These options are imperfect at best.

A *matching market* would offer a better solution. In the used car example, each agent would tell the market what car, if any, she owned, how much she values the other agents' cars, and how much she values her own car. The market *mechanism* would then identify the trades that make the agents collectively as happy as possible, optimizing *welfare*. It would also determine how much each agent should pay.

However, agents will always purse the goal of making themselves as well off as possible. An ideal mechanism must therefore address the strategic nature of agents and also guarantee that running the mechanism will be rational for its operator. More precisely, an ideal mechanism would have the properties of

- **Budget-balance**, whose weak form says that the mechanism cannot distribute more money than it collects (i.e. it cannot lose money), and whose strong form says that the mechanism must distribute exactly the total amount of money that it collects from agents,

- **Individual rationality**, which states that no agent will ever be made worse off than it began,
- **Strategy-proofness**, which describes mechanisms in which agents reveal their true preferences and which offer no benefit to misrepresentation, and
- **Efficiency**, which refers to the selection of allocations that optimize welfare, i.e. maximize the sum of valuations, of market participants.

The Myerson-Satterthwaite impossibility, famous in mechanism design, precludes the existence of a mechanism that satisfies all four of these properties at once [27]. In pursuing an effective, if necessarily imperfect, design, I address the impossibility in two ways. First, I attempt to relax the requirement of efficiency to *approximate efficiency* and show that this does nothing to broaden the design space. I then consider two previous approaches that relax efficiency and budget-balance, respectively, and conclude that they, too, solve the mechanism design problem inadequately. This motivates the relaxation of strategy-proofness and leads to my proposal of NetAscendingClock, a iterative mechanism that is budget-balanced, individually rational, exactly efficient, and fair.

1.1 Related Work

The approaches I present in this thesis build on techniques from several related bodies of the mechanism design literature and the literature on auction theory.

The discrete exchange economy with money is a special case of a two-sided combinatorial exchange. Shapley and Shubik initially characterized the two-sided setting with unit-demand and proved the existence of equilibrium prices [38]. Related double auction mechanisms, particularly with single-minded bidders, have also been well-studied; a double auction is a special case of the discrete exchange economy with money. Previous work has prioritized budget-balance and strategy-proofness over exact efficiency, ranging with McAfee's canonical trade-reduction static double auction to dynamic double auctions to double auctions with package-demanding bidders [23, 18, 12, 11].

Economies of agents with initial endowments and unit demand but without money are known as Shapley-Scarf economies; the standard mechanism for such economies is Gale's top trading cycles (TTC) mechanism [37]. The Shapley-Scarf model has also been extended to include some agents who are not initially endowed [2, 3], and some previous work has generalized Shapley-Scarf economies to include money but produced restrictive results. For example, Quinzii (1984) and Wako (1991) worked with budget-constrained agents, so that the supply of money in the economy was fixed. They showed that the core, i.e. the allocations selected by TTC, coincided with equilibrium allocations. Their models require the strong assumption that agent budgets are known to the mechanism [33, 41].

Two papers have worked on the same setting as I do. Miyagawa (2001) required strategy-proofness, budget-balance, individual rationality and a weak notion of group non-manipulability. He showed the negative result that TTC was the only mechanism with these properties [26]. More recently, Andersson et al. (2016) constructed a mechanism that is a strategy-proof, individually rational and group non-manipulable. However, they showed these properties over a restricted preference domain [7].

The novel mechanism in this thesis, NetAscendingClock, requires relaxing strategy-proofness to achieve efficiency but provides intuition about agent behavior by taking an iterative approach. Parkes and Ungar (2000) made similar strategic assumptions in the design of an iterative combinatorial auction, and Parkes et al. (2001) took a similar approach in the general combinatorial exchange setting [30]. I discuss NetAscendingClock's strategic properties through the framework of *approximate strategy-proofness* by showing that the mechanism meets the conditions, as described by Parkes and Lubin (2012), for a *price taking* assumption in which agents respond truthfully when presented with prices [31]. Nisan et. al (2011) provide a more detailed analysis of best response auctions for single-item and multiple-item two-sided settings, noting that these designs are frequently used in practice and connect to classical Vickrey-Clarke-Groves mechanisms in those settings [29, 40, 13, 19].

NetAscendingClock's iterative approach is closely related to ascending clock auctions for multiple items, originating with the work of Demange, Gale and Sotomayor (1986) [14]. I connect the computational properties of NetAscendingClock to recent work on the computational properties of two-sided clock mechanisms by Andersson, Andersson and Talman (2013), whose results unify previous work [5, 24, 35, 25]. I show that NetAscendingClock achieves its game theoretical and algorithmic properties by connecting its approach to previous mechanisms for the related problem of room-assignment rent-division. In room-assignment rent-division, a group of unit-demanding agents decide both how to split a set of indivisible items and how to share a fixed cost; the typical application is to assigning rooms in a multi-bedroom apartment and sharing the rent. In particular, I relate NetAscendingClock to the work of Abdulkadiroğlu et al (2004) and to the work of Gal et al. (2016) [1, 15]. Andersson, Ehlers, and Svensson (2013) consider the manipulability of allocation rules for room-assignment rent-division, in which mechanisms are typically not strategy-proof [6].

The impossibility result I present in Chapter 3 is closely related to previous impossibilities in general settings, such as Gibbard-Satterthwaite [17, 36], and Myerson-Satterthwaite [27]. For a general survey on matching see Niederle et al [28].

1.2 Contribution

My first contribution is to show that even with a relaxation to approximate efficiency, strategy-proof, budget-balanced and individually rational mechanisms do not exist. Confronted with this impossibility, I evaluate in sequence the viability of sacrificing each one of the desirable properties.

I ultimately claim that strategy-proofness should be relaxed in favor of budget-balance, individual rationality, and exact efficiency and propose NetAscendingClock as a solution with these properties.

More specifically, my contributions are that:

- I prove that budget-balance, individual rationality, strategy-proofness, and the weaker notation of approximate efficiency are mutually incompatible in the discrete exchange economy with money when two or more agents participate.
- To illustrate that existing mechanisms are undesirable, I show that the Vickrey-Clarke-Groves mechanism violates budget-balance and note that even with exogenous information about agent valuations, TTC provides inadequate welfare guarantees.
- I demonstrate that NetAscendingClock is budget-balanced, individually rational, and efficient relative to agents' revealed preferences. I also show that the mechanism satisfies envy-freeness, a fairness property. I show that the iterative mechanism is computationally tractable and that it can be applied in a direct form, as well.
- I show through simulation that TTC is inefficient relative to VCG and NetAscendingClock, that VCG's budget deficit is persistent even as markets grow large, and that NetAscendingClock and VCG achieve efficiency by facilitating longer cycles of trades than top trading cycles, a property that could, in an application, be either an asset or a liability.

In Chapter 2, I present the mathematical model, formally define the required game theoretical properties, and define the computational problem. In Chapter 3, I build the foundation of the argument in support of dropping strategy-proofness, showing the impossibility result and discussing the shortcomings of VCG and TTC. In Chapter 4, I introduce NetAscendingClock, prove its iterative approach converges, and show that it satisfies budget-balance, individual rationality, and exact efficiency and does so fairly. In Chapter 5, I describe simulations comparing NetAscendingClock with VCG and TTC and present their results. Chapter 6 concludes and considers directions for future work.

Chapter 2

Model and Preliminaries

In this chapter, I present the model for the discrete exchange economy with money. I then define several game theoretical and computational properties that will provide the framework for evaluating mechanisms.

2.1 The Model

A group of indivisible items, each of which is initially owned by one agent, is to be re-allocated among their owners. There is also a perfectly divisible good, "money." Agents have unit demand, such that they have no use for more than one item at a time. Formally, a **discrete exchange problem** is a three-tuple $\langle A, G, v\rangle$ where

1. $A = 1 \ldots n$ is a set of agents.
2. $G = 1 \ldots n$ is a set of items.[1]
3. $v = [v_i(j)]_{i \in A, j \in G} \in \mathbb{R}^{n \times n}_{\geq 0}$ is a valuation matrix where $v_i(j)$ denotes the value of item j for agent i. The valuations v_i of agent i are referred to as both a vector and a function.

Agent $i \in A$ is the initial owner of good $i \in G$. Reported valuations are denoted by $\hat{v}$, since agents may make misreports. I also refer to v_{-i} ($\hat{v}_{-i}$) as the (resp. reported) valuations of all agents except agent i. The assumption that agent valuations (and reports) are non-negative is without loss of generality.

For notational convenience, I define the function $OPT : \mathcal{P}(A) \times \mathcal{P}(G) \to \mathbb{R}$ to be the optimal achievable welfare when assigning a subset of agents A' to a subset of goods G', where $\mathcal{P}(\cdot)$ is the powerset.

[1]The assumption that $|A| = |G|$ is without loss of generality. Settings where $|A| > |G|$ can be trivially converted to $|A| = |G|$ by giving each unendowed agent a worthless dummy item. This observation allows some agents to be strictly sellers while other agents are strictly buyers, thereby greatly broadening the usefulness of the model.

The setting where $|G| > |A|$ in combination with initial ownership violates the unit-demand assumption. This model is a special case of the general combinatorial exchange, which addresses the $|G| > |A|$ case.

A **assignment** μ is a matching of items to agents such that each agent is assigned to exactly one item. The item assigned to agent i under μ is then μ_i. Call $\mathcal{M}$ the set of all possible assignments. Monetary **transfers** made by or to agents are denoted by $z \in \mathbb{R}^n$, such that agent i makes or receives transfers z_i. When $z_i > 0$, money is paid by the agent to the operator. When $z_i < 0$, money is paid by the operator to the agent. When **prices** on items are well-defined, they are denoted by $p \in \mathbb{R}^n$. An **allocation** is a pair $(\mu, z) \in \mathcal{M} \times \mathbb{R}^n$, or a pair $(\mu, p) \in \mathcal{M} \times \mathbb{R}^n$ when prices are well-defined. When an allocation is provided with prices, it is implied that each payment is $z_i = p_{\mu_i} - p_i$, the price difference between the agent's assigned and initial items.

Each agent $i \in A$ has a utility function $u_i : G \times \mathbb{R} \to \mathbb{R}$ that is **quasi-linear** in the agent's payment:

$$u_i(g, z_i) = v_i(g) - v_i(i) - z_i$$

I also refer to $u_i(g, p)$ as the utility function with respect to prices.

The **demand set** of an agent i is the set of items

$$D_i(p) = \{g : g = \operatorname*{argmax}_{g' \in G} u_i(g', p)\}$$

where each item in the set maximizes the agent's utility given the prices. Demand sets lead to a descriptive equilibrium concept for an allocation: Walrasian equilibrium.

Definition 2.1. *An allocation (μ, p) is a **Walrasian equilibrium** if for every agent i, $\mu_i \in D_i(p)$, and μ is a valid assignment.*

A **mechanism** is fully described by a pair of functions on agents' reported valuations: a choice rule and a payment rule.

Definition 2.2. *A **choice rule** $x : \mathbb{R}^{n \times n} \to \mathcal{M}$ selects an assignment given reported valuations.*

Definition 2.3. *A **payment rule** $t : \mathbb{R}^{n \times n} \to \mathbb{R}^n$ selects a payment vector given reported valuations.*

While a mechanism is a pair of functions (x, t), I also refer to a mechanism as single function $f : \mathbb{R}^{n \times n} \to \mathcal{M} \times \mathbb{R}^n$ from reports to allocations. I will denote by $u_i(f(\hat{v}))$ the utility to agent i in the allocation chosen by f given reports $\hat{v}$.

2.2 Game Theoretical Properties

I now present the game theoretical properties that will govern mechanisms in the discrete exchange economy with money. I will present definitions for six properties, in sequence: efficiency, c-approximate efficiency (c-efficiency), envy-freeness, strategy-proofness, individual rationality, and budget-balance. With the exception of c-approximate efficiency, each of these properties is standard.

Definition 2.4. *A assignment* μ *is* ***efficient*** *with respect to reported valuations if*

$$\mu = \operatorname*{argmax}_{\mu' \in \mathcal{M}} \sum_{i \in A} v_i(\mu'_i) \tag{2.1}$$

A mechanism is efficient if for any report $\hat{v} \in \mathbb{R}^n$*, the assignment* $x(\hat{v})$ *is efficient.*

Note that a mechanism can only be efficient with respect to reports; if agents do not report true valuations, a selected assignment may not be efficient. Efficiency is one metric that I will use to assess the performance of mechanisms both in theory and in simulation. c-Approximate efficiency, or simply c-efficiency, relaxes the efficiency requirement and allows for the exploration of a broader class of mechanisms.

Definition 2.5. *An assignment* μ *is* ***c-approximately efficient****, or c-efficient, if for any* $\hat{v}$ *and* $c \in (0, 1]$,

$$\sum_{i \in A} \hat{v}_i(\mu_i) \geq c \operatorname*{argmax}_{\mu' \in \mathcal{M}} \sum_{i \in A} \hat{v}_i(\mu'_i)$$

I will at points argue that fairness between agents is a useful design consideration. The basic metric of fairness I will use is envy-freeness.

Definition 2.6. *An allocation* (μ, z) *is* ***envy-free*** *if for each agent* $i \in A$

$$u_i(\mu_i, z_i) \geq u_i(\mu_j, z_j) \, \forall j \in A \tag{2.2}$$

In an envy-free allocation, every agent weakly prefers its own allocation to the allocation of any other agent. In this setting, envy-free allocations are exactly efficient:

Proposition 2.1. *Let* $(\mu, p) \in \mu \times \mathbb{R}^n$ *be an envy-free allocation. Then* (μ, p) *is efficient.*

Proof. Let $(\mu, p) \in \mathcal{M} \times \mathbb{R}^n_+$ be an envy-free allocation, and consider another allocation (η, q). Then:

$$u_i(\mu_i, p_{\mu_i}) \geq u_i(g, p_g), \ \forall i \in A, \ g \in G \qquad \text{(Envy-freeness)}$$
$$\implies \sum_{i \in A} v_i(\mu_i) = \sum_{i \in A} u_i(\mu_i, p_{\mu_i}) \geq \sum_{i \in A} u_i(\eta_i, p_{\eta_i}) = \sum_{i \in A} v_i(\eta_i) - \sum_{i \in A} p_{\eta_i} + \sum_{i \in A} p_i = \sum_{i \in A} v_i(\eta_i)$$

□

Svensson (1983) showed the related result that envy-freeness implies efficiency in room-assignment rent-division problems [39].

Having established properties on outcomes, I now present properties to characterize agents' strategic behavior.

Definition 2.7. *A mechanism f is* ***strategy-proof*** *or dominant strategy incentive-compatible if for all agents $i \in A$ with true v_i*

$$u_i(f(v_i, \hat{v}_{-i})) \geq u_i(f(\hat{v}_i, \hat{v}_{-i})), \forall\ \hat{v}_i \in \mathbb{R}^n$$

A mechanism is strategy-proof if it is a best response for every agent to report its true valuations given the reports of others. While strategy-proofness has been a central requirement in the literature for mechanism design, I will argue in this thesis that its relaxation is appropriate in this setting.

Definition 2.8. *A mechanism f is* ***individually rational*** *if for all agents $i \in A$ and for any $\hat{v} \in \mathbb{R}^{n \times n}$*

$$u_i(f(\hat{v})) \geq 0$$

Individual rationality characterizes mechanisms in which no agent can be made strictly worse off for having participated and influences the practical viability of a proposed mechanism.

Definition 2.9. *A mechanism f is weakly budget-balanced iff $\sum_{i \in A} z_i(\hat{v}) \geq 0$ and strictly budget-balanced (or simply, budget-balanced) when $\sum_{i \in A} z_i(\hat{v}) = 0$.*

Budget-balance is a strategic consideration for the market operator: if calculating an allocation requires an outlay of capital greater than the payments collected from agents, it's very unlikely that a rationally motivated market operator will ever choose to operate the market. Budget-balance is a fundamental property of any practically viable mechanism.

2.3 Graph Properties

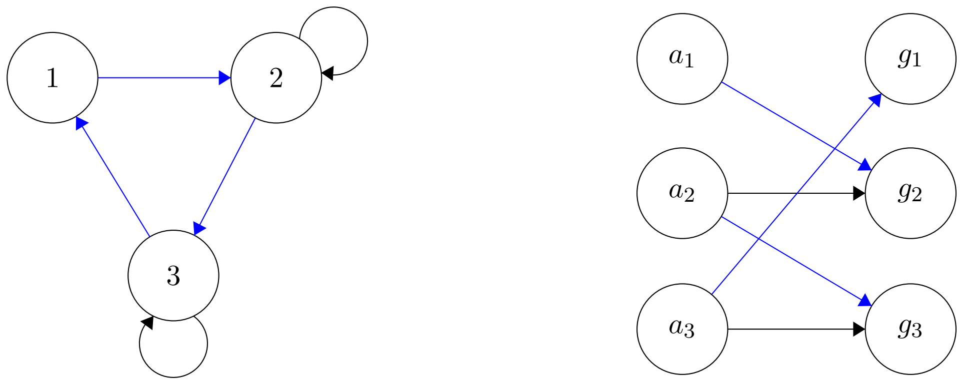
The agent reports $\hat{v}$ can be usefully interpreted as an edge-weighted, directed graph in matrix form. This *valuation graph* has A as its set of nodes. The edge (s, t) with weight $\hat{v}_s(t)$ represents the reported valuation of agent s for the item initially owned by agent t. By construction, a valuation graph is always complete.

Similarly, at prices p, the agent demand sets $D_i(p)$ define a directed graph. This *demand graph* $D(p)$ also has A as its set of nodes, but its directed edges are unweighted. An edge (s, t) is present in the graph if and only if $t \in D_s(p)$. A demand graph is usually not complete.

2.3.1 Cycle Covers and Matchings

Given a valuation graph, note that any assignment μ is valid if and only if the set of edges $\{(i, \mu_i), \forall i \in A\}$ forms a vertex-disjoint cycle cover.

Definition 2.10. *Given a graph $G = (V, E)$, a* ***vertex-disjoint cycle cover*** *is a set of edges $E' \subseteq E$ such that for each $v \in V$, v has in-degree exactly 1 and out-degree exactly 1.*



(a) Instance of Cycle Cover

(b) Equivalent Instance of Bipartite Matching

Figure 2.1: Example of reduction from the cycle cover problem to the bipartite matching problem. The cover and corresponding matching are shown in red.

Note that an assignment is efficient if and only if it is a *maximum-weight* vertex-disjoint cycle cover of the valuation graph. The cycle cover must be vertex-disjoint because of the unit-demand assumption. Self-loops are allowed in the cycle cover. An efficient mechanism therefore selects a maximum-weight vertex-disjoint cycle cover.

Two useful properties of the demand graph are also characterized by vertex-disjoint cycle covers. First, an assignment is a Walrasian equilibrium if and only if it is a vertex-disjoint cycle cover in the demand graph. Furthermore, a Walrasian equilibrium exists at p if and only if there exists a vertex-disjoint cycle cover in $D(p)$.

Clearly, computing vertex-disjoint cycle covers will figure prominently in any mechanism for the discrete exchange economy with money. A vertex-disjoint cycle cover can be identified or shown not to exist in polynomial time in the size of the graph. Likewise, a maximum-weight set of vertex-disjoint cycles can be found in polynomial time; because $\hat{v}$ represents a complete graph and $\hat{v} \in \mathbb{R}_{\geq 0}^{n \times n}$, a maximum-weight set of vertex-disjoint cycles necessarily covers the graph.

Proposition 2.2. *A (maximum-weight) vertex-disjoint cycle cover can be computed, or shown not to exist, in polynomial time in the size of the graph.*

Proof. Consider the following known reduction from the maximum-weight vertex-disjoint cycle cover problem to the problem of maximum-weight perfect matching in a bipartite graph.

Given a directed graph $G = (V, E)$ represented in matrix form by some $\hat{v}$, construct the bipartite graph $G_b = (V_L + V_R, E')$. Call V_L and V_R the left and right sets of nodes, respectively, in the bipartite graph. V_L and V_R each contain one node for each node in V. The set of edges in G_b is then $E' = \{(u_L, v_R) : (u, v) \in E\}$. If an edge exists between node u and node v in G, then an edge exists in the bipartite graph between node u_L and the node v_R in G_b. Note that this reduction requires $O(|V| + |E|)$.

By construction, a vertex-disjoint cycle cover C exists in G if and only if a perfect matching M exists in G_b. To see that this is true, note that for any edge $(u, v) \in E$ there is an edge $(u_L, v_R) \in E'$. If C is a vertex-disjoint cycle cover, it necessarily contains $|V|$ edges. Because the cover is vertex-disjoint, each $v \in V$ has exactly one in-edge and exactly one out-edge. Because every $e \in E$ has a corresponding $e' \in E'$, the edges in E' that correspond to the edges $C \subseteq E$ must form a perfect matching. The reverse direction follows in similar form.

It is well-known that a maximum-weight matching in a bipartite graph can be found in polynomial time in the size of graph using, for example, Kuhn's Hungarian algorithm [22]. The polynomial time reduction therefore shows that a maximum-weight vertex-disjoint cycle cover can be computed or shown not to exist in polynomial time in the size of the graph. □

An example of the reduction is shown in Figure 2.1. Throughout the thesis, it will frequently be useful to refer to the bipartite matching version of the discrete-exchange economy with money.

The second graph theoretic result that will be required is Hall's theorem.

Theorem 2.1. *Hall (1935) [21]. Given a bipartite graph $G = (L + R, E)$ where $N_G(S)$ is the set of nodes adjacent to any node in the set S of nodes, a matching exists that covers L if and only if*

$$\forall S \subseteq L, |S| \leq |N_G(S)|$$

Note that in a bipartite demand graph, the neighbor set of a node i is $D_i(p)$. As a result, Hall's theorem provides a necessary and sufficient condition for the existence of Walrasian equilibria. This fact will prove quite useful in this thesis.

Chapter 3

Impossibility and Insufficiencies

As I noted in the introduction, the optimal mechanism for the discrete exchange economy with money would satisfy all of the properties of

- Strategy-proofness (SP),
- Individual rationality (IR),
- (weak) budget-balance (BB),
- and efficiency.

Positive results, such as McAfee's double auction [23] and Demange, Gale, and Sotomayor's multi-item auction [14], have shown that mechanisms simultaneously satisfying all of these properties do exist in other settings. However, the canonical Myerson-Satterthwaite impossibility on two-sided exchange precludes the existence of such an ideal mechanism for the discrete exchange economy with money because a two-sided exchange economy is a special case of the discrete exchange economy with money [27].

It is possible that an approximately efficient mechanism, if it had other desirable properties, could perform well for the discrete exchange economy with money. However, this chapter begins by showing an impossibility result that strategy-proofness, individual rationality, budget-balance, and approximate efficiency cannot be simultaneously satisfied by any mechanism, either.

The chapter continues by presenting two families of mechanisms in the literature that are naturally applied to this setting: the Vickrey-Clarke-Groves (VCG) mechanism and Gale's top trading cycles (TTC) mechanism. I characterize the properties of VCG – strategy-proofness, individually rationality and exact efficiency but not budget-balance and note the properties of TTC – strategy-proofness, individually rationality, and budget-balance but not efficiency. I use these characterizations to motivate the relaxation of strategy-proofness that follows in Chapter 4.

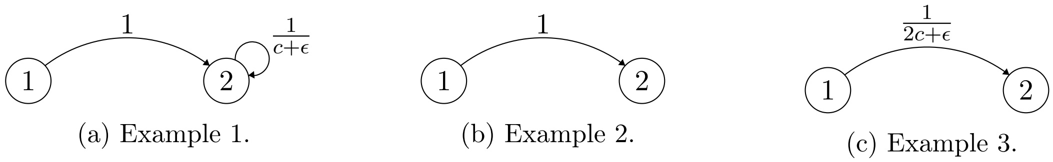


Figure 3.1: Examples for the impossibility result. Each node is labeled with the agent it represents, and each edge (i, j) is labeled with agent i's reported valuation for good j. Omitted edges have weight 0.

3.1 Impossibility

In this section, I will give an impossibility result and then provide examples of mechanisms in the literature that satisfy subsets of the desiderata.

Theorem 3.1. *There exists no mechanism for two agent discrete exchange economies with money that is simultaneously strategy-proof, individually rational, (weakly) budget-balanced, and c-efficient for* $c \in (0, 1]$.

Proof. The proof is by contradiction. Throughout the argument, $z_{e,i}$ refers to the payment made in example e by agent i in Figure 3.1. Similarly, $\hat{v}_{e,i}$ refers to the report in example e by agent i, and $\mu_{e,i}$ is the item assigned to agent i with the reports in example e.

For contradiction, let $f = (x, t)$ be a mechanism that is strategy-proof, individually rational, (weakly) budget-balanced and c-efficient. The properties of such a mechanism f must hold for any true agent valuations v.

Consider two agents with reported valuations as specified in Figure 3.1a, where the weighted edge (i, j) in the figure indicates $\hat{v}_{1,i}(j)$. Call these reports $\hat{v}_1$.

In order for an allocation to be c-efficient, the assignment must be $\mu_1 = (1, 2)$, because all alternative assignments μ' have $\sum_{i \in A} \hat{v}_{1,i}(\mu'_i) = 0$. To satisfy c-efficiency, then,

$$x(\hat{v}_1) = (1, 2)$$

Given that f must choose this assignment, the mechanism must assign agent 2 a payment $z_{1,2}$ such that individual rationality is maintained, as follows:

$$u_2(\mu_1, z_1) = 0 - \frac{1}{c + \epsilon} - z_{1,2} \geq 0 \implies z_{1,2} \leq -\frac{1}{c + \epsilon}$$

Call $\hat{v}_2$ the reports shown in Figure 3.1b. When agents make these reports, c-efficiency requires that $x(\hat{v}_2) = \mu_2 = (1, 2)$. Consider the case that $\hat{v}_{2,2} = v_{2,2}$, i.e. that $\hat{v}_{2,2}$ is agent 2's truthful report. Because f is strategy-proof, report $\hat{v}_{1,2}$ cannot be a useful deviation for agent 2 away from its true report $\hat{v}_{2,2}$. It must follow that:

$$\begin{aligned} u_2(\mu_1, z_1) &\leq u_2(\mu_2, z_2) \\ 0 - v_2(1) - z_{1,2} &\leq 0 - v_2(1) - z_{2,2} \\ z_{2,2} \leq z_{1,2} &\leq -\frac{1}{c+\epsilon} \end{aligned}$$

The last inequality follows from the previous bound on $z_{1,2}$. Furthermore, because f is weakly budget-balanced, it must be the case that

$$\begin{aligned} z_{2,1} + z_{2,2} \geq 0 & \\ z_{2,1} &\geq -z_{2,2} \geq -z_{1,2} \\ \implies z_{2,1} &\geq \frac{1}{c+\epsilon} \end{aligned}$$

To reach the contradiction, consider the example report $\hat{v}_3$ shown in Figure 3.1c. As before, c-efficiency requires that the exchange occurs. Because f is strategy-proof, it must be the case that

$$\begin{aligned} u_1(\mu_2, z_2) &\geq u_1(\mu_3, z_3) \\ v_1(2) - v_1(1) - z_{2,1} &\geq v_1(2) - v_1(1) - z_{3,1} \\ 1 - 0 - z_{2,1} &\geq 1 - 0 - z_{3,1} \\ z_{3,1} &\geq z_{2,1} \end{aligned}$$

Consider then the utility to agent 1 if $\hat{v}_{3,1} = v_1$ i.e. $\hat{v}_{3,1}$ are agent 1's true preferences. Then:

$$\begin{aligned} u_1(f(\hat{v}_3)) &= v_1(2) - v_1(1) - z_{3,1} \\ &\leq \frac{1}{2c+\epsilon} - 0 - z_{2,1} \\ &\leq \frac{1}{2c+\epsilon} - \frac{1}{c+\epsilon} \\ &< 0 \end{aligned}$$

This shows that f cannot be individually rational for agent 1 with true valuations $v_1 = \hat{v}_{3,1}$. This is a contradiction to the existence of f. □

Note that the impossibility must also hold for the $n > 2$ agent case because we can reduce a two-agent instance to an n agent instance by adding $n - 2$ agents who own items that every agent, including the first two, value at 0. This impossibility eliminates c-efficiency as a design objective, because we have options that are exactly efficient as soon as at least one property is dropped.

Furthermore, we can observe that the impossibility result is tight, meaning that mechanisms

Mechanism	SP?	IR?	BB?	($c-$)Efficiency?
Top Trading Cycles with Fixed Prices	Yes	Yes	Yes	No
Vickrey-Clarke-Groves (VCG)	Yes	Yes	Weak Deficit	Exact
Two-sided VCG	Yes	No	Yes	Exact
NetAscendingClock	No	Yes	Yes	Exact

Table 3.1: The mechanisms considered and their properties. Properties listed are with respect to reported valuations.

with any subset of the properties exist. I have already claimed the properties of TTC and VCG and will consider them in detail later in the chapter. Each of the mechanisms that I analyze are listed with their properties in Table 3.1, with the exception of the two-sided VCG mechanism, which is known to be strategy-proof, budget-balanced, and efficient but is not be individually rational in this setting because it ignores initial endowments.

To observe quickly mechanisms with the other property subsets, note that the identity mechanism, in which every agent is assigned its initial endowment and paid nothing, is trivially individually rational, budget-balanced and strategy-proof. We can construct a trivially individually-rational, budget-balanced and efficient mechanism by calculating an efficient assignment and assigning payments so that all agent utilities are 0; because efficiency implies that welfare has improved, weak budget-balance is immediately satisfied. Finally, we can ignore agent endowments and use the two-sided VCG mechanism to get strategy-proofness, budget-balance and efficiency.

We are now confronted with a dilemma about which property should be relaxed and which should be prioritized. Budget-balance cannot be easily sacrificed, since mechanisms that fail to satisfy budget-balance require a market operator who will be bear the cost of operation. A rational (i.e. profit-seeking) operator would never do so. Lack of individual rationality threatens the *unraveling* of a market, which occurs when so many agents prefer to either maintain the status quo or to discover and negotiate trades outside of the mechanism that the market can no longer function [34]. An efficient mechanism that is not strategy-proof might select suboptimal allocations because of agent misreports, and an strategy-proof mechanism that is not efficient could simply provide inadequate improvement on the status quo.

In the remainder of this chapter, I will demonstrate that the standard solutions, namely the TTC mechanism and the VCG mechanism, have theoretical flaws that make them unconvincing: TTC, even when prices are introduced, does not guarantee efficiency, and the appropriate VCG mechanism fails to satisfy budget-balance. I also show that neither VCG nor TTC selects allocations that are guaranteed to be envy-free or Walrasian equilibria. I use these observations to justify the relaxation of strategy-proofness in NetAscendingClock.

3.2 Top Trading Cycles with Fixed Prices

Gale's *top trading cycles* (TTC) mechanism is one of the fundamental positive results in the design of multi-sided matching markets. In its standard form, the mechanism is defined on the ordinal preference domain rather in the cardinal preference domain that is the focus of this thesis.

In ordinal TTC, agents report rankings relative to their initial endowments. Given these reports, the mechanism identifies mutually agreeable cycles of trades such that every agent "trades up." To do this, the mechanism proceeds in rounds. In the first round, every agent i points to the owner of the item (perhaps its own) that maximizes i's valuation. If the preferences on trades indicated by agents form a cycle, these cycles are recorded and the agents and items in them are removed. The point-and-clear process continues until every agent participates in a cycle, which may be a self-loop. The mechanism is strategy-proof and individually rational and selects outcomes that are Pareto efficient (but not *necessarily* welfare efficient) [37].

The adaptation of TTC to cardinal preferences is due to Miyagawa (2001) [26]. Miyagawa presents an intuitive reduction from an instance of a discrete exchange with money to an instance of a discrete exchange without money: set a vector of prices p, calculate each agent's utility for each item $g \in G$ with respect to p, and then construct each agent's ordinal preferences by placing items in order of decreasing utility for that agent, breaking ties at random. Ordinal TTC is then used to determine an assignment, with payments calculated using the prices.

Unavoidably, this reduction destroys information about the agents' cardinal preferences because degrees of magnitude are exchanged for a simple ordering. The quality of outcomes therefore depends on the careful selection of the prices used in the reduction. The example below illustrates this point.

Example 3.1. *Let o_i be the ordinal preferences of agent i, where $\succ_i$ is agent i's strict preference relation over items. Consider the two-agent economy defined by the following true valuation matrix:*

$$v = \begin{bmatrix} 500 & 1000 \\ 5 & 10 \end{bmatrix}$$

When $p = (0, 0)$:

$$\begin{aligned} u_1 &= (500, 1000) - (0, 0) + (0, 0) \\ &= (500, 1000) && \implies o_1 = 2 \succ_1 1 \\ u_2 &= (5, 10) - (0, 0) + (0, 0) \\ &= (5, 10) && \implies o_2 = 2 \succ_1 1 \end{aligned}$$

When $p = (0, 6)$:

$$\begin{aligned} u_1 &= (500, 1000) - (0, 6) + (0, 0) \\ &= (494, 1000) && \implies o'_1 = 2 \succ_1 1 \\ u_2 &= (5, 10) - (0, 6) + (6, 6) \\ &= (11, 10) && \implies o'_2 = 1 \succ_1 2 \end{aligned}$$

In a weak sense, o_i does capture the "shape" of each v_i, but we can see that it does not represent the cardinal preferences particularly well. The representation is substantially improved by the introduction of well-chosen prices — in fact, choosing good prices can make the selected allocation a Walrasian equilibrium. When $p = (0, 6)$ induces the preferences o', this occurs: each agent is assigned the first item in its ordinal preferences, which at least weakly maximizes its utility. Because each agent's utility is maximized, the allocation is a Walrasian equilibrium and is therefore both envy-free and exactly efficient.

However, the fact that TTC might proceed to a second clearing iteration immediately demonstrates that the mechanism cannot in general select envy-free allocations or find Walrasian equilibria, unless there is a method that can select Walrasian equilibrium prices without breaking strategy-proofness. The Myerson-Satterthwaite impossibility indicates that this cannot be possible. Miyagawa in fact offers the following stronger result:

Theorem 3.2. *Miyagawa (2001), Theorem 1. If a mechanism is strategy-proof, individually rational, non-bossy, and onto, then it is a fixed-price core mechanism (TTC with fixed prices) [26].*[2]

In a non-bossy mechanism, no agent can change the allocations of others without changing its own allocation; Miyagawa includes non-bossiness to distinguish TTC from VCG, which is individually-rational and strategy-proof but not non-bossy.

Miyagawa's theorem is vital to understanding to the generalization of TTC to a setting with cardinal preferences. The theorem states that as soon as strategy-proofness, individual rationality and non-bossiness are required as properties, the mechanism *must* be TTC with prices and the prices *must* be fixed independently of agent reports. For this reason, I call this mechanism *TTC with fixed prices.*

3.2.1 Efficiency Approximation with Exogenous Information

Miyagawa's theorem leaves us with the vital question of how prices should be fixed without considering agent reports. In the paper, Miyagawa acknowledges this but does not provide an answer.

[2]All of the mechanisms considered in this thesis are onto, because we assign each agent to one and only one item and require that all agents be assigned.

The theorem implies that good prices need to be discovered exogenously, because agent reports cannot be considered. With relatively general conditions on distributions of agent valuations, however, I claim that top trading cycles with fixed prices can achieve an ex-ante constant approximation of efficiency as the size of the market gets large.

Consider a market with agents i such that $v_i(j) \sim D(g_j)$, where g_j is a hyper-parameter to the generative distribution for valuations and is known to the operator; these hyper-parameters might be market prices for the items from some external source. I make the following conjecture:

Conjecture 3.1. *Given a known $g \in \mathbb{R}^n$ and $v_i \sim D(g_i)$ for all $i \in A$,*

$$\lim_{n\to\infty} \mathbb{E}_{v\sim D(g)} \left[\frac{OPT(A, G)}{\sum_{i\in A} v_i(TTC(v)_i)} \right] = c$$

for constant $c \geq 1$.

The conjecture above therefore says that the expected ratio of the optimal welfare to the welfare achieved by the mechanism will approach a constant as markets grow large. If the conjecture holds for some c close enough to 1, TTC with fixed prices may provide efficiency that is at least adequate in some sense.

And yet, even this conjecture is a form of begging the question of how prices should be fixed. The conjecture still requires that the operator know the hyper-parameters (perhaps from past transactions with similar items) and requires that agent valuations follow a fixed distribution given those hyper-parameters (perhaps because previous prices are known to all agents but they have idiosyncratic preferences). This is not impossible to imagine, but it clearly places a higher burden on the operator than would a mechanism that can set prices as a function of agent reports. In Chapter 5, I will assume that the hyper-parameters are known to the operator in order to evaluate the efficiency performance of TTC with fixed prices through simulation when agent valuations are draw from a fixed distribution.

Finally, note that Miyagawa's theorem highlights the tension between strategy-proofness and efficiency when budget-balance and individual rationality are required. TTC with fixed prices prioritizes the former by relaxing exact efficiency to Pareto efficiency. NetAscendingClock does the opposite, instead relaxing strategy-proofness in favor of finding efficient Walrasian equilibria.

3.3 The VCG Mechanism

Where TTC is the standard solution concept for settings with initial endowments and ordinal preferences, the jack-of-all-trades Vickrey-Clarke-Groves (VCG) family of mechanisms is applicable in any general setting. The choice rule in a VCG mechanism is the welfare-optimal assignment, such that

$$x(\hat{v}) = \underset{\mu \in \mathcal{M}}{\text{argmax}} \sum_{i \in A} \hat{v}_i(\mu_i)$$

Consistent with the model as I have described it, the VCG assignment is the maximum-weight bipartite matching in the graph between agents and items. Efficiency follows immediately from the choice rule.

In a VCG mechanism, each agent pays the magnitude of the welfare externality that it exerts on the other agents, where the externality of agent i is calculated as a function of all reports *excluding* agent i's. The discrete exchange economy with money requires a VCG payment rule that respects agents' initial endowments, as follows:

Definition 3.1. *A VCG payment rule that respects initial endowments is*

$$t_i(\hat{v}) = OPT(A \setminus i, G \setminus i) - OPT(A \setminus i, G \setminus a_i)$$

This VCG payment rule respects the initial ownership of item i by agent i when calculating agent i's externality on the other agents. The rule calculates the difference between the optimal welfare to all other agents when agent i participates and the optimal welfare to others when agent i does not participate. When agent i participates, item i is available for assignment to other agents. On the other hand, when agent i does not participate, item i is excluded because agent i never introduced it into the economy. Because the payment rule does not depend on the agent's own report, VCG payments are called *agent-independent*. Agent-independence implies strategy-proofness.

The VCG mechanism for the discrete exchange economy with money is individually-rational, exactly efficient, and strategy-proof. It is not budget-balanced.

Theorem 3.3. *The initial-endowments formulation of the VCG mechanism for the discrete exchange economy with money is individually rational, exactly efficient, and strategy-proof.*

Proof. I show that $\forall i \in A, u_i(\mu_i, z_i) \geq 0$.

$$\begin{aligned} u_i(\mu_i, z_i) &= v_i(\mu_i) - v_i(i) - z_i \\ &= v_i(\mu_i) - v_i(i) - OPT(A \setminus i, G \setminus i) + OPT(A \setminus i, G \setminus \mu_i) \\ &= OPT(A, G) - v_i(i) - OPT(A \setminus i, G \setminus i) \\ &\geq 0 \end{aligned}$$

The third equality follows because μ_i is the item assigned to i in the welfare optimal assignment. The inequality follows from the fact that $OPT(A, G)$ is the welfare of the efficient assignment of the agents A to the items G. If $v_i(i) + OPT(A \setminus i, G \setminus i)$ were larger in magnitude, this optimality would be violated. This shows that VCG with initial endowments is individually rational.

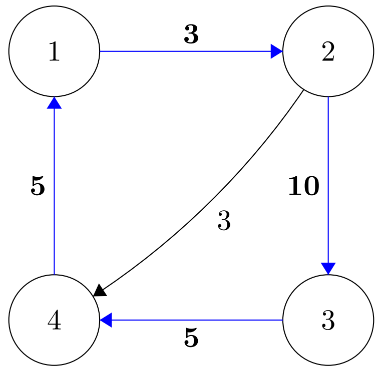


Figure 3.2: An example economy, on which VCG runs a deficit. The edges shown in blue form the efficient assignment.

Efficiency follows from the definition of the choice rule, and strategy-proofness follows from the agent-independence of the payment rule. □

Recall that the Myerson-Satterthwaite impossibility requires that individually rational, efficient, and strategy-proof mechanisms cannot also be budget-balanced across all problem instances. Because VCG has the former three properties, it follows that there exists some instance in which VCG is not budget-balanced. For completeness, I now construct such an example.

Example 3.2. *Consider the valuations shown in Figure 3.2. The mechanism begins by identifying the welfare optimal assignment of goods to agents. It is evident by inspection in this valuation network that the optimal assignment is* $(2, 3, 4, 1)$*, where* $V^{OPT} = 10 + 5 + 5 + 3 = 23$.

The payments are then:

$$\begin{aligned} z_1 &= (10 + 5 + 0) - (10 + 5 + 5) = -5 \\ z_2 &= (5 + 5 + 0) - (5 + 5 + 3) = -3 \\ z_3 &= (3 + 5 + 3) - (5 + 3 + 10) = -7 \\ z_4 &= -(3 + 10 + 0) - (3 + 10 + 5) = -5 \\ \sum_{i \in A} z_i &= -20 \end{aligned}$$

While this example confirms the impossibility, it provides no information about VCG's overall budget properties for the discrete exchange economy. The VCG mechanism turns out to be governed by a stronger budget property: the mechanism runs a weak deficit on any instance of the discrete exchange economy with money.

Theorem 3.4. *The VCG mechanism with initial endowments runs a weak deficit, i.e.* $\sum_{i \in A} z_i \leq 0$*, in all instances of the discrete exchange economy with money.*

Proof. I begin by noting a weaker upper bound on the payments made by the agents, as follows

$$\begin{aligned}
z_i &= OPT(A \setminus i, G \setminus i) - OPT(A \setminus i, G \setminus a_i) \\
\sum_{i \in A} z_i &= \sum_{i \in A} OPT(A \setminus i, G \setminus i) - OPT(A \setminus i, G \setminus a_i) \\
&= \sum_{i \in A} OPT(A \setminus i, G \setminus i) - (OPT(A, G) - v_i(a_i))) \\
&= \sum_{i \in A} OPT(A \setminus i, G \setminus i) - (n-1)OPT(A, G) \qquad (3.1)
\end{aligned}$$

Note that $V(A \setminus i, G \setminus i) \leq V(A, G)$ because $v_i(j) \geq 0$. This gives the following bound:

$$\sum_{i \in A} z_i \leq V(A, G)$$

This bound says that the operator receives at most the optimal welfare in payments from the agents; this is also trivially required by individual rationality. I now show the stronger bound that

$$\sum_{i \in A} z_i \leq 0$$

.

Recall the definition of a valuation graph from Chapter 2. Rather than the complete valuation graph, let $\gamma(A')$ be the *partial* valuation graph that includes only the edges in the optimal assignment of the agents A' to the goods initially owned by the agents A'. Note that this graph leads to welfare of at most $V(A', G')$.

Construct the graphs $\gamma_i = \gamma(A \backslash i)$ for each $i \in A$. Note that each γ_i has total welfare $V(A \backslash i, G \backslash i)$. Take the union of these graphs by "superimposing them" on each other, allowing for edges to be repeated in the union graph. Call this superimposed graph Γ.

Each node in Γ has $n-1$ incoming edges and $n-1$ outgoing edges (possibly self loops), because each agent is held out of exactly one of the n partial valuation graphs γ_i. By iterative application of Hall's theorem, Γ can be decomposed into $n-1$ graphs, each containing n nodes, such that each node in any of the graphs has out-degree exactly 1 and in-degree exactly 1. Because each graph has n nodes, the welfare in each of these graphs is at most $V(A, G)$, and because there are n graphs γ_i that each had total edge weight $V(A \setminus i, G \setminus i)$, this shows that

$$\sum_{i \in A} V(A \setminus i, G \setminus i) \leq (n-1)V(A, G) \implies \sum_{i \in A} V(A \setminus i, G \setminus i) - (n-1)V(A, G) \leq 0$$

Returning to the expression for the sum of payments in inequality 3.1, we have:

$$\sum_{i\in A} z_i = \sum_{i\in A} V(A\setminus i, G\setminus i) - (n-1)V(A,G) \le 0$$

as desired. □

While my primary goal was to show that VCG always runs a weak deficit, I also note that this deficit is bounded by the optimal welfare.

Theorem 3.5. *The deficit of the VCG mechanism for the discrete exchange economy with money is at most $V(A,G)$, such that $\sum_{i\in A} z_i \ge -V(A,G)$.*

Proof. I continue with the notation from the previous proof. Let μ be an efficient assignment.

I will lower bound $V(A\setminus i, G\setminus i)$ in terms of $V(A,G)$. To do so, consider removing each agent i from $\gamma(A)$ to form $\gamma(A\setminus i)$.

There are two possible cases to describe. If i has a self-loop in $\gamma(A)$, then $\gamma(A\setminus i)$ is still a valid because the rest of the graph is unaltered by removing i; that is, every agent $j \in A\setminus i$ still has an in-edge and an out-edge. This shows for this case that $V(A\setminus i, G\setminus i) \ge V(A,G) - [v_i(\mu_i) - v_i(i)]$.

If i is part of a cycle in $\gamma(A)$ that is not a self-loop, call π_i the node that precedes i in the cycle and σ_i the node that follows i in the cycle. When i is removed, delete the directed edges (π_i, i) and (i, σ_i) from $\gamma(A)$ and replace these edges with the edge (π_i, σ_i). Because agent σ_i's assignment, given by its out-edge, is unchanged, its valuation is also unchanged. However, the change to agent π_i's valuation is $v_{\pi_i}(\sigma_i) - v_{\pi_i}(i)$. This shows that

$$V(A\setminus i, G\setminus i) \ge V(A,G) - v_i(\sigma_i) - [v_{\pi_i}(\sigma_i) - v_{\pi_i}(i)]$$

Summing across i we have:

$$\sum_{i\in A} V(A\setminus i, G\setminus i) \ge \sum_{i\in A} V(A,G) - v_i(\sigma_i) - [v_{\pi_i}(\sigma_i) - v_{\pi_i}(i)]$$

Because an assignment is always a vertex-disjoint cycle cover, each agent appears exactly once as σ_i and once as π_i. It follows from this observation and the expression above that

$$\sum_{i\in A} V(A\setminus i, G\setminus i) \ge nV(A,g) - 2V(A,G)$$

$$\begin{aligned}\sum_{i\in A} z_i &= \sum_{i\in A} V(A\setminus i, G\setminus i) - (n-1)V(A,G) \\ &\ge (n-2)(V(A,G)) - (n-1)V(A,G) \\ &= -V(A,G)\end{aligned}$$

This shows the desired result that the deficit is at most the optimal welfare. □

The budget properties of the VCG mechanism cast doubt on its usefulness in practice. A purely rational market operator would never choose to implement it because it is guaranteed to never make a profit. Strategic concerns in settings where exact efficiency is required would be the only justification: if the operator could relax exact efficiency but not strategy-proofness, it would choose TTC, and if it could relax strategy-proofness, it would choose NetAscendingClock.

3.3.1 Lack of Fairness in VCG Allocations

Even setting aside VCG's budget deficiencies, efficiency and incentive-compatibility are not enough to make VCG compelling: VCG allocations can be unstable because they are not fair.

Theorem 3.6. *The VCG mechanism for the discrete exchange economy with money is not envy-free.*

Proof. Consider the reported valuations show in this valuation matrix:

$$\hat{v} = \begin{bmatrix} 5 & 2 & 0 & 2 \\ 7 & 1 & 4 & 6 \\ 1 & 1 & 4 & 6 \\ 4 & 3 & 6 & 6 \end{bmatrix}$$

Given this report, the selected allocation is

$$\mu = (2, 1, 4, 3)$$
$$z = (-6, 3, 0, -2)$$

(Note that the total budget is -5.) Given this allocation, agent 2's utility is:

$$7 - 1 - 3 = 3$$

However, agent 2 sees that agent 0 received a payment of 6 for receiving item 2. Agent 2's utility given the same payment and if it kept its own item would be

$$1 - 1 - (-6) = 6$$

While it is clearly counterintuitive that agent 2 would receive a net payment for keeping its own good, this example demonstrates that VCG is not envy-free. Note that the example also shows the non-triviality of the bound on payments, because agent 2 is assigned a positive payment. □

The lack of envy-freeness indicates that VCG does not necessarily select Walrasian equilibria. Furthermore, the payments are not easily interpretable, since they are exclusively defined as payments to agents rather than prices on items. In contrast, NetAscendingClock will provide both envy-freeness and interpretable prices in addition to budget-balance, individual rationality and exact efficiency.

3.4 Can VCG and TTC be combined?

We have seen that VCG and TTC have complementary properties: both are strategy-proof and individually rational, while the former is exactly efficient but not budget-balanced and the latter is not efficient but is budget-balanced. Because of this complementarity, first intuition might suggest that composing the VCG and TTC mechanisms could result in a high-quality mechanism.[3] Using VCG payments as prices to TTC would resolve VCG's budget problems by making payments inter-agent rather than between agents and the market operator, and the VCG payments could induce good TTC outcomes because the VCG payment rule is responsive to reports.

The composition fails for several theoretical reasons. If strategy-proofness were maintained, Myerson-Satterthwaite would require that exact efficiency could not be satisfied. Theorem 3.1 further guarantees that c-approximate efficiency cannot be satisfied, either. Moreover, Miyagawa's fixed-price theorem requires that introducing responsive VCG prices must break strategy-proofness.

For the sake of completeness, I illustrate how this lack of strategy-proofness emerges in the example below.

Example 3.3. *Consider the four-agent economy defined by the following valuation matrix:*

$$v = \begin{bmatrix} 8 & 10 & 8 & 0 \\ 1 & 4 & 3 & 7 \\ 7 & 2 & 4 & 2 \\ 8 & 7 & 4 & 0 \end{bmatrix}$$

The pairwise VCG payments for this valuation matrix are:

$$z = \begin{bmatrix} 0 & -1 & -3 & 1 \\ 1 & 0 & -2 & -4 \\ 3 & 2 & 0 & 4 \\ -1 & -2 & -4 & 0 \end{bmatrix}$$

where z_{ij} is the payment made by or to agent i for receiving item j. Note that because the VCG payment is defined separately for each agent-item pair, each agent faces its own vector of payments. Each agent will therefore face its own distinct prices when the VCG payments are repurposed as prices (not to mention that these prices do not satisfy the assumption of non-negativity). We will

[3] In fact, this intuition spurred the entire line of inquiry for this thesis!

see shortly that this introduces a new strategic phenomenon.

Given truthful reports, the TTC mechanism with the agents facing the payment matrix above selects the following allocation:

$$\mu = (3, 1, 2, 4)$$
$$z = (0, -3, 0, 3)$$

Consider the following useful deviation for agent 1:

$$\hat{v} = \begin{bmatrix} 6 & 7 & 0 & 5 \\ 1 & 4 & 3 & 7 \\ 7 & 2 & 4 & 2 \\ 8 & 7 & 4 & 0 \end{bmatrix}$$

With this deviation, the VCG payment matrix is now

$$z' = \begin{bmatrix} 0 & -1 & -3 & 1 \\ 1 & 0 & -2 & -2 \\ 4 & 5 & 0 & 4 \\ -1 & 0 & -4 & 0 \end{bmatrix}$$

Note that all of the payments faced by agent 1 are unchanged. This is consistent with the agent-independent pricing property of VCG. However, the payments faced by all of the other agents have changed. Because payments are now made between agents rather than to the market operator, this change matters.

The significance emerges when one notes that the TTC assignment is also unchanged given this misreport: all agents are assigned the same items. However, the payment to agent 2 for buying agent 1's good gets less negative; in other words, agent 1 owes agent 2 a smaller payment, so agent 1's utility increases.

The phenomenon here can be summarized as follows: because VCG payments are agent-independent, an agent cannot influence the prices it faces on other items by manipulating its own report. However, because TTC imposes inter-agent payments, an agent can use its report to manipulate the prices faced by *other* agents for the good that it initially owned. This indicates that the combination of TTC and VCG breaks both efficiency and strategy-proofness, as expected. VCG payments cannot be robustly used as prices in TTC.

This is consistent with the fact that strategy-proofness for VCG is known to break when an allocation that is not *exactly* efficient is chosen. This undesirable property presents one of the central challenges to applying VCG in combinatorial settings, where computing optimal welfare assignments can be NP-hard but using approximately efficient solutions can break strategy-proofness [10].

Chapter 4

Clock Methods: NetAscendingClock

In the previous chapter, I showed that no mechanism can be simultaneously strategy-proof, budget-balanced, individually rational and c-efficient. I further demonstrated that two conventional designs, VCG and TTC, have properties that make them inadequate for use in practice: VCG cannot balance the budget, while TTC provides no efficiency or even c-efficiency guarantees. These observations motivate the relaxation of strategy-proofness to find a mechanism that is budget-balanced, individually-rational and at least c-efficient.

This chapter pursues the stronger objective of defining a mechanism that is budget-balanced, individually rational, exactly efficient, and satisfies the additional fairness property of envy-freeness. Such a mechanism necessarily selects a Walrasian equilibrium.

I begin by proving the existence of Walrasian equilibrium in the discrete exchange economy with money by constructing a direct mechanism. That mechanism identifies the equilibrium through linear programming. As we know, the mechanism must necessarily lack strategy-proofness, and the direct mechanism on its own provides no further game theoretic intuition about how agents will behave in the absence of a dominant strategy equilibrium.

To build such intuition, I present NetAscendingClock, an indirect mechanism based on an iterative ascending-clock auction. The chapter's goal is to present details of the mechanism, prove that it converges to a Walrasian equilibrium in a finite number of iterations, and show that the mechanism is budget-balanced, individually rational, exactly efficient, and envy-free.

The central game theoretic benefit of NetAscendingClock is that the mechanism allows us reasonably to make a *price taking* assumption. Price taking, a qualitative measure of approximate strategy-proofness, states that agents will make truthful reports as long as the mechanism makes utility-maximizing allocations to agents given the prices that the mechanism announces. As we will see, NetAscendingClock satisfies this property through its selection of Walrasian equilibria. While a more formal analysis of NetAscendingClock's manipulability through quantitative methods like those summarized in Parkes and Lubin (2012) is warranted, I leave this as a direction for future work [31].

Informally, NetAscendingClock replicates the market action of a self-organizing multi-sided market, by running n simultaneous English auctions in which each seller can react to an increase in revenue it collects for its item by placing an updated bid on another item. The allocation is chosen when every bidder wins exactly one of the English auctions.

More formally, consider the bipartite demand graph as defined in Chapter 2. Given a set of posted prices, starting at zero for every item, each agent indicates the item or items that maximize its utility at the current prices. If a perfect matching exists in this demand graph, the mechanism selects the corresponding assignment and uses the current prices to set agent payments based on that assignment. If no perfect matching exists, NetAscendingClock identifies a set of items whose prices should be increased and then presents the agents with the new prices in the next iteration.

This chapter provides a detailed description of the mechanism by showing that the discrete exchange economy with money can be reduced to a special case of a room-assignment rent-division problem. This leads to a rule on how prices should be adjusted and provides proof that the iterative mechanism converges to an envy-free allocation and inherits the properties of budget-balance, individual rationality and efficiency from a previous mechanism for room-assignment rent-division. The principal innovations in the construction of the mechanism are showing that initial ownership does not break convergence and that the the mechanism can identify interpretable non-negative prices on items.

In the initial presentation of NetAscendingClock, I treat as a black box the detailed algorithm by which items are chosen for price updating. I return to this issue in the last section and prove that each iteration in NetAscendingClock can be tractably computed.

4.1 Walrasian Equilibria

In this section, I present a direct mechanism that is budget-balanced, individually rational, efficient, and envy-free. This direct mechanism therefore identifies a Walrasian equilibrium. I also note that the direct mechanism can add additional fairness constraints to allocations for the discrete exchange economy with money.

The theorem that establishes these claims follows a similar result for room-assignment rent-division (which will be introduced below) from Gal et. al. (2016) [15].

Theorem 4.1. *Let $f_1, \ldots, f_t : \mathbb{R}^n \to \mathbb{R}$ be linear functions where t is polynomial in n. Given an instance of the discrete exchange with money, a solution (μ, p) that maximizes the minimum $f_q(u_1(\mu, p), \ldots, u_n(\mu, p))$ over $q \in [t]$ subject to envy-freeness can be computed in polynomial time. Any such solution is a Walrasian equilibrium.*

Proof. The proof is by construction. Consider the following algorithm:

Algorithm 4.1. *Direct Mechanism for Walrasian Equilibrium.*

1. *Given reports* $\hat{v}$*, find* $\mu = x(\hat{v})$ *by solving the maximum weight matching problem in the bipartite valuation graph represented by* $\hat{v}$*.*

2. *Compute prices* p *by solving the following program:*

$$\text{maximize } R \tag{4.1}$$

$$\text{subject to } R \leq f_q(u_1(\mu, p), \ldots, u_n(\mu, p)) \qquad \forall q \in [t] \tag{4.2}$$

$$\hat{v}_i(\mu_i) - \hat{v}_i(i) - p_{\mu_i} + p_i \geq \hat{v}_i(g) - \hat{v}_i(i) - p_g + p_i \qquad \forall i \in A, g \in G \tag{4.3}$$

$$p_g \geq 0 \qquad \forall g \in G \tag{4.4}$$

$$R \in \mathbb{R} \tag{4.5}$$

3. *Set* $z_i = p_{\mu_i} - p_i$ *for each* $i \in A$*.*

That this algorithm selects allocations with the properties in the theorem can be quickly verified. Step 1 demonstrates clearly that the algorithm selects efficient assignments. Individual rationality is satisfied because $v_i(i) - v_i(i) - p_i + p_i = 0$, a lower bound on each agent's utility that is enforced by constraint 4.3. That constraint also enforces envy-freeness by definition. Because every agent is assigned exactly one item and initially owned exactly one item, budget-balance follows from the payment structure in step 3. Constraint 4.4 enforces the non-negativity of prices.

That this algorithm requires only polynomial time in the size of the economy can also be quickly verified. I have noted that the maximum weight matching problem can be solved in polynomial time. The mathematical program in step 2 has a clearly linear objective, and its constraints are by assumption linear with respect to prices. This shows that the program is a linear program. The number of constraints and variables is polynomial in the size of the economy, showing that this linear program can be tractably solved.

Calculating payments in step 3 requires linear time. This constructively proves the theorem. □

Maximizing the minimum of a set of linear functions of the agent utilities allows for the addition of tighter fairness constraints, such as the value- and money-Rawlsian solutions that maximize the minimum amount of money given to any agent subject to envy-freeness. Gal et al. conclude that such maximin solutions over *utility*, such that the minimum utility of any agent is maximized, are the most equitable in room-assignment rent-division [15]. Analyzing fairness alternatives is outside the scope of this paper, but I note the design flexibility also applies to the discrete exchange economy with money.

Algorithm 4.1 also offers benefits in computational efficiency over the VCG mechanisms. In total, Algorithm 4.1 requires only 2 optimizations that are polynomial-time in the number of agents. VCG requires, in constrast, $O(|A|)$ solutions to the maximum weight matching problem of size

$O(|A|)$: one to find the optimal welfare assignment, and one or two more to calculate each VCG payment, depending on how the optimal welfare assignment is memoized. In a large economy, the computational improvement would be non-trivial.

The theorem provides a useful corollary characterizing Walrasian equilibrium-selecting mechanisms in the discrete exchange economy with money:

Corollary 4.1. *Any mechanism that selects Walrasian equilibria in the discrete exchange economy with money is individually rational and budget-balanced and selects allocations that are envy-free and efficient.*

We will see that NetAscendingClock selects Walrasian equilibria.

4.2 Preliminaries for Clock Methods

In this section, I provide the basic tools required to describe NetAscendingClock. I first define overdemanded sets, which are used to identify items for price updating, and then I describe the mechanism from which NetAscendingClock inherits its properties.

4.2.1 Iteration and Overdemanded Sets

The goal of this section is to define the set of items whose prices should be changed at each iteration. Through this chapter, I refer to iterations with τ, beginning at $\tau = 1$. Consistent with the notation defined in Chapter 2, any set that is based on agent demand is defined with respect to prices p.

Recall from Chapter 2 that $D_i(p)$ is the demand set, i.e. the set of utility maximizing items, for agent i at a vector of prices p. NetAscendingClock uses agent demand sets to identify a set of items that block the formation of an assignment so that the prices on these items can be raised.

Such sets of items are in the family of overdemanded sets, which, as the name suggests, are those items that are demanded by too many agents for an assignment to be possible. Given a set of items S, the set of agents demanding only items in S is

$$O(S, p) = \{i : D_i(p) \subseteq S\}$$

The set of agents demanding any item in S is

$$U(S, p) = \{i : D_i(p) \cap S \neq \emptyset\}$$

Definition 4.1. *Given a set of prices p and demand sets $D_i(p)$ for each agent i, a set of items T is* ***overdemanded*** *if $|O(T, p)| > |T|$, such that the items in T are demanded by more agents than there are items.*

This definition describes intuition: there is obviously no way to assign three agents to two items, for example. By Hall's theorem, overdemanded sets provide a basic stopping condition for the iterative algorithm: if no overdemanded set exists, then Hall's theorem says that a perfect matching must exist in the bipartite demand graph of agents to items, where the neighbor set of an agent in the graph is its demand set.

Choosing an overdemanded set as the set of items whose prices should be raised at each iteration is by itself too general to guarantee convergence, however. For example, note that if any set of items is overdemanded, so is the set of all items. Raising prices on all items simultaneously would clearly leave agent demand sets unchanged. This fact motivates the minimal overdemanded set.

Definition 4.2. *A set of items T is a* ***minimal overdemanded set*** *if it is overdemanded and no non-empty proper subset $K \subset T$ is also overdemanded. Let $MOD(p)$ be the family of minimal overdemanded sets at prices p.*

In their one-sided multi-item auction, Demange, Gale, and Sotomayor proved that a mechanism that iteratively raises prices on minimal overdemanded sets converges to an efficient Walrasian equilibrium [14]. This observation is one of the properties underlying NetAscendingClock.

4.3 From Rent-Division to NetAscendingClock

Presenting NetAscendingClock is simplified significantly by connecting the discrete exchange economy with money to the related problem of room-assignment rent-division. In this problem, a set of agents A must determine how to divide a set of rooms R where $|R| = |A|$ and how to split a shared cost c, the rent, such that each agent is assigned a payment z_i where $\sum_{i \in A} z_i = c$.

Abdulkadiroğlu et al. (2004) defined an iterative, budget-balanced, and individually-rational mechanism for room-assignment rent-division that converges to an efficient and envy-free equilibrium allocation in a finite number of steps. Their mechanism broadens the minimal overdemanded set to the *full overdemanded set*. The full overdemanded set, $OD(p)$, is defined constructively by Algorithm 4.2.

Algorithm 4.2. *Construction of the full overdemanded set.*

1. *Given prices p and the demand graph $D(p)$ of all the agents' demand sets, find all minimal overdemand sets with respect to $D(p)$ and take their union. Call the intermediate set S.*

2. *Delete all items in S from the demand sets, so that $D_i(p) = D_i(p) \setminus S$, $D(p) = D(p) \setminus S$.*

3. *Find all minimal overdemand sets with respect to the updated demand sets. If any exist, add them to S and return to the previous step. Otherwise, continue.*

4. *Return S.*

It follows from the iterative construction of $OD(p)$ that this set is unique at a given price vector p. I leave the computational tractability of Algorithm 4.2 for discussion in Section 4.6.

Given the definition of $OD(p)$, we can define the room-assignment rent-division algorithm.

Algorithm 4.3. *Room-assignment rent-division*

1. *Set* $p = (\frac{c}{n}, \ldots, \frac{c}{n})$ *where* $n = |R| = |A|$.
2. *Find* $OD(p)$ *at the current prices.*
3. *If* $OD(p) = \emptyset$, *a perfect matching exists. Return that matching and* p.
4. *If* $OD(p) \neq \emptyset$, *increment* p_r *for each* $r \in OD(p)$ *until the demand set of some agent demanding only items in* $OD(p)$ *grows to include an item* $r \notin OD(p)$. *To maintain budget-balance, lower the price of each* $r \notin OD(p)$. *Return to Step 2.*

In the algorithm above, p is the rent associated with each room. The algorithm runs until the rents are divided so that every agent can be assigned to a room in its demand set. Abdulkadiroğlu et al. show that this algorithm converges, in a finite number of steps, to an envy-free allocation and that the algorithm is budget-balanced and individually rational [1]. They note Svensson's result that envy-free allocations are efficient in room-assignment rent-division [39].

While the algorithm calls for the continuous increasing and decreasing of prices, Abdulkadiroğlu et al. note that the only instances in time that matter are exactly those where some agent's demand set grows. If the mechanism is implemented indirectly, so that each agent is asked to report only its demand set at each iteration rather than reporting its complete cardinal valuations, the mechanism can require that all valuations be integer-valued and simply increase prices by one unit at a time. If the mechanism is implemented as a direct revelation mechanism, however, the price increment required to expand at least one agent's demand set is:

$$\phi(p) = \begin{cases} \min_{j \in O(OD(p),p)} \left(\max_{i \in G} u_j(i,p) - \max_{i \notin OD(p)} u_j(i,p) \right) & OD(p) \neq \emptyset \\ 0 & OD(p) = \emptyset \end{cases}$$

This increment is applied to change item prices as follows:

$$p_i^{\tau+1} = \begin{cases} p^\tau - \frac{|OD(p)|}{n}\phi(p^\tau) & i \notin OD(t) \\ p^\tau + \frac{n-|OD(p)|}{n}\phi(p^\tau) & i \in OD(t) \end{cases} \tag{4.6}$$

Beginning with the room-assignment rent-division mechanism above brings us most of the way to a mechanism for the discrete exchange economy with money. To see this, consider the special case of room-assignment rent-division when $c = 0$. When $c = 0$ in the algorithm above, either a perfect matching will exist when $p = 0^n$ or prices are changed such that budget-balanced is enforced. The mechanism will iterate until an envy-free, and therefore efficient, assignment is found. In the discrete exchange economy with money, such an assignment is a Walrasian equilibrium. I therefore begin with a direct application of the room-assignment rent-division algorithm to the discrete exchange economy with money.

The room-assignment rent-division algorithm charges each agent only the price of the item it is assigned. The utility function is therefore:

$$u_i(\mu, p) = v_i(\mu_i) - v_i(i) - p_{\mu_i}$$

This price-updating rule balances the budget.

Lemma 4.1. *For all iterations t, the price updating rule in Equation 4.6 maintains budget-balance.*

Proof. The proof is by induction. By assumption, we have $\sum_{i \in A} p_i^0 = 0$, since $p_i^0 = 0$ for all $i \in A$.

Consider then $p^{\tau+1}$ given that budget-balance held at p^τ. From this it follows:

$$\begin{aligned}
\sum_{i \in A} p_i^{\tau+1} &= \sum_{i \in OD(t^\tau)} p_i^{\tau+1} + \sum_{j \notin OD(p^\tau)} p_i^{\tau+1} \\
&= \sum_{j \in OD(p^\tau)} \left(p_i^\tau + \frac{n - |OD(p^\tau)|}{n} \phi(p^\tau) \right) + \sum_{i \notin OD(p^\tau)} \left(p_i^\tau - \frac{|OD(p^\tau)|}{n} \phi(p^\tau) \right) \\
&= \sum_{i \in A} p_i^\tau + \sum_{j \in OD(p^\tau)} \frac{n - |OD(p^\tau)|}{n} \phi(p^\tau) - \sum_{i \notin OD(p^\tau)} \frac{|OD(p^\tau)|}{n} \phi(p^\tau) \\
&= 0 + |OD(p^\tau)| \frac{n - |OD(p^\tau)|}{n} \phi(p^\tau) - (n - |OD(p^\tau)|) \frac{|OD(p^\tau)|}{n} \phi(p^\tau) \\
&= 0
\end{aligned}$$

This completes the proof. □

Because the room-assignment rent-division algorithm induces the same demand sets when applied to an instance of the discrete exchange economy with money as it does for an instance of room-assignment rent-division with $c = 0$, the mechanism clearly converges. The allocation selected is both efficient and envy-free. The subtraction of the agent's valuation for its initial item does not change its demand set.

However, note that in the current form, envy-freeness does not imply individual rationality because agents do not receive payment for the items they initially owned. This is illustrated by the example below.

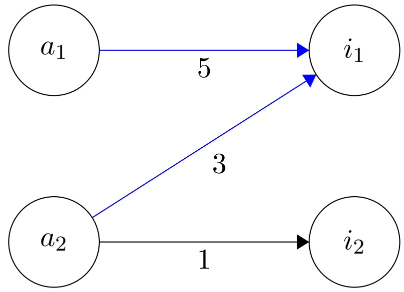


Figure 4.1: Rent division example valuation graph, in two-sided form. Edges that are also in the demand graph at $p^1 = (0, 0)$ are shown in blue.

Example 4.1. *Consider the valuations specified as a matrix below and also shown in Figure 4.1.*

$$\hat{v} = \begin{bmatrix} 5 & 0 \\ 3 & 1 \end{bmatrix}$$

Initialize $p_i^1 = 0$ for each item. At these prices, both agent 1 and agent 2 demand item 1. The set $S = \{1\}$ is therefore overdemanded; it is also clearly minimal overdemanded. No other item is demanded by any agent, so $OD(p^1) = \{1\}$ and $O(\{1\}, p^1) = \{1, 2\}$. Following the price incrementing rule, we have

$$\begin{aligned} \phi(p^1) &= \min_{i \in \{0,1\}} \left(\max_{j \in G} u_i(j, p^1) - \max_{j \notin OD(p)} \tilde{u}_i(j, p^1) \right) \\ &= 3 - 1 = 2 \end{aligned}$$

The updated prices are then:

$$\begin{aligned} p_1^2 &= 0 + \left(\frac{2-1}{2} \right) 2 = 1 \\ p_2^2 &= 0 - \left(\frac{1}{2} \right) 2 = -1 \end{aligned}$$

At iteration 2, then, agent 1 maximizes its utility by demanding item 1, and agent 2 is indifferent between item 1 and item 2. With these demand sets, there is perfect matching. Agent 1 is assigned item 1, its own, and agent 2 is assigned item 2, its own. However, they are each assigned a non-zero payment. For agent 1 in particular, note that $u_1(1, p_1) = 5 - 5 - 1 = -1$.

The example here illustrates that changing demand sets by adjusting only the price on the assigned item *does* result in convergence to a clearing assignment. The crucial observation is that an agent must be compensated for the item it initially owned, rather than all payments simply going to the operator. This diverges from room-assignment rent-division, where each share of the cost is paid to a third party (the landlord).

While this formulation is not individually rational, it does converge to an envy-free allocation, implying a Walrasian equilibrium, because the mechanism stops only when each agent can be assigned an item from its utility-maximizing demand set. All that remains is to restore individual rationality.

The natural modification to achieve individual rationality is simply to make payments inter-agent. I call this modified mechanism RentDivision.

The utility function in this modified scheme is exactly consistent with TTC:

$$u_i(\mu, p) = v_i(\mu_i) - v_i(i) - [p_{\mu_i} - p_i]$$

While the underlying room-assignment rent-division algorithm enforces budget-balance by increasing and decreasing prices at the same time, it is more straight-forward to observe that budget-balance is maintained by this formulation than the unmodified version, because every price is paid exactly once and received as payment exactly once.

RentDivision still converges to a Walrasian equilibrium because it induces the same demand sets as the room-assignment rent-division mechanism. Formally:

Lemma 4.2. RentDivision *converges to an individually rational Walrasian equilibrium in a finite number of iterations.*

Proof. In RentDivision, prices start at 0. By noting that agent demand sets will be the same at the first iteration in both the modified and unmodified approachs, it follows inductively that RentDivision induces the same demand sets as the unmodified room-assignment rent-division algorithm. More precisely, we have:

$$\begin{aligned}\underset{j \in 1 \dots n}{\operatorname{argmax}}\, u_i(\mu_j, p) &= \underset{i \in 1 \dots n}{\operatorname{argmax}}\, v_i(\mu_j) - v_i(i) - p_{\mu_j} + p_i \\ &= \underset{j \in 1 \dots n}{\operatorname{argmax}}\, v_i(\mu_j) - v_i(i) - p_{\mu_j}\end{aligned}$$

The second equality follows by symmetry, since the p_i term appears in the agent's utility for every item it can be assigned. This shows that the induced demand sets are unchanged by the inclusion of inter-agent payments. RentDivision therefore converges to a Walrasian equilibrium. □

With the new payment structure, individual rationality follows immediately, because an agent can always keeps its own item and pay nothing. More formally, consider an allocation that violates individual rationality, such that an agent leaves the market with negative utility. This implies that the agent was allocated an item outside of its demand set, because the agent could have received zero utility. Leaving with negative utility therefore contradicts RentDivision's stopping condition.

The analysis to this point leads to the following theorem:

Theorem 4.2. *The* RentDivision *mechanism for the discrete exchange economy with money is individually rational, budget-balanced, and efficient and converges to a Walrasian equilibrium in a finite number of iterations.*

Proof. I have shown that RentDivision is individually rational and budget-balanced and converges to a Walrasian equilibrium, which is by definition envy-free. Efficiency follows from the combination of individual rationality and envy-freeness, as shown in Chapter 2. □

At this stage, the RentDivision mechanism seems compelling: it has satisfied our desired properties of individual rationality, budget-balance and efficiency. I have also shown that it converges to a Walrasian equilibrium and is therefore envy-free. We have also seen that we can reduce an instance of a discrete exchange economy with money to an instance of room-assignment rent-division by setting $c = 0$ and using the rent-division prices to set inter-agent payments.

At this stage, we can also observe a connection to TTC: cycles are constructed iteratively using prices on purchase and sale. Rather than choosing between multiple demanding agents by finding closed cycles, however, RentDivision adjusts prices so that optimal cycles are identified. Furthermore, if these prices were available to TTC before agent reports were made, TTC would select the same Walrasian equilibrium. It seems reasonable, then, to think of RentDivision as providing the price discovery rule that Miyagawa showed could not exist in a strategy-proof version of TTC and therefore as a more flexible generalization of TTC to the discrete exchange economy with money.

4.4 Non-Negative Prices: NetAscendingClock

One piece of intuition is missing, however: while I have referred to the iter-agent payments in RentDivision as prices, prices "in the wild" are not typically posted as negative. We could interpret the positive and negative prices as bids and asks in the language of two-sided markets. On the other hand, a negative price lacks explanatory power.

This shortcoming in intuition can be addressed by a further modification of the room-assignment rent-division mechanism, leading to the final formulation of what I call the NetAscendingClock mechanism. The mechanism follows the same iterative procedure as the previous two mechanisms, with an adjustment to the price updating rule. The proposed price updating rule is:

$$p_i^{\tau+1} = \begin{cases} p_i^\tau & i \notin OD(p) \\ p_i^\tau + \phi(p) & i \in OD(p) \end{cases}$$

The payment rule given prices is unchanged from RentDivision; the utility function is therefore also unchanged. Note that the structure of the new price-updating rule precludes negative prices as long as the prices are initialized at at least 0. Once I have demonstrated that NetAscendingClock inherits the properties of RentDivision – namely, convergence to Walrasian equilibrium, efficiency, budget-balance, and individual-rationality, as well as intuitive, interpretable prices – we will have arrived at a mechanism that matches the objective towards which I set out at the beginning of this thesis.

First note that once again, every agent pays for the item it receives and is paid for the item it initially owned. Every price is clearly added exactly once and subtracted exactly once from the balance of payments, leading to a strict budget of zero. NetAscendingClock is therefore budget-balanced. The mechanism continues to satisfy individual-rationality, as well, because the iterative procedure terminates only when every agent can be assigned an item from its demand set given the prices, and every agent can achieve utility of zero by demanding its own item.

It remains then to show that the mechanism both converges and selects the envy-free allocation, which will imply that the allocation is both efficient and a Walrasian equilibrium. Because the price updating rule has changed, this requires proof.

Lemma 4.3. NetAscendingClock *converges to Walrasian equilibrium in a finite number of iterations and selects the welfare-optimal assignment of items to agents.*

Proof. I show that NetAscendingClock induces the same demand sets as RentDivision at every corresponding iteration. Let $p_{r,i}$ be the RentDivision prices, and $p_{n,i}$ be the NetAscendingClock prices.

At iteration $\tau = 1$, $p_r^1 = p_n^1 = 0^n$. Because the utility functions are structurally equivalent in the two mechanisms, the induced demand sets are identical at the first iteration of both mechanisms.

Assuming that the mechanisms induced the same demand sets at iteration $\tau \geq 1$, then, I show that the mechanisms must also induce the same demand sets at iteration $\tau + 1$. If the same set of items were demanded by every agent at iteration τ in both mechanisms, then the same set of items $OD(p)$ had their prices updated at τ in both mechanisms. To observe that the demand sets are in turn identical at the following iteration, note that the price changes in the respective mechanisms induce the same utility functions with respect to items at $\tau + 1$ if they induced the same set at τ. Iteratively updating the utility function between iteration τ and iteration $\tau + 1$, we can simply observe that the change in two utility functions is exactly equal between iterations. This is demonstrated algebraically in Figure 4.2 and Figure 4.3.

I have shown that at $\tau = 1$, the mechanisms induce the same utility functions and therefore the same demand sets in the first iteration. Both mechanisms therefore update the same set of item-prices. I have also shown that given the same set up of item-price updates at iteration τ, the two mechanisms induce the same utility function at iteration $\tau + 1$.

The two mechanisms therefore begin at the same prices, induce the same series of demand sets and result in the same agent utilities at every iteration. They must select the same assignment. Because RentDivision converges to an efficient Walrasian equilibrium, NetAscendingClock must as well. □

$$
\begin{aligned}
u_{r,i}(g,p^{t'+1}) &= v_i(g) - v_i(i) - p_g^{t'+1} + p_i^{t'+1} \\
&= \begin{cases}
v_i(g) - v_i(i) - \left(p_g^{t'} + \frac{n-|OD(p)|}{n}\phi(p^{t'})\right) + \left(p_i^{t'} + \frac{n-|OD(p)|}{n}\phi(p^{t'})\right) & g \in OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - \left(p_g^{t'} + \frac{n-|OD(p)|}{n}\phi(p^{t'})\right) + \left(p_i^{t'} - \frac{|OD(p)|}{n}\phi(p^{t'})\right) & g \in OD(p),\ i \notin OD(p) \\
v_i(g) - v_i(i) - \left(p_g^{t'} - \frac{|OD(p)|}{n}\phi(p^{t'})\right) + \left(p_i^{t'} + \frac{n-|OD(p)|}{n}\phi(p^{t'})\right) & g \notin OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - \left(p_g^{t'} - \frac{|OD(p)|}{n}\phi(p^{t'})\right) + \left(p_i^{t'} - \frac{|OD(p)|}{n}\phi(p^{t'})\right) & g \notin OD(p),\ i \notin OD(p)
\end{cases} \\
&= \begin{cases}
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} & g \in OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} - \phi(p^{t'}) & g \in OD(p),\ i \notin OD(p) \\
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} + \phi(p^{t'}) & g \notin OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} & g \notin OD(p),\ i \notin OD(p)
\end{cases} \\
&= \begin{cases}
u_{o,i}(g,p^{t'}) & g \in OD(p),\ i \in OD(p) \\
u_{o,i}(g,p^{t'}) - \phi(p^{t'}) & g \in OD(p),\ i \notin OD(p) \\
u_{o,i}(g,p^{t'}) + \phi(p^{t'}) & g \notin OD(p),\ i \in OD(p) \\
u_{o,i}(g,p^{t'}) & g \notin OD(p),\ i \notin OD(p)
\end{cases}
\end{aligned}
$$

Figure 4.2: Iteratively updated utility function for RentDivision.

$$
\begin{aligned}
u_{n,i}(g,p^{t'+1}) &= v_i(g) - v_i(i) - p_g^{t'+1} + p_i^{t'+1} \\
&= \begin{cases}
v_i(g) - v_i(i) - \left(p_g^{t'} + \phi(p^{t'})\right) + \left(p_i^{t'} + \phi(p^{t'})\right) & g \in OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - \left(p_g^{t'} + \phi(p^{t'})\right) + \left(p_i^{t'} + 0\right) & g \in OD(p),\ i \notin OD(p) \\
v_i(g) - v_i(i) - \left(p_g^{t'} + 0\right) + \left(p_i^{t'} + x(p^{t'})\right) & g \notin OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - \left(p_g^{t'} + 0\right) + \left(p_i^{t'} + 0\right) & g \notin OD(p),\ i \notin OD(p)
\end{cases} \\
&= \begin{cases}
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} & g \in OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} - \phi(p^{t'}) & g \in OD(p),\ i \notin OD(p) \\
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} + \phi(p^{t'}) & g \notin OD(p),\ i \in OD(p) \\
v_i(g) - v_i(i) - p_g^{t'} + p_i^{t'} & g \notin OD(p),\ i \notin OD(p)
\end{cases} \\
&= \begin{cases}
u_{n,i}(g,p^{t'}) & g \in OD(p),\ i \in OD(p) \\
u_{n,i}(g,p^{t'}) - \phi(p^{t'}) & g \in OD(p),\ i \notin OD(p) \\
u_{n,i}(g,p^{t'}) + \phi(p^{t'}) & g \notin OD(p),\ i \in OD(p) \\
u_{n,i}(g,p^{t'}) & g \notin OD(p),\ i \notin OD(p)
\end{cases}
\end{aligned}
$$

Figure 4.3: Iteratively updated utility function for NetAscendingClock.

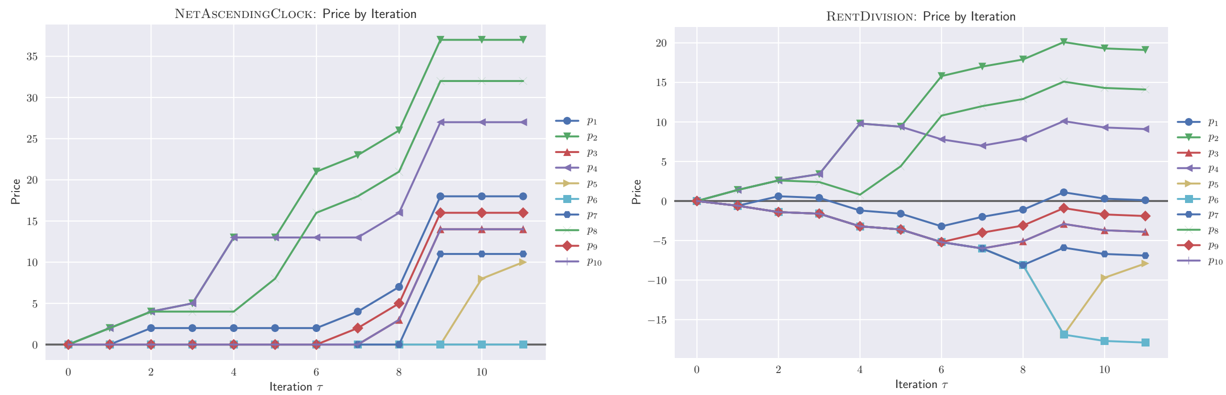


Figure 4.4: An example of price dynamics in a market with 10 agents in NetAscendingClock and RentDivision. Note that the differences in price magnitudes are equal in the two mechanisms.

I have just shown that NetAscendingClock is individually rational, budget-balanced, efficient, and envy-free. To illustrate the difference between NetAscendingClock and RentDivision, consider the price dynamics for a ten-agent example market shown in Figure 4.4. As the figure shows, both mechanisms identify a price change so that the price of at least one item changes at each iteration relative to the other prices – this is consistent with the mechanism's expansion of at least one agent's demand set at every iteration. The ascending clock dynamics are clearly visible here: prices increase over time until, at the last iteration, eight of the ten items have distinct prices and the market clears.

4.5 Connections

Notice that in Figure 4.4 the differences between the prices on individual prices chosen by RentDivision and NetAscendingClock appear equal. In this section, I confirm this observation by providing a bidirectional conversion between the zero-mean prices p_r set by RentDivision and the non-negative prices p_n set by NetAscendingClock. I have shown already that the mechanisms select the same allocation and assign the same payments to agents. Converting between the prices shows yet another layer of equivalence between the two mechanisms.

Theorem 4.3. *Given p_r identified by* RentDivision *and p_n identified by* NetAscendingClock*, there exists a conversion from p_r to p_n and a conversion from p_n to p_r.*

Proof. Consider any pairs of prices $p_{r,i}, p_{r,j} \in p_r$ and $p_{n,i}, p_{n,j} \in p_n$ generated by RentDivision and NetAscendingClock respectively. At any iteration of the mechanism, there are three possibilities for how $p_{r,i}, p_{r,j}$ changed with respect to each other. Either both i and j were in the full overdemanded set, exactly one was in the full overdemanded set, or neither was in the full overdemanded set. This observation also holds for $p_{n,i}, p_{n,j}$. I have shown previously that the two mechanisms select the same full overdemanded set at every iteration. I use this fact and the previous observation to show that at every iteration and for any pair of items i, j, $p_{r,i} - p_{r,j} = p_{n,i} - p_{n,j}$. The proof is by induction.

At $\tau = 1$, all prices are $p_{r,i}, p_{r,j}, p_{o,i}, p_{o,j} = 0$. The property follows trivially in this base case. Let S^τ be the full overdemanded set at iteration τ. Proceeding in cases from iteration $\tau + 1$, the price differences change as follows:

$$\begin{aligned}
p_{r,i}^{\tau+1} - p_{r,j}^{\tau+1} &= (p_{r,i}^\tau + \frac{n - |S^\tau|}{n}\phi(p_n^\tau)) - (p_{r,j}^\tau + \frac{n - |S^\tau|}{n}\phi(p_n^\tau)) && (i, j \in S^\tau) \\
&= p_{r,i}^\tau - p_{r,j}^\tau \\
p_{r,i}^{\tau+1} - p_{r,j}^{\tau+1} &= (p_{r,i}^\tau + \frac{n - |S^\tau|}{n}\phi(p_n^\tau)) - (p_{r,j}^\tau + \frac{n - |S^\tau|}{n}\phi(p_n^\tau)) && (i, j \notin S^\tau) \\
&= p_{r,i}^\tau - p_{r,j}^\tau \\
p_{r,i}^{\tau+1} - p_{r,j}^{\tau+1} &= (p_{r,i}^\tau + \frac{n - |S^\tau|}{n}\phi(p_n^\tau)) - (p_{r,j}^\tau - \frac{|S^\tau|}{n}\phi(p_n^\tau)) && (i \in S^\tau, j \notin S^\tau) \\
&= p_{r,i}^\tau - p_{r,j}^\tau + \phi(p_n^\tau) \\
p_{r,i}^{\tau+1} - p_{r,j}^{\tau+1} &= (p_{r,i}^\tau - \frac{|S^\tau|}{n}\phi(p_n^\tau)) - (p_{r,j}^\tau + \frac{n - |S^\tau|}{n}\phi(p_n^\tau)) && (i \notin S^\tau, j \in S^\tau) \\
&= p_{r,i}^\tau - p_{r,j}^\tau - \phi(p_n^\tau)
\end{aligned}$$

It is straightforward to verify the same property for p_n:

$$\begin{aligned}
p_{n,i}^{\tau+1} - p_{n,j}^{\tau+1} &= (p_{n,i}^{\tau+} + \phi(p_n^\tau)) - (p_{n,j}^\tau + \phi(p_n^\tau)) && (i, j \in S^\tau) \\
&= p_{n,i}^\tau - p_{n,j}^\tau \\
p_{n,i}^{\tau+1} - p_{n,j}^{\tau+1} &= (p_{n,i}^{\tau+} + 0) - (p_{n,j}^\tau + 0) && (i, j \notin S^\tau) \\
&= p_{n,i}^\tau - p_{n,j}^\tau \\
p_{n,i}^{\tau+1} - p_{n,j}^{\tau+1} &= (p_{n,i}^\tau + \phi(p_n^\tau)) - (p_{n,j}^\tau + 0) && (i \in S^\tau, j \notin S^\tau) \\
&= p_{n,i}^\tau - p_{n,j}^\tau + \phi(p_n^\tau) \\
p_{n,i}^{\tau+1} - p_{n,j}^{\tau} &= (p_{n,i}^\tau + 0) - (p_{n,j}^\tau + 0) && (i \notin S^\tau, j \in S^\tau) \\
&= p_{n,i}^\tau - p_{n,j}^\tau - \phi(p_n^\tau)
\end{aligned}$$

This shows by induction that the relative magnitudes of the respective components of p_r and p_n are equal at every iteration and are therefore equal at the last iteration. It follows from the same argument that for any prices p and q where q is a positive affine transformation of p, p and q will induce the same demand sets. Note that p_n is an affine transformation of p_r, and vice versa, as follows:

To convert from p_r to p_n, add $\min_i p_{r,i}$ to each component of $p_{r,i}$ so that all prices are non-negative. While this conversion does not necessarily produce p_n, it produces a non-negative price vector that induces the same demand sets as p and produces exactly p_n if any $p_{n,i} = 0$.

To convert from p_n to p_r, subtract $\bar{p}_n = \frac{1}{n}\sum_i p_{n,i}$ from each component of p_n so that prices are centered at 0. The vector of prices centered around 0 with relative magnitudes equal to p_r is unique, so this conversion provides p_n exactly. □

4.6 Computing NetAscendingClock Iterations

When defining NetAscendingClock, I presented an algorithm to construct the full overdemanded set that relied on a black box to find minimal overdemanded sets. In this section, I confirm that each iteration of NetAscendingClock is computationally tractable by providing a polynomial time algorithm for the identification of the full overdemanded set.

To do so, I note that Andersson et al. (2012) proved that any simple ascending auction converges to a minimum Walrasian equilibrium when a *maximum cardinality set in excess demand* is used to select prices for updating.

Definition 4.3. *A set of items $S \subseteq G$ is a set in* ***excess demand*** *if for any non-empty subset K of S, $|U(K,p) \cap O(S,p)| > |K|,\ \forall K \subseteq S,\ |K| \geq 1$.*

The family of sets in excess demand is denoted by $ED(p)$, and the maximum cardinality set in excess demand is $ED^*(p)$. Andersson et al. prove that $ED^*(p)$ is unique at any price vector and provide a algorithm based on the Ford-Fulkerson network flow algorithm with alternating augmenting paths, that either proves the existence of a perfect matching or finds $ED^*(p)$, in polynomial time in the size of the economy. Furthermore, Andersson et al. show that $MOD(p) \subseteq ED(p)$, such that any minimal overdemanded set is a set in excess demand, and show that the family of sets in excess demand is closed under union [5].

The tractability of iterations in NetAscendingClock follows from the fact that the maximum cardinality set in excess demand and the full overdemanded set are equivalent.

Theorem 4.4. *There exists a polynomial time algorithm to discover the full overdemanded set required by* NetAscendingClock*.*

Proof. I first show that $OD(p) \in ED(p)$. I then show that $OD(p)$ must be of maximum cardinality in the family of sets in excess demand. Because the maximum cardinality set in excess demand is unique, this will show that $OD(p) = ED^*(p)$, providing a polynomial time algorithm for the selection of $OD(p)$.

I demonstrate that $OD(p)$ is a set in excess demand by showing that the intermediate set of items selected in the construction of $OD(p)$ is a set in excess demand at every iteration. Let τ be the iteration in the construction of $OD(p)$, beginning at $\tau = 1$. Call S^τ the set of items that have been added to the in-construction set at or before iteration τ.

At $\tau = 1$, all minimal overdemanded sets are identified. The union of each of these sets in taken, and the resulting set is S^1. Note that any minimal overdemanded set is a set in excess demand, and the family of sets in excess demand is closed under union. S^1 is therefore a set in excess demand.

At the end of each iteration, the set of items S^τ is deleted from the demand set $D_i^\tau(p)$ of every agent for use in the next iteration $\tau + 1$. Formally:

$$D_i^{\tau+1}(p) = D_i^\tau(p) \setminus S^\tau$$

According to Algorithm 4.2 that constructs $OD(p)$, at iteration $\tau + 1$ the set M of sets of items that are minimal overdemanded with respect to the new demand sets $D_i^{\tau+1}(p)$ are joined with S^τ to create $S^{\tau+1}$, i.e. $S^{\tau+1} = S^\tau \cup \left(\bigcup_{m \in M} m \right)$. Note that these sets are *not* minimal overdemanded with respect to the original demand sets $D_i^1(p)$, because they would otherwise have already been added to S^τ.

Lemma 4.4. *Given $S^\tau \in ED(p)$ and a set T that is minimal overdemanded with respect to $D^\tau(p)$, $S^\tau \cup T \in ED(p)$.*

Proof. For notational clarity, I refer to $ED'(p)$ as the family of sets that are in excess demand with respect to $D^\tau(p)$ and to $MOD'(p)$ as the family of sets that are minimal overdemanded with respect to $D^\tau(p)$. Likewise I refer to $O'(K, p)$ as the set of agents for whom $D_i^\tau(p) \subseteq K$, consistent with the prior definition of O and $U'(K, p)$ as the set of agents for whom $D_i^\tau(p) \cap K \neq \emptyset$, also consistent with the prior definition of U.

By definition and assumption, then,

$$T \in MOD'(p) \subseteq ED'(p)$$

The result follows by showing that $S^\tau \cup T$ satisfies the definition of a set in excess demand with respect to the unaltered set of items, such that for all non-empty $K \subseteq S^\tau \cup T$

$$|U(K, p) \cap O(S^\tau \cup T, p)| > |K|$$

The proof is in cases with respect to K. When $K \subseteq S^\tau$, note that $U(K, p) \cap O(S^\tau \cup T, p) \supseteq U(K, p) \cap O(S^\tau, p)$. Because $S \in ED(p)$, by definition the condition then holds when $K \subseteq S$.

Consider $K \subseteq T$. As I have noted, if $T \in MOD'(p)$, then $T \in ED'(p)$. This yields that for all non-empty $K \subseteq T$,

$$|U'(K, p) \cap O'(T, p)| > |K|$$

Note that the set of agents $O'(T, p) \subseteq O(S^\tau \cup T, p)$ because $D_i^\tau(p) = D_i(p) \setminus S^\tau$ for all i. It follows that $U'(K, p) \cap O(S^\tau \cup T) \supseteq U'(K, p) \cap O'(T, p)$.

Finally note that because $S^\tau \cap T = \emptyset$, it must be the case that $U'(K,p) = U(K,p)$. Therefore

$$\begin{aligned}
U(K,p) \cap O(S^\tau \cup T, p) &\supseteq U'(K,p) \cap O'(T,p) \\
|U(K,p) \cap O(S^\tau \cup T, p)| &\geq |U'(K,p) \cap O'(T,p)| > |K|
\end{aligned}$$

where the last inequality follows from the definition of $ED'(p)$. This shows that the excess demand condition holds for all $K \subseteq T$.

Finally, consider $K \subseteq S^\tau \cap T$ where $K \not\subseteq S^\tau$ and $K \not\subseteq T$. For any such K, let $A = K \cap S$ and $C = K \setminus S$. By construction A and C are non-empty and disjoint, and $A \cup C = K$. Because $S^\tau \in ED(p)$ and $T \in ED'(p)$, by definition

$$\begin{aligned}
|U(A,p) \cap O(S^\tau, p)| &> |A| \\
|U'(C,p) \cap O'(T,p)| &> |C|
\end{aligned}$$

By construction $O(S^\tau \cup T, p) = O(S^\tau, p) \cup O'(T,p)$. I note also that $U(C,p) = U'(C,p)$. Then:

$$\begin{aligned}
|U(K,p) \cap O(S^\tau \cap T, p)| &= |U(A \cup C, p) \cap O(S^\tau \cup T, p)| \\
&= |U(A \cup C, p) \cap \left(O(S^\tau, p) \cup O'(T,p)\right)| \\
&= |U(A,p) \cap O(S,p) \cup \left(U'(C,p) \cap O'(T,p)\right)| \\
&= |U(A,p) \cap O(S,p)| + |\left(U'(C,p) \cap O'(T,p)\right)| \\
&> |A| + |C| \\
&= |A \cup C| \\
&= |K|
\end{aligned}$$

This shows that any subset of $S^\tau \cup T$ satisfies the excess demand condition. We have that $S^\tau \cup T \in ED(p)$. □

The construction of $OD(p)$ begins with the identification and union of all sets in $MOD(p)$. Because $MOD(p) \subseteq ED(p)$, the construction of $OD(p)$ begins with a set in excess demand. The algorithm then adds all sets in $MOD'(p)$ at each iteration. I have shown that $ED(p)$ is closed under union with such sets. By induction, $OD(p)$ is a set in excess demand at every iteration of its construction and at the termination of the algorithm, i.e. $OD(p) \in ED(p)$.

I prove now that $OD(p)$ is maximum.

Lemma 4.5. *OD(p) is a maximum cardinality set in excess demand at p.*

Proof. Assume for contradiction that there is some set $T \in ED(p)$ such that $|T| > |OD(p)|$. Let $S = T \setminus OD(p)$. By construction S is non-empty, $S \cap OD(p) = \emptyset$, and $S \subseteq T$. Because $T \in ED(p)$, we have:

$$|U(S,p) \cap O(T,p)| > |S|$$

Furthermore, because $S \cap OD(p) = \emptyset$, it must be the case that

$$U(S,p) \cap O(T,p) = U'(S,p) \cap O'(T,p)$$
$$|U'(S,p) \cap O'(T,p)| > |S|$$

This shows that $S \in ED'(p)$. Any set in excess demand is by definition also overdemanded; any set in excess demand with respect to D^τ is therefore also overdemanded with respect to D^τ. This fact proves the existence of a set $S' \subseteq S$ such that $S' \in MOD'(p)$, because S is overdemanded and therefore must contain a minimal overdemanded set, perhaps itself.

The existence of a set $S' \in MOD'(p)$, along with the fact that $S' \cap OD(p) = \emptyset$ by construction, is a contradiction, because the iterative construction of $OD(p)$ only terminates when no such set S' exists. This contradiction proves the lemma. $OD(p)$ is a maximum cardinality set in excess demand. □

Andersson et al. showed that the maximum cardinality set in excess demand is unique. This completes the proof of the theorem, showing that $OD(p)$ can be constructed with the Ford-Fulkerson algorithm proved by Andersson et al. to select the maximum cardinality set in excess demand. □

This demonstrates that each iteration in NetAscendingClock can be computed in polynomial time in the size of the economy. The NetAscendingClock mechanism can therefore be used as an iterative mechanism to identify budget-balanced Walrasian equilibria, which are necessarily envy-free and efficient. Because any agent's utility is lower bounded by zero, NetAscendingClock is also individually rational.

Chapter 5

Computational Analysis

In this chapter, I illustrate the theoretical differences between VCG, TTC, and NetAscendingClock by applying these mechanisms to simulated data. Recall that to this point I have demonstrated that VCG is strategy-proof, efficient, and individually rational but does not balance the budget. TTC with fixed prices is strategy-proof, individually rational, and budget-balanced but provides no guarantees about welfare optimality and also requires a exogenous price setting rule. In this chapter, my analysis focuses in particular on the inefficiency of TTC and on the deficit of VCG. I also examine the conjecture from Chapter 3 that TTC can achieve an ex-ante constant approximate of welfare with respect to a fixed distribution of agent valuations.

5.1 Simulation Design

Because we are interested in understanding both the properties of the mechanisms *and* when these properties are relevant, I test each mechanism on economies of varying sizes and with agents of varying preferences. The simulations are designed to satisfy the condition in the TTC efficiency conjecture I described in Chapter 3.

As such, I generate data by fixing hyper-parameters and using these hyper-parameters to sample agent valuations. To evaluate the mechanisms' properties in a diversity of settings, I assume that there are three types of agents. Each agent within a type has idiosyncratic preferences drawn from the same fixed distribution as the preferences of the other agents of that type. I construct economies consisting exclusively of each of the three types of agents (homogeneous economies), as well as economies containing equal mixtures of the types of agents (heterogeneous economies).

Concretely, I vary market size between three and 50 agents, and for each market size I perform 100 draws of agent valuations. To generate these valuations, I first generate ten draws of the hyper-parameter for each item in the economy (so that for an economy of size 50, 500 individual hyper-parameters are drawn to create ten hyper-parameter vectors g each of length 50). Each hyper-parameter is sampled as a random variable $g_j \sim \textsc{Unif}(400, 600)$. The valuation of agent i for

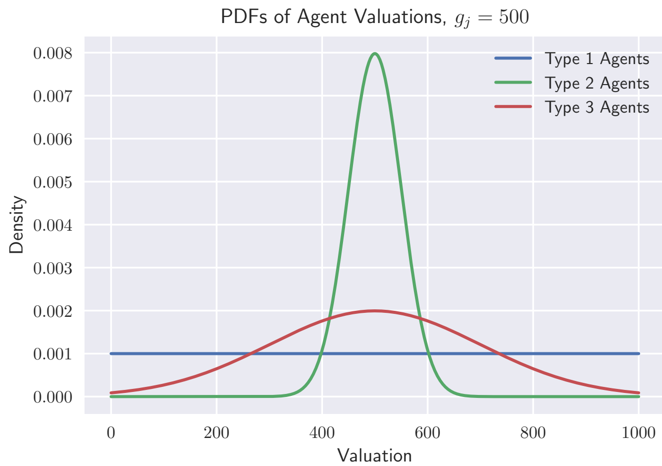


Figure 5.1: Probability density functions for agent types when the hyper-parameter is $g_j = 500$, $\sigma_2 = 50$ and $\sigma_3 = 200$. We can see that the different types lead to agents with a diversity of possible beliefs about the value of the item.

item j is then drawn as follows depending on agent i's type:

$$v_{1,i}(j) \sim \textsc{Unif}(0, 2g_j) \quad \text{(Type 1)}$$

$$v_{2,i}(j) \sim g_j + \epsilon_{2,i}, \epsilon_{2,i} \sim N(0, \sigma_2 = 50) \quad \text{(Type 2)}$$

$$v_{3,i}(j) \sim g_j + \epsilon_{3,i}, \epsilon_{3,i} \sim N(0, \sigma_3 = 200) \quad \text{(Type 3)}$$

$\epsilon_{2,i}$ and $\epsilon_{3,i}$ represent a perturbation around the hyper-parameter and σ_i is the standard deviation of each respective normal distribution. The perturbation represents the idiosyncratic preferences of each agent. The PDFs of the agent-item valuation distributions are shown in Figure 5.1.

After generating data, I run TTC without prices, TTC with fixed prices, VCG, and NetAscendingClock on each economy and record the following statistics of interest:

- The efficiency ratio $\frac{OPT(A,G)}{\sum_{i \in A} v_i(\mu_i)}$ achieved under TTC with and without prices, for comparison with the exactly efficient VCG and NetAscendingClock mechanisms and for evaluation of the conjecture about expected constant approximation of welfare.
- Total VCG deficit, for comparison with the strict budget balance of NetAscendingClock.
- Utility per agent, for comparison of all three mechanisms' outcomes including payments.
- Average trade cycle length, to evaluate the practicality of each mechanism's identified trades.

Simulating TTC with fixed prices clearly requires a rule on how prices should be fixed. To establish this rule, I run a cross-validation on a multiplier $\alpha \in [0, 2]$ of the hyper-parameter such

Economy	α
Heterogeneous	1.077
Type 1 Homogeneous	1.899
Type 2 Homogeneous	1.0
Type 3 Homogeneous	0.949

Table 5.1: Cross-validated multiplier values for TTC price fixing. Optimal welfare curves given α are shown in the appendix.

that prices are set as $p_i = \alpha g_i$. I use the range $[0, 2]$ because $\alpha = 0$ is the special case of TTC without prices, and $\alpha = 2$ would set prices equal to the largest possible valuation of any agent. I select the value of α that maximizes total welfare on an independent set of valuations constructed in the same way and of the same size as the valuations that are used to evaluate performance. I perform the optimization of α separately for the heterogeneous economy and each of the homogeneous economies. The selected values of α are shown in Table 5.1.

Note that this cross-validation process might result in an over-estimation of the efficiency of TTC with fixed prices compared to its effectiveness in practice. In a real application, the true distribution of agent valuations with respect to hyper-parameters would not be known to the operator, so training data for cross-validation might not be available. The process also assumes that agent valuations do not change between the selection of α and the running of the mechanism. Nonetheless, because I claim that TTC's *inefficiency* is undesirable, this price generation process is sufficient.

5.2 Simulation Results

5.2.1 Efficiency Performance (TTC with Fixed Prices)

I first evaluate the performance of TTC with fixed prices. To match the statement of the conjecture, I average efficiency ratios across simulations that share hyper-parameters at each market size. From the 100 sample dataset, this results in ten sample mean efficiency ratios at each market size.

Figure 5.2 shows a error-bar plot of this statistic. The mean shown is the average of the sample means, with respect to each hyper-parameter, over the simulations; the bars show the standard error of this statistic. The figure provides support to the conjecture that TTC with fixed prices can achieve an ex-ante constant approximation of welfare, since we can see evidence of convergence to a roughly constant approximation of welfare in each type of economy. There is also evidence that the conjecture holds even when the economy has agents of mixed types.

Figure 5.2 also indicates that, as expected, including prices improves TTC's efficiency performance. We see that TTC with fixed prices achieves about 90 percent of optimal welfare on average from a relatively small market size. Whether or not this welfare approximation would be adequate depends on the application. I note as well that this welfare approximation is achieved with ex-

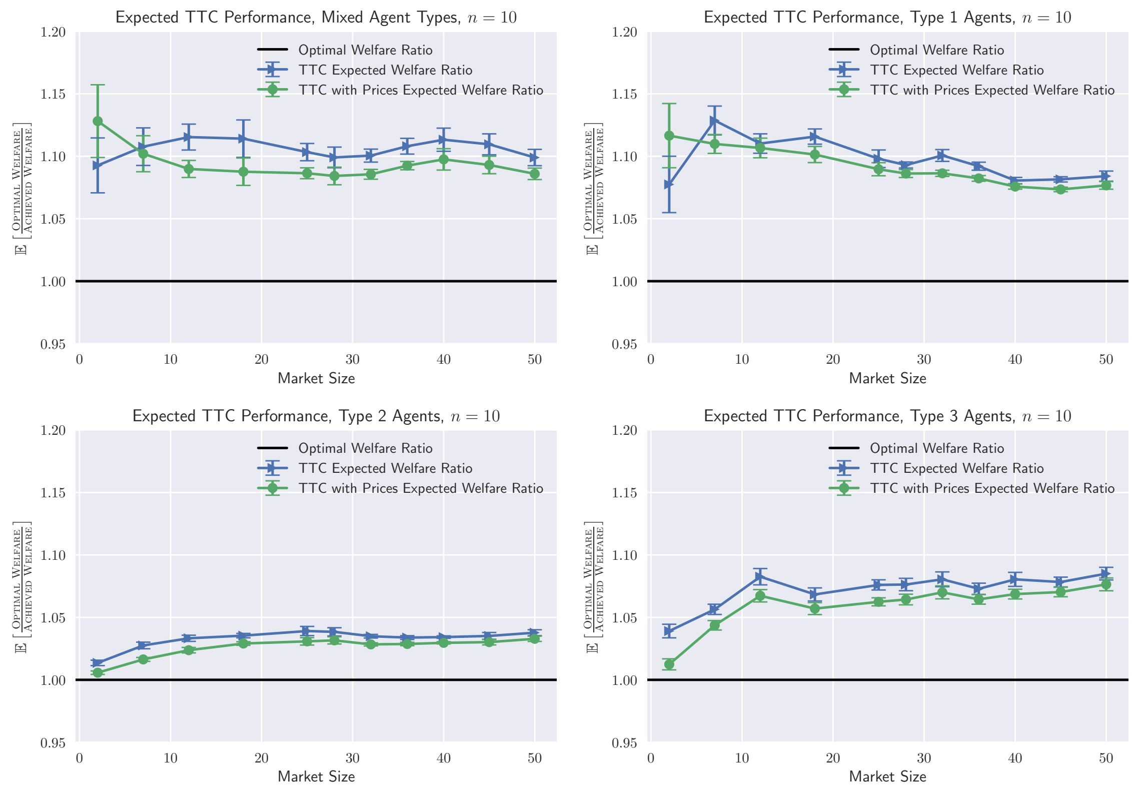


Figure 5.2: Ex-ante efficiency ratio for TTC with fixed prices. In each of the different types of economies, both TTC and TTC with fixed prices appear to achieve a constant approximation of efficiency in expectation over the agent valuation distributions.

actly known values of the hyper-parameters and agent valuations that are exactly independent and identically distributed with respect to the hyper-parameters.

These conditions may not hold in practice and warrant further investigation of TTC's performance, either under different simulated conditions, on real data, or in a lab experiment in vein of Guillen and Kesten (2008), which evaluated TTC on ordinal preferences [20].

5.2.2 Utility and Deficit (VCG)

Because agents make and receive different payments in each mechanism, we have to consider achieved utility, rather than achieved welfare, when comparing VCG, TTC, and NetAscendingClock. This is particularly important because VCG can run a deficit, thereby providing a higher total utility to the agents than the strictly budget-balanced TTC and NetAscendingClock mechanisms can.

The metric of interest here is utility per agent. In contrast to the evaluation of TTC's ex-ante efficiency, I regard each draw of valuations as a unique trial, rather than averaging over draws of the hyper-parameter, because we are now interested in ex-post utility, rather than ex-ante efficiency. For this simulation, then, we have $n = 100$ for each market size.

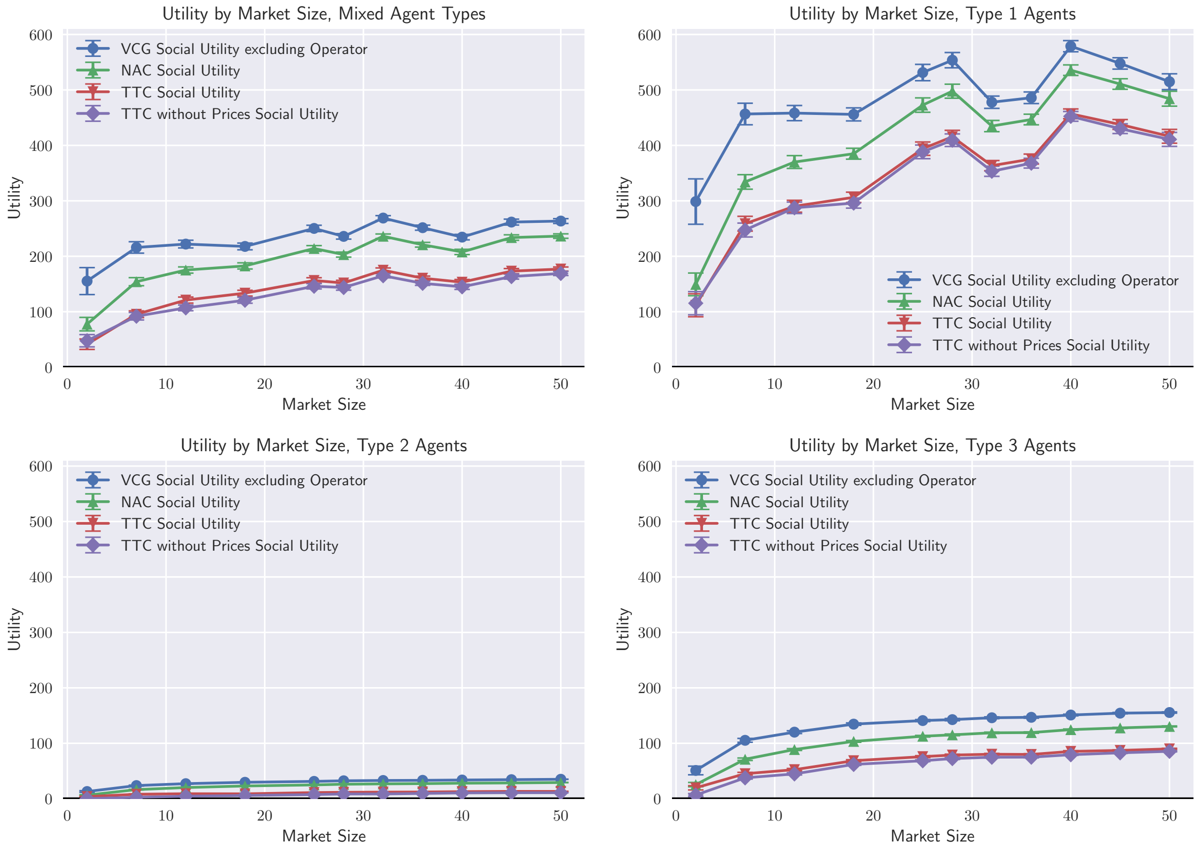


Figure 5.3: Utility per agent by market size for each type of economy. As we would expect, economies in which agents' valuations have higher variance have larger gains from trade.

The results of the simulations for utility per agent are shown in Figure 5.3. As we would expect to see, utility per agent grows roughly linearly in the size of the market under all three mechanisms. I observe the differences between VCG, TTC with and without prices, and NetAscendingClock. As expected, VCG achieves the highest utility per agent as long as the negative utility of the operator is excluded. If the operator's utility is also included, VCG and NetAscendingClock are identical because each is efficient and the sum of payments including the operator's cost is zero.

The more important observation is to note that higher variance distributions for agent valuations result larger gains in utility. Because NetAscendingClock is also exactly efficient, however, we can see in Figure 5.3 that this utility increase must come from corresponding growth in the VCG deficit. Figure 5.4 confirms this by showing that the Type 1 economy has the largest VCG budget deficits and that deficits increase in order of increasing variance in agent valuations.

Overall, this analysis indicates that VCG's theoretical budget imbalance cannot be ignored. This conclusion is particularly important when agents have potentially very different preferences for items from each other, since VCG requires a larger capital expenditure in that case.

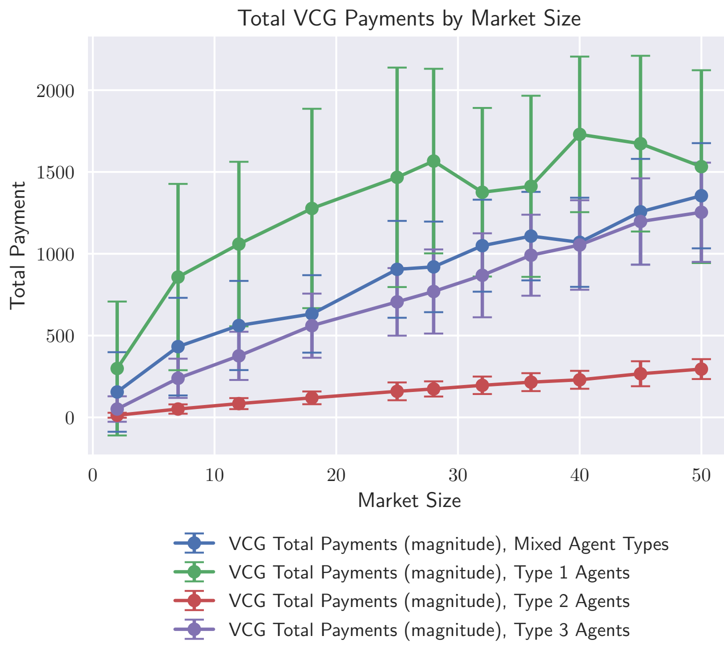


Figure 5.4: Consistent with the larger gains from trade in economies, VCG deficits also increase in economies with higher variance agent valuations.

5.2.3 Cycle Formation (All Mechanisms)

A core practical concern about the assignments chosen in a discrete exchange economy is the length of the resulting cycle of trades that have to take place. Longer chains require more logistical coordination to ensure that trades actually take place. This might make long cycles difficult to enforce in practice. Previous work on kidney exchange markets has noted both the practical challenges of long chains and the efficiency gains that allowing long chains can provide [8].

The simulation results show that a similar phenomenon underlies the efficiency of NetAscendingClock. Figure 5.5 depicts the average trade cycle length by market size for each of the economies. As the figure shows, the exactly efficient mechanisms select much longer chains of trades than does TTC with fixed prices.

The conclusion we should draw based on this result depends somewhat on the logistical details of the application. If trades are easy to enforce – for example, when assets are digital and therefore can be automatically transferred by the operator – long cycles might be unconcerning, especially when they enable exact efficiency.

Markets that would be easier to implement with short chains face a dual problem: while TTC with fixed prices might perform relatively efficiently and select short cycles, agents may be more likely to back out on trades because selected allocations are not envy-free. As a result, the envy-freeness of NetAscendingClock therefore might offer more stability than TTC with fixed prices

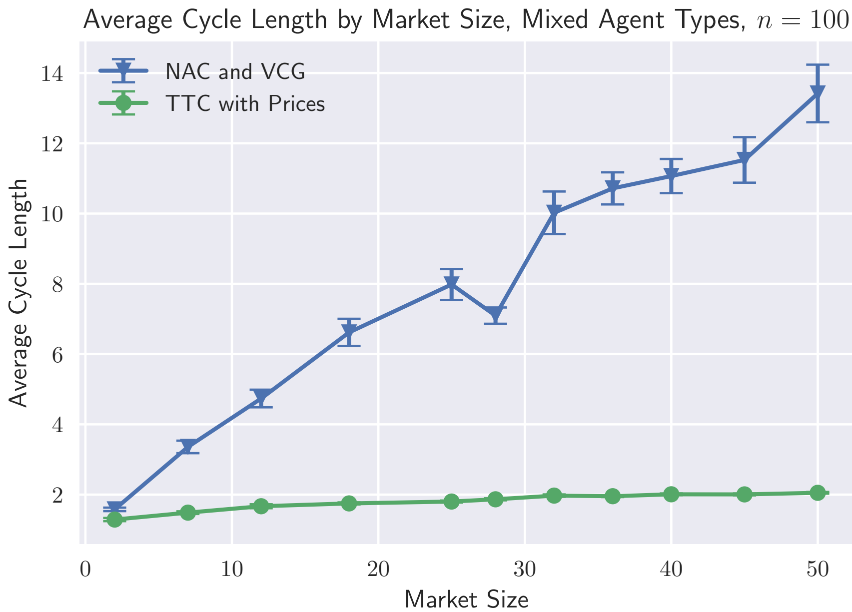


Figure 5.5: Average cycle length by market size. It is intuitive that expected average cycle length would grow with the size of the economy – VCG and NetAscendingClock each have this property, while TTC does not. Cycle length results for each type of economy are included in the appendix.

even through TTC selects shorter chains.

5.3 Summary

The simulation results in this chapter demonstrate that the theoretical flaws of VCG and TTC with fixed prices present practical barriers to their application in practice. VCG's budget deficit is particularly concerning. TTC with fixed prices, and even TTC without prices, could perform well and offer logistical benefits. However, NetAscendingClock resolves the shortcomings of each of these mechanisms. The simulation results indicate that NetAscendingClock's manipulability (even setting aside concepts of approximate strategy-proofness) might be acceptable if the details of an application make the application of VCG or TTC with prices impossible.

Chapter 6

Conclusion

Introducing money into discrete exchange models opens a broad class of compelling applications in which money would be both beneficial and expected. In this thesis, I have considered the theoretical and practical implications of including money in an exchange economy.

My work highlights the unavoidable tension between exact dominant strategy equilibrium and approximate efficiency. I extended the Myerson-Satterthwaite impossibility to show that approximate efficiency, individual rationality, budget-balance and strategy-proofness are mutually incompatible in a discrete exchange economy with money. Because individual rationality and budget-balance cannot be relaxed due to practical concerns about unraveling, I focused on the common practical goal of designing an intuitive mechanism with desirable computational, welfare, and fairness properties. While NetAscendingClock does not guarantee strategy-proofness, it represents a more compelling generalization of top trading cycles to economies with money than does a price-fixing approach. Furthermore, NetAscendingClock presents a novel connection between the literature on exchange economies and the literature on room-assignment rent-division and has the advantage of simplicity.

NetAscendingClock's manipulability offers the richest direction for future research. Formally quantifying the usefulness of strategic misreports would be an important next step. One approach would follow Andersson et al. (2013) to characterize mechanisms for the discrete exchange economy with money that are minimally manipulable across all possible envy-free mechanisms. Other options would be to measure NetAscendingClock's manipulability via quantitative metrics such as ex-post regret, as described by Parkes and Lubin [31], or to investigate whether NetAscendingClock satisfies a relaxed strategic property like Azevado and Budish (2012)'s strategy-proofness in the large [9].

A natural next step would be to extend the static discrete exchange setting to dynamic models where agents enter and exit the market over time. Recent work has generalized exchange economies without money into this setting, and a dynamic generalization with money would further lower the barrier to the application of a matching market with money in practice.

Appendix A

Additional Figures

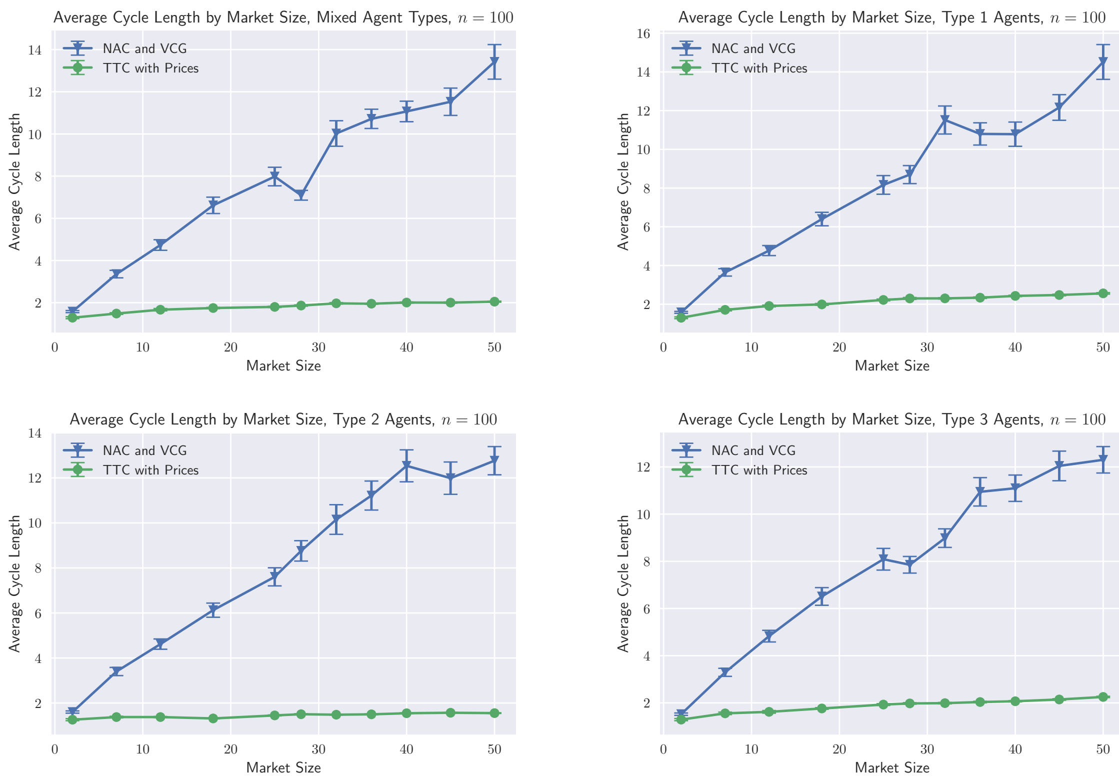


Figure A.1: Cycle lengths for each of composition of economy. We see that exact efficiency depends upon long cycles in each of the settings that are evaluated.

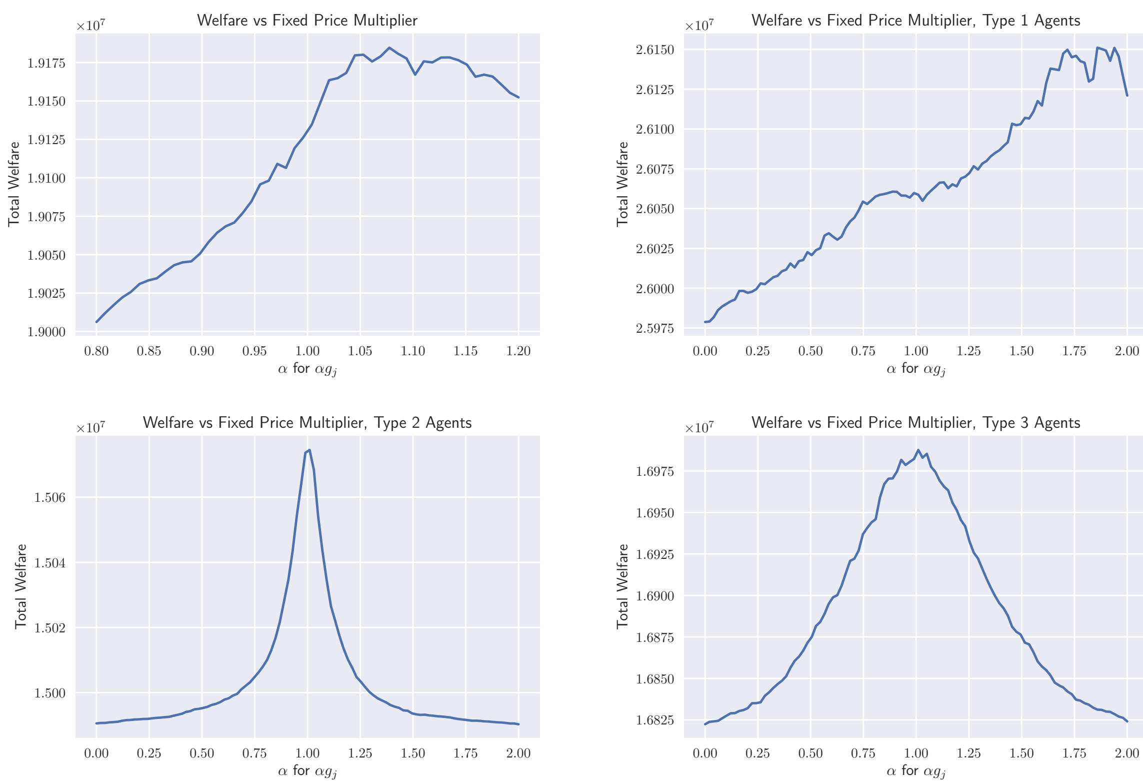


Figure A.2: Optimal welfare for varying values of α in cross-validation. Results indicate that higher prices are desirable when the probability of larger valuations increases.

Bibliography

[1] Abdulkadiroğlu, A., Sönmez, T., and Ünver, M. U. Room assignment-rent division: A market approach. *Social Choice and Welfare 22*, 3 (2004), 515–538.

[2] Abdulkadirolu, A., and Sönmez, T. House allocation with existing tenants. *Journal of Economic Theory 88*, 2 (1999), 233 – 260.

[3] Abdulkadirolu, A., and Sönmez, T. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica 66*, 3 (1998), 689–701.

[4] Adali, S., Sikdar, S., and Xia, L. Mechanism design for multi-type housing markets. *CoRR abs/1611.07636* (2016).

[5] Andersson, T., Andersson, C., Talman, A. J., and J. Sets in excess demand in simple ascending auctions with unit-demand bidders. *Annals of Operations Research 211*, 1 (12 2013), 27–36. Copyright - Springer Science+Business Media New York 2013; Document feature - ; Equations; Last updated - 2014-08-30.

[6] Andersson, T., Ehlers, L., and Svensson, L.-G. Budget balance, fairness, and minimal manipulability. *Theoretical Economics 9*, 3 (2014), 753–777.

[7] Andersson, T., Ehlers, L., and Svensson, L.-G. Transferring ownership of public housing to existing tenants: A market design approach. *Journal of Economic Theory 165* (2016), 643 – 671.

[8] Ashlagi, I., Gamarnik, D., Rees, M. A., and Roth, A. E. The need for (long) chains in kidney exchange, 07 2012. Copyright - Copyright National Bureau of Economic Research, Inc. Jul 2012; Document feature - Tables; ; Graphs; Equations; Last updated - 2015-12-18.

[9] Azevedo, E. M., and Budish, E. Strategyproofness in the large as a desideratum for market design. In *Proceedings of the 13th ACM Conference on Electronic Commerce* (New York, NY, USA, 2012), EC '12, ACM, pp. 55–55.

[10] Blumrosen, L., and Nisan, N. Combinatorial auctions. In *Algorithmic Game Theory*, N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, Eds. Cambridge University Press, Cambridge, 2007, ch. 11, pp. 267–298.

[11] Bredin, J., Parkes, D. C., and Duong, Q. Chain: A dynamic double auction framework for matching patient agents. *J. Artif. Int. Res. 30*, 1 (Sept. 2007), 133–179.

[12] Chu, L. Y., and Shen, Z.-J. M. Truthful double auction mechanisms. *Operations Research 56*, 1 (2008), 102–120.

[13] Clarke, E. H. Multipart pricing of public goods. *Public Choice 11* (1971), 17–33.

[14] Demange, G., Gale, D., and Sotomayor, M. Multi-item auctions. *Journal of Political Economy 94*, 4 (1986), 863–872.

[15] Gal, Y. K., Mash, M., Procaccia, A. D., and Zick, Y. Which is the fairest (rent division) of them all? *Journal of the ACM 10*, 10 (September 2016).

[16] Gale, D. Equilibrium in a discrete exchange economy with money. *International Journal of Game Theory 13*, 1 (1984), 61–64.

[17] Gibbard, A. Manipulation of voting schemes: A general result. *Econometrica 41*, 4 (1973), 587–601.

[18] Gonen, M., Gonen, R., and Pavlov, E. Generalized trade reduction mechanisms. In *Proceedings of the 8th ACM Conference on Electronic Commerce* (New York, NY, USA, 2007), EC '07, ACM, pp. 20–29.

[19] Groves, T. Incentives in teams. *Econometrica 41*, 4 (1973), 617–631.

[20] Guillen, P., and Kesten, O. Matching markets with mixed ownership: The case for a real-life assignment mechanism*. *International Economic Review 53*, 3 (2012), 1027–1046.

[21] Hall, P. On representatives of subsets. *Journal of the London Mathematical Society 10* (1935), 26–30.

[22] Kuhn, H. W. The hungarian method for the assignment problem. *Naval Research Logistics (NRL) 52*, 1 (2005), 7–21.

[23] McAfee, R. A dominant strategy double auction. *Journal of Economic Theory 56*, 2 (1992), 434 – 450.

[24] Mishra, D., and Talman, D. Overdemand and underdemand in economies with indivisible goods and unit demand, 2007. Copyright - Copyright FEDERAL RESERVE BANK OF ST LOUIS 2007; Last updated - 2015-10-03.

[25] Mishra, D., and Talman, D. Characterization of the walrasian equilibria of the assignment model. *Journal of Mathematical Economics 46*, 1 (2010), 6 – 20.

[26] Miyagawa, E. House allocation with transfers. *Journal of Economic Theory 100*, 2 (2001), 329 – 355.

[27] Myerson, R. B., and Satterthwaite, M. A. Efficient mechanisms for bilateral trading. *Journal of Economic Theory 29*, 2 (1983), 265 – 281.

[28] Niederle, M., Roth, A., and Sonmez, T. Matching and market design. In *The New Palgrave Dcitionary of Economics*, S. N. Durlauf and L. E. Blume, Eds. 2008.

[29] Nisan, N., Schapira, M., Valiant, G., and Zohar, A. Best-response auctions. In *Proceedings of the 12th ACM Conference on Electronic Commerce* (New York, NY, USA, 2011), EC '11, ACM, pp. 351–360.

[30] Parkes, D. C., Kalagnanam, J., and Eso, M. Achieving budget-balance with vickrey-based payment schemes in exchanges. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2* (San Francisco, CA, USA, 2001), IJCAI'01, Morgan Kaufmann Publishers Inc., pp. 1161–1168.

[31] Parkes, D. C., and Lubin, B. Approximate strategyproofness. *Current Science 103*, 9 (2012), 1021–1032.

[32] Parkes, D. C., and Ungar, L. H. Iterative combinatorial auctions: Theory and practice. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence* (2000), AAAI Press, pp. 74–81.

[33] Quinzii, M. Core and competitive equilibria with indivisibilities. *International Journal of Game Theory 13*, 1 (1984), 41–60.

[34] Roth, A. E. What have we learned from market design?*. *The Economic Journal 118*, 527 (2008), 285–310.

[35] Sankaran, J. K. On a dynamic auction mechanism for a bilateral assignment problem. *Mathematical Social Sciences 28*, 2 (1994), 143 – 150.

[36] Satterthwaite, M. A. Strategy-proofness and arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory 10*, 2 (1975), 187 – 217.

[37] Shapley, L., and Scarf, H. On cores and indivisibility. *Journal of Mathematical Economics 1*, 1 (1974), 23 – 37.

[38] Shapley, L. S., and Shubik, M. The assignment game i: The core. *International Journal of Game Theory 1*, 1 (1971), 111–130.

[39] Svensson, L.-G. Large indivisiblities: An analysis with respect to price equilibrium and fairness. *Econometrica* (1983).

[40] Vickrey, W. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance 16*, 1 (1961), 8–37.

[41] Wako, J. Strong core and competitive equilibria of an exchange market with indivisible goods. *International Economic Review 32*, 4 (1991), 843–852.