Integrating Market Makers, Limit Orders, and Continuous Trade in Prediction Markets

HODA HEIDARI, University of Pennsylvania
SÉBASTIEN LAHAIE, Microsoft Research
DAVID M. PENNOCK, Microsoft Research
JENNIFER WORTMAN VAUGHAN, Microsoft Research

We provide the first concrete algorithm for combining market makers and limit orders in a prediction market with continuous trade. Our mechanism is general enough to handle both bundle orders and arbitrary securities defined over combinatorial outcome spaces. We define the notion of an ϵ-fair trading path, a path in security space along which no order executes at a price more than ϵ above its limit, and every order executes when its market price falls more than ϵ below its limit. We show that under a certain supermodularity condition, a fair trading path exists for which the endpoint is efficient, but that under very general conditions, reaching an efficient endpoint via an ϵ-fair trading path is not possible. We develop an algorithm for operating a continuous market maker with limit orders that respects the ϵ-fairness conditions in the general case in which the supermodularity condition may not hold. We conduct simulations of our algorithm using real combinatorial predictions made during the 2008 U.S. Presidential election and evaluate it against a natural baseline according to trading volume, social welfare, and violations of the two fairness conditions.

Additional Key Words and Phrases: Prediction markets; Limit order books; Automated market making

1. INTRODUCTION

Buying a security is a statement that, in a trader's view, the security is underpriced compared with its expected payoff. This statement is not cheap talk; if the trader is wrong, she stands to lose money. A prediction market aggregates many statements of this form about securities whose payoffs correspond to events we would like to predict, such as election outcomes or product sales. With sufficient activity, prediction markets often outperform competing prediction methods [Berg et al. 2008]. Moreover, continuous-trade markets (think of the stock market) provide real-time predictions that react with remarkable speed to news and information, such as a candidate's poor debate performance in an election race.

Prediction markets rely on trade. Without liquidity, a market faces a serious chicken-and-egg problem: a lack of trading opportunities discourages traders from participating, which in turn reduces future trading opportunities. For this reason, many predic-

tion market mechanisms offer a subsidy to jump-start trade by providing incentives for the first trader to catalyze the process. A common form of subsidy involves an automated market maker built into the market institution. The marker maker provides guaranteed liquidity, offering to buy or sell any security at any time, at some changing price. For prediction markets, the standard market maker operates by keeping track of a *cost function* that defines how much traders must pay to move the market from one state (vector of security quantities) to another. The cost function approach can be used to ensure that the market marker's worst-case loss is bounded, that trading is path independent, and that traders are unable to benefit from arbitrage [Hanson 2003b; Chen and Pennock 2007; Abernethy et al. 2013].

Even with a market maker, a trader may not see favorable transactions at current prices. If so, she may want to place a *limit order*, an offer to buy a security at a price equal to or less than a specified threshold. A limit order allows a trader to say "I won't pay the current price, but if it drops to ℓ or below, I want to buy." Limit orders are both a convenience, freeing traders from constantly monitoring for price changes, and a communication channel for traders, adding to the market's liquidity.

In this work, we provide and analyze a means of integrating limit orders into the cost function framework, which to date only accommodates market orders. We are concerned with *continuous markets*, in which trade orders may be placed, matched, and executed at any time, as opposed to *call markets*, in which orders are collected and executed at discrete, pre-specified times (e.g., once per hour). A continuous market maintains an order book of standing orders, waiting to be matched with arriving orders. It provides a complete, auditable record of trade execution over time. Both continuous and call markets present advantages, but the convenience of anytime trading has made the former the default in financial markets [Harris 2002, Chapter 5].

In Section 2, we review the concepts of cost functions and limit orders, and describe the convex programming framework of Agrawal et al. [2011] for running call prediction markets. Their framework provides a building block for the continuous trade mechanism developed in this paper. Our approach also links with ideas of Hanson [2003a], who describes a method for continuously executing limit orders in the presence of a market scoring rule. In Section 3, we formalize Hanson's intuition by defining the notion of an ϵ-fair trading path: a path in security space along which no order executes at a price more than ϵ above its limit, and any order executes when its market price falls more than ϵ below its limit.

In Section 4, we show that under a supermodularity condition, a fair trading path exists for which the endpoint is efficient, in the sense of maximizing social welfare or gains from trade. The proof is constructive. Supermodularity holds, for example, for a standard market maker defined over disjoint securities, with limit orders allowed only on single securities. However, even a standard market maker may fail to satisfy the condition if bundle orders are allowed, or if the market is incomplete. We show that, under very general conditions, there exist market states and orders such that reaching an efficient endpoint via a fair trading path is not possible.

In Section 5, we develop the Fair Trading Path Algorithm for the general case (with or without supermodularity) for operating a continuous market maker with limit orders respecting the ϵ-fair bounds along the way. The algorithm iteratively solves for an efficient order fill of maximal volume, but limits the quantity of each order executed per step in order to stay on an approximately fair trading path and terminate at a state in which no standing limit order is profitable.

In Section 6, we conduct simulations of our algorithm in two settings: (1) a negative example in which no efficient fair trading path exists, and (2) an example constructed using combinatorial predictions made during the 2008 U.S. Presidential election. We

evaluate our algorithm against a natural baseline according to trading volume, social welfare, and violations of the two fairness conditions. Section 7 concludes.

Related Work

Our contribution can be viewed from a few different perspectives with respect to the literature. The one we emphasize is the integration of limit orders and cost functions, related to ideas outlined in a note by Hanson [2003a]. Our contribution can also be viewed as a continuous-market counterpart to the call market of Agrawal et al. [2011].

Call prediction markets have long provided the ability to place limit orders. Building on the classic work of Shapley and Shubik [1977], Lange and Economides [2005] derive a pari-mutuel call market that accommodates both limit orders and bundle orders for binary-payoff securities. Bossaerts et al. [2002] design an exchange mechanism called *combined value trading* that allows bundle orders and clears the market using linear programming, showing in laboratory experiments that the mechanism facilitates trade in thin markets. Fortnow et al. [2005] also use linear programming as the matching engine for a combinatorial call market for compound securities defined as arbitrary Boolean functions of an underlying state space.

In two remarkable papers, Peters et al. [2006] and Agrawal et al. [2011] provide a convex programming framework that subsumes much of this previous work, which they call the *convex pari-mutuel call auction mechanism*. The framework yields an efficient implementation of the Lange and Economides [2005] mechanism, and also introduces a market maker into the combined value trading LP. It is interesting to note that, in the evolution of call markets, limit orders were accommodated early on while market makers were only introduced more recently.

The reverse holds for the development of prediction markets with continuous trade. There we begin with the *market scoring rules* introduced by Hanson [2003b], and their equivalent cost function versions derived by Chen and Pennock [2007]. Abernethy et al. [2013] extend the cost function framework to arbitrary-payoff securities, but still allow solely market orders. Recently, Chakraborty et al. [2015] have drawn on Hanson's ideas to integrate limit orders and cost functions in a market for a single security, empirically evaluating the mechanism with an emphasis on its price discovery properties.

2. PRELIMINARIES

We begin with a review of cost-function-based market making. We then introduce limit orders and explain how they are handled in call markets, specifically the convex programming framework of Agrawal et al. [2011].

2.1. Cost Functions

Let $\Omega = \{\omega_1, \ldots, \omega_N\}$ be a set of mutually exclusive and exhaustive states of the world. We consider a market that offers a fixed menu of K securities defined over these states. The payoffs of these securities are described by an arbitrary function $\phi : \Omega \to \mathbf{R}^K$, with $\phi_j(\omega)$ specifying the value of security j if the final outcome is $\omega \in \Omega$.

The cost function framework [Chen and Pennock 2007; Abernethy et al. 2013] consists of a continuous market where traders arrive sequentially and request bundles of shares from a centralized market maker. The model only accommodates market orders, though traders may query the cost of any bundle before placing orders. The market maker is always willing to buy or sell arbitrary security bundles $\boldsymbol{\theta} \in \mathbf{R}^K$, where θ_j denotes the number of shares of security j. The market maker charges traders according to a convex potential function C called the *cost function*. Let $\mathbf{q}$ be the current market state, with q_j denoting the number of shares of security j that have been bought from the market maker so far. A trader who requests bundle $\boldsymbol{\theta}$ is charged $C(\mathbf{q} + \boldsymbol{\theta}) - C(\mathbf{q})$. The instantaneous price of security j is then $p_j(\mathbf{q}) = \partial C(\mathbf{q})/\partial q_j$.

Choosing the cost function C fully determines the way in which market prices are set. Abernethy et al. [2013] give necessary and sufficient conditions on C for the resulting market to satisfy a set of nice properties including bounded loss for the market maker and a lack of arbitrage opportunities for traders. In particular, they show that C must be convex and that the derived price function $\mathbf{p}$ must map to the convex hull of the space $\phi(\Omega)$, the image of the payoff function. They additionally show that any cost function of this form can be written as

$$C(\mathbf{q}) = \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \mathbf{p}^\top \mathbf{q} - R(\mathbf{p}) \tag{1}$$

for a strictly convex function R, where $\mathcal{H}$ denotes the convex hull. Furthermore, assuming R is continuous and defined over $\mathcal{H}(\phi(\Omega))$, the instantaneous price vector satisfies

$$\mathbf{p}(\mathbf{q}) = \nabla C(\mathbf{q}) = \arg \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \mathbf{p}^\top \mathbf{q} - R(\mathbf{p}). \tag{2}$$

We assume throughout the paper that the cost function takes this form, with an additional restriction on R. Abernethy et al. [2013] call R a *pseudo-barrier* function if $\lim_{t \to \infty} \|\nabla R(\mathbf{p}^t)\| = \infty$ for any convergent sequence $\{\mathbf{p}^t\}$ of points in $\mathcal{H}(\phi(\Omega))$ whose limit is in the relative boundary of $\mathcal{H}(\phi(\omega))$. Throughout the paper, we assume that R is a pseudo-barrier and bounded on $\mathcal{H}(\phi(\omega))$. In particular, our results use the following lemma, which may be of independent interest. It shows that there is a fixed subspace over which a cost function obtained from a pseudo-barrier is linear at every point. The proof is in the appendix.[1]

LEMMA 2.1. *Let C be a cost function defined via (1) with R a strictly convex pseudo-barrier function. If there exist bundles $\mathbf{q}, \mathbf{r} \in \mathbf{R}^K$ and a constant $\gamma \in \mathbf{R}$ such that $\gamma = C(\mathbf{q}+\mathbf{r}) - C(\mathbf{q}) = C(\mathbf{q}) - C(\mathbf{q}-\mathbf{r})$, then for all $\mathbf{q}' \in \mathbf{R}^K$ and all $\lambda \in \mathbf{R}$, $C(\mathbf{q}' + \lambda \mathbf{r}) - C(\mathbf{q}') = \lambda \gamma$. Additionally, for all $\omega \in \Omega$, $\phi(\omega)^\top \mathbf{r} = \gamma$.*

One commonly used instance of a cost function market is the logarithmic market scoring rule (LMSR) [Hanson 2003b, 2007]. The LMSR is defined for complete markets and uses Arrow-Debreu securities: $K = N$, and for each $j = 1, \cdots, N$ the payoff is $\phi_j(\omega) = 1$ if $\omega = \omega_j$ and 0 otherwise. The LMSR uses cost function $C(\mathbf{q}) = b \log \sum_{k=1}^N e^{q_k/b}$, resulting in instantaneous prices $p_j(\mathbf{q}) = e^{q_j/b} / \sum_{k=1}^N e^{q_k/b}$. Here $b > 0$ is a liquidity parameter controlling the trade-off between the rate at which prices change with trades and the market maker's worst-case loss. Lemma 2.1 applies to the LMSR because it is indeed based on a pseudo-barrier function (negative entropy) over the simplex of probability distributions. In this case the lemma establishes the well-known fact that the LMSR cost function is linear along the $\mathbf{1}$ vector at every market state.

2.2. Limit Orders

In our setting, traders may place either market orders or limit orders. A limit order is specified by a triple $(\boldsymbol{\theta}, n, \ell)$. Here $\boldsymbol{\theta} \in \mathbf{R}^K$ is a security bundle, n is the number of shares of this bundle that the trader would like to purchase, and ℓ is the maximum price the trader is willing to pay per share. It is convenient to assume that $\boldsymbol{\theta}$ is normalized with either $\|\boldsymbol{\theta}\|_1 = 1$ or $\|\boldsymbol{\theta}\|_\infty = 1$, but this assumption is without loss of generality and our results do not depend on any specific normalization. Market orders can be represented as limit orders with $\ell = \infty$ (or the maximum payoff of bundle $\boldsymbol{\theta}$), so we may assume that all orders are limit orders. We allow for limit orders to be partially executed. That is, the trader may be sold x shares of the bundle $\boldsymbol{\theta}$ for any $0 \le x \le n$.

---

[1]The appendix appears in the long version of this paper available on the authors' websites.

A limit-order market maintains an order book (a set of limit orders) to keep track of orders that have not yet been fully executed. In a call market, a new order book is provided at each trading session. In a continuous market, the order book contains orders that have accumulated over time without fully executing. When describing the behavior of a continuous market, we consider a single point in time in which the order book contains m existing orders $(\boldsymbol{\theta}_i, n_i, \ell_i)$ for $i = 1, \ldots, m$, and a new trader arrives with order $(\boldsymbol{\theta}_0, n_0, \ell_0)$. We record the orders in the matrix $\boldsymbol{\Theta} = [\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_m]$, where each bundle is a column of the matrix, and in the (column) vectors $\mathbf{n} = [n_0, \ldots, n_m]$ and $\boldsymbol{\ell} = [\ell_0, \ldots, \ell_m]$. Given an order book $(\boldsymbol{\Theta}, \mathbf{n}, \boldsymbol{\ell})$, an *order fill* $\mathbf{x} \in \mathbf{R}^{m+1}$ denotes the number of units of each order that are executed. An order fill is *feasible* if $\mathbf{0} \leq \mathbf{x} \leq \mathbf{n}$. We will refer to the 1-norm $\|\mathbf{x}\|_1$ of a feasible order fill as its *volume*. In the cost function framework, the instantaneous price of order i after executing fill $\mathbf{x}$, starting from initial market state $\mathbf{q}$, is given by

$$\pi_i(\mathbf{x}) \equiv \boldsymbol{\theta}_i^\top \mathbf{p}(\boldsymbol{\Theta}\mathbf{x} + \mathbf{q}) = \boldsymbol{\theta}_i^\top \nabla C(\boldsymbol{\Theta}\mathbf{x} + \mathbf{q}).$$

These instantaneous *order* prices should not be confused with the instantaneous *security* prices, which recall are given by $p_j(\mathbf{q}) = \partial C(\mathbf{q})/\partial q_j$ for each security $j = 1, \ldots, K$. Instantaneous order prices are in fact a linear combination of these instantaneous security prices at the market state $\boldsymbol{\Theta}\mathbf{x} + \mathbf{q}$.

The various call markets in the literature all attempt to find an order fill $\mathbf{x}$ that controls the eventual payout $\phi(\omega)^\top \boldsymbol{\Theta}\mathbf{x} = \sum_{i=0}^{m} x_i \boldsymbol{\theta}_i^\top \phi(\omega)$ across states of the world $\omega \in \Omega$. For instance, one might bound the maximum payout across states to be non-positive (so that there is no risk of loss), or no more than some fixed subsidy. If feasible, one might constrain the payout to always be zero. Various objectives can be layered on top of the constraints to select a particular order fill: maximizing volume, maximizing value to the traders (according to their limit prices), and so on [Rothschild and Pennock 2014]. Agrawal et al. [2011] propose a sophisticated convex program where the eventual payout is controlled by both a subsidy and a loss function over any amounts the subsidy does not cover. For our purposes, it is enough to state that their program is, as they prove, equivalent to

$$\max_{\mathbf{x}} \ \boldsymbol{\ell}^\top \mathbf{x} - C(\boldsymbol{\Theta}\mathbf{x} + \mathbf{q}) + C(\mathbf{q}) \tag{3}$$

$$\text{s.t.} \qquad \mathbf{0} \leq \mathbf{x} \leq \mathbf{n} \tag{4}$$

where C is a cost function of the form (1) and $\mathbf{q}$ is the current market state. We should note that the framework of Agrawal et al. [2011] is cast in terms of a complete market with Arrow-Debreu securities. We believe their framework extends to securities with general payoffs, but some of their results, such as the equivalence just mentioned, would require additional conditions. For our purposes this is immaterial because we build directly on top of the cost function framework.

One interpretation of the objective function (3) is as follows. If we take the limit prices as the traders' values per share, the first term in the objective captures the value of the order fill to the traders. Drawing on the known mathematical equivalence between cost functions and the convex *risk measures* that are used to quantify risk in mathematical finance [Föllmer and Schied 2002; Othman and Sandholm 2011], we can interpret $C(\boldsymbol{\Theta}\mathbf{x} + \mathbf{q}) - C(\mathbf{q})$ as the change in the market maker's risk after filling the order $\boldsymbol{\Theta}\mathbf{x}$. The objective is therefore the value to the traders net of the risk incurred by the market maker, as quantified by its cost function. We focus on a second interpretation of this objective as the overall social welfare of traders in Section 3.1.

While the convex program prescribes an order fill, there are still several possible pricing schemes in a call market. The most standard is to use market clearing prices, which are formally obtained as the dual optimal solution to the program. Another

possibility is to charge the limit prices (i.e., a first-price auction). The VCG payments in such an auction also have connections to cost-function payments [Agrawal et al. 2011]. We will return to this efficiency-maximizing program as it will form the basis of our continuous market, and we will see that continuous order execution pins down a pricing scheme that is a hybrid of limit and cost function pricing.

3. CONTINUOUS ORDER EXECUTION

Our interest is in algorithms that allow the integration of limit orders with cost-function based market making. More than a decade ago, before the cost function framework was well understood, Hanson wrote a brief note outlining an intuitively correct way of integrating limit orders with market scoring rules such as the LMSR [Hanson 2003a]. He described the following process (our notation in brackets replaces his):

> "Each book order with a limit $[\ell_i]$ imposes a constraint on the market maker prices $[\pi_i]$. It says that until that order expires, or all its quantity is used, the prices must satisfy $[\ell_i \leq \pi_i]$. This constraint is binding on a particular plane in the price vector space. Thus the above description of a new order trading with the market maker is only valid until the price vector reaches one of these planes. At that point the new order trades with both the market maker and the book order at the same time, moving the market maker prices in the plane of the book order. This continues until a price or quantity limit of one of the orders is reached, or until another book order plane is reached."

This idea can be illustrated through a simple example. Consider a complete market with two states and two Arrow-Debreu securities. Suppose trader 1 enters the market and submits a limit order $(\boldsymbol{\theta}_1 = (1, 0), n_1, \ell_1)$ on the first security. Suppose that the current market state $\mathbf{q}$ is such that the trade cannot be executed: $\ell_1 \leq p_1(\mathbf{q})$. The order would then go into the book. Now suppose trader 0 arrives and places a market order $(\boldsymbol{\theta}_0 = (0, 1), n_0, \ell_0 = \infty)$ on the second security. Following the intuition in Hanson's argument, the market maker would begin to execute this new order. Let τ be the quantity of the second security that could be traded before trader 1's limit price is hit, so $p_1(\mathbf{q} + \tau \boldsymbol{\theta}_0) = \ell_1$. If $\tau \geq n_0$, then the market order can be fully executed. Otherwise, the market maker would execute the first τ shares. At this point, the two traders would effectively "trade with each other" at the current market price. Since the market is complete, with any cost function from the class we consider, purchasing a bundle containing one of each security always costs \$1 and does not affect the instantaneous market prices [Chen and Pennock 2007; Abernethy et al. 2013]. Therefore, to "trade with each other," we can think of the traders purchasing $\min(n_0 - \tau, n_1)$ shares of the bundle $(1, 1)$ from the market maker and splitting them up so that trader 1 receives the shares of the first security at a cost of ℓ_1 per share and trader 0 receives the shares of the second at a cost of $1 - \ell_1$. If trader 0's order has not been fully executed, the trader may continue to trade with the market maker at this point.

If we alter this example so that there is a third security on a third state available, it already becomes trickier to reason about what should happen. Purchasing the bundle $(1, 1, 0)$ does not keep the price of the first security constant, complicating the question of how the traders should "trade with each other" once trader 1's limit price is hit. Matters get even more complex when we move to markets with larger state spaces, arbitrary securities for sale, and more traders. Hanson [2003a] does not provide an algorithm for determining which trades should be executed, only stating that iterative numerical methods would probably be required.

However, this example does provide principles that a market maker should strive to achieve when implementing an order book on top of a cost-function based market. In particular, for any order fill, there should exist a corresponding continuous execution

path such that for any intermediate order fill $\mathbf{x}'$ along that path, the orders in the book should be respected. This means that at any point $\mathbf{x}'$ along the path, if an order $(\boldsymbol{\theta}, n, \ell)$ has not yet been fully executed, we should have $\ell_i \leq \pi_i(\mathbf{x}')$; the book should not be crossed. Additionally, if the order is currently being executed at the point $\mathbf{x}'$, it should not be executed at a price higher than its limit price; for such $\mathbf{x}'$, we must have $\ell_i \geq \pi_i(\mathbf{x}')$. In the next section, we formalize these criteria.

3.1. Fair Trading Paths

We now formalize this intuition, leading to output criteria for an algorithm implementing limit orders on top of a cost-function based market maker.

Definition 3.1 (*Trading path*). A *trading path* for order book $(\boldsymbol{\Theta}, \mathbf{n}, \boldsymbol{\ell})$ at initial market state $\mathbf{q}$ is a monotonically non-decreasing, continuous mapping $\mathcal{P} : [0, 1] \to \mathbf{R}^{m+1}$ with $\mathcal{P}(0) = \mathbf{0}$ and $\mathcal{P}(1) \leq \mathbf{n}$.

A trading path describes the evolution of order execution. Note that only the image of the path matters—paths with the same image are equivalent. The domain is chosen as $[0, 1]$ only as a normalization. We next define an (approximately) fair trading path.

Definition 3.2 (*Fair trading path*). A trading path $\mathcal{P}$ for orders $(\boldsymbol{\Theta}, \mathbf{n}, \boldsymbol{\ell})$ at initial market state $\mathbf{q}$ is an ϵ-*fair trading path* if

(1) For all $i \in \{0, \cdots, m\}$ and any a and b such that $0 \leq a < b \leq 1$, if $\mathcal{P}_i$ is strictly increasing over $[a, b]$, then for all $t \in [a, b]$
$$\pi_i(\mathcal{P}(t)) \leq \ell_i + \epsilon. \tag{5}$$
(2) For all $i \in \{1, \cdots, m\}$, for any $t \in [0, 1]$ such that $\mathcal{P}_i(t) < n_i$,
$$\pi_i(\mathcal{P}(t)) \geq \ell_i - \epsilon. \tag{6}$$

If these conditions hold with $\epsilon = 0$, $\mathcal{P}$ is called a *fair trading path*.

The first condition captures the requirement that no trader should pay (too much) more than his limit price. The second captures the requirement that the market price of any unfilled order in the book can never fall (too far) below its limit price. Note that the second condition only applies to existing orders in the book. This reflects the pricing convention in continuous markets, which says that existing orders trade at their limit price while arriving orders trade at the best available price.

   Let $\bar{\mathbf{x}} = \mathcal{P}(1)$ be the endpoint of a fair trading path. By integrating and aggregating order prices along the path, it is straightforward to show that the market maker collects a revenue of $C(\boldsymbol{\Theta}\bar{\mathbf{x}} + \mathbf{q}) - C(\mathbf{q})$. This means that charging according to a fair trading path respects the principle that a cost function should reflect the revenue collected, assuming $C(\mathbf{0}) = 0$ [Chen and Pennock 2007]. From this viewpoint, the objective (3) has a simpler interpretation in terms of efficiency that does not use the notion of risk measure: the objective is the total utility to the traders (i.e., social welfare), where utility is the reported value (limit price) of a trade minus its cost as given by the cost function.

   Integrating instantaneous prices along a fair trading path yields a unique, well-defined way to price orders. Note that when $\epsilon = 0$, the conditions together imply that orders in the book (i.e., excluding the new order) always execute exactly at their limit price. Thus each order $i \neq 0$ is charged $\ell_i \bar{x}_i$. Meanwhile, the new order 0 is charged

$$C(\boldsymbol{\Theta}\bar{\mathbf{x}} + \mathbf{q}) - C(\mathbf{q}) - \sum_{i \neq 0} \ell_i \bar{x}_i. \tag{7}$$

In some circumstances it may be unreasonable to charge a trader even a small amount over his limit price, in which case condition (5) may be too weak of a requirement with

$\epsilon > 0$. In practice, even when $\epsilon > 0$, a market maker could adopt the convention of charging exactly the limit price for orders in the book and charging the new order as in (7) as long as this does not exceed the new order's limit.

While the notion of a fair trading path yields a pricing scheme, note that the trivial trading path that does not execute any orders satisfies our definition. To obtain more interesting paths, we impose requirements on the endpoint.

3.2. Endpoint Criteria

We now derive conditions that should naturally signal the end of trading. In the process, we obtain further insights into the fair trading path conditions, and how one might construct a path satisfying them. A natural condition to impose is that the endpoint should be efficient (maximize reported social welfare). Let

$$F(\mathbf{x}; \mathbf{q}) \equiv \boldsymbol{\ell}^\top \mathbf{x} - C(\boldsymbol{\Theta}\mathbf{x} + \mathbf{q}) + C(\mathbf{q}) \tag{8}$$

be the objective function capturing the efficiency of $\mathbf{x}$. (For the sake of clarity we occasionally suppress the parameter $\mathbf{q}$.) Recall the convex program of Agrawal et al. [2011] introduced earlier:

$$\max_{\mathbf{x}} \quad F(\mathbf{x}; \mathbf{q}) \tag{9}$$

$$\text{s.t.} \ \ 0 \le x_i \le n_i \ \ (i = 0, \dots, m) \tag{10}$$

Here order 0 is the arriving order and orders $1, \dots, m$ are in the book before arrival (either completely unfilled or partially filled with n_i shares remaining). This is a straight-forward convex program with box constraints. We say that a feasible order fill is *efficient* at state $\mathbf{q}$ if it maximizes this convex program. If $\mathbf{x}^*$ is an optimal (i.e., efficient) solution, then the optimality conditions for such programs [Boyd and Vandenberghe 2009, p. 142] state that, for each order $i = 0, 1, \dots, m$,

$$x_i^* > 0 \implies \pi_i(\mathbf{x}^*) \le \ell_i, \tag{11}$$

$$x_i^* < n_i \implies \pi_i(\mathbf{x}^*) \ge \ell_i. \tag{12}$$

See the appendix for a detailed derivation. Note that these conditions are closely related to conditions (5–6) in the definition of a fair trading path. For instance, if order i is partially filled at the optimum, so that $0 < x_i^* < n_i$, then $\pi_i(\mathbf{x}^*) = \ell_i$. Any further infinitesimal trade of the order would occur at its limit price, respecting the pricing scheme derived in the previous section.

However, we will see that efficiency is too stringent of an endpoint criterion. The reason is that fair trading paths must be monotone—orders cannot be reversed—and there may not exist a monotone path to an efficient solution respecting (5–6) for $\epsilon = 0$. Returning to the optimality conditions, (12) arises because it should not be possible to improve efficiency by further trading order i. Condition (11) arises because reversing order i should also not improve efficiency. But since we are enforcing monotonicity in the path, and orders cannot be reversed, the latter should not apply. Thus we say that an order fill is *complete* if for each order $i = 0, \dots, m$ condition (12) holds. An efficient order fill is always complete, but not vice-versa. We will use the following equivalent characterization of completeness.

Definition 3.3 (*Complete order fill*). An order fill $\mathbf{x}$ is *(maximally) complete* if $\mathbf{0}$ is a (unique) efficient fill at market state $\mathbf{q} + \boldsymbol{\Theta}\mathbf{x}$ with quantities $\mathbf{n} - \mathbf{x}$.

In more intuitive terms, an order fill is complete if, after executing it, it is efficient to no longer trade, and maximally complete if any further trade strictly reduces efficiency. The concept of a maximally complete order fill will prove important later for algorithmic correctness. Note that F is not strictly concave, because cost functions are

not strictly convex. (For instance, the LMSR is linear at any point along the **1** direction.) Thus there may be multiple efficient order fills, or multiple complete order fills, and maximality further selects among them.

4. EFFICIENT TRADING PATHS

In this section we study conditions under which there exists a fair trading path that reaches an efficient order fill. The close connection between conditions (5–6) and conditions (11–12) suggests a few ideas for constructing such a path. First, one could simply take an efficient fill $\mathbf{x}^*$ and use a straight line from $\mathbf{0}$ to $\mathbf{x}^*$. While the fair trading path conditions would be met at the endpoint, and the path is monotone, there is no reason for the limit prices of orders in the book to match the instantaneous order prices at which they execute along the way. Alternatively, one could trace out a path by gradually filling order 0. More precisely, consider the correspondence

$$S(\alpha) = \arg\max_{\mathbf{x}}\{F(\mathbf{x}) \mid \mathbf{0} \leq \mathbf{x} \leq \mathbf{n}, 0 \leq x_0 \leq \alpha n_0\}, \tag{13}$$

where $\alpha \in [0, 1]$. If the correspondence admits a continuous, monotone selection, then by the optimality conditions for our convex program the result will be a fair trading path, and also an efficient path by construction. This section formally investigates whether and under what conditions such a construction is possible.

For the remainder of the paper, we will make the following standing assumption.

ASSUMPTION 1. *Before arrival of the new order, the previous order fill is maximally complete. Equivalently,* $\mathbf{0}$ *is the unique element in* $S(0)$.

The condition holds trivially if the order book is initially empty. Otherwise, one could start the continuous market with the call auction of Agrawal et al. [2011], ensuring that a solution of maximal volume is chosen. In fact, it is common practice in financial markets to initialize a continuous market via a call auction [Harris 2002, Chapter 5]. Our constructions in this section and the next will yield maximally complete fills, so that Assumption 1 continues to hold for all future arrivals.

4.1. Existence under Supermodularity

We show that a fair trading path reaching an efficient order fill exists when the objective function F is *supermodular*. Let $\mathbf{x} \wedge \mathbf{y}$ denote the component-wise minimum of vectors $\mathbf{x}$ and $\mathbf{y}$, and let $\mathbf{x} \vee \mathbf{y}$ denote their component-wise maximum. Note that for any two vectors $\mathbf{x}$ and $\mathbf{y}$ satisfying constraints (10), both $\mathbf{x} \wedge \mathbf{y}$ and $\mathbf{x} \vee \mathbf{y}$ are also feasible. Supermodularity is defined as follows.

Definition 4.1 (*Supermodularity*). A function $f : \mathbf{R}^n \to \mathbf{R}$ is *supermodular* if $f(\mathbf{x}) + f(\mathbf{y}) \leq f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y})$. A function f is *submodular* if $-f$ is supermodular.

Note that supermodularity of the objective F depends on the cost function C but also the order bundles Θ. It is not sufficient for C to be submodular (but nor is it necessary). However, as an important special case, we have that F is supermodular if C is submodular and the bundles in Θ do not overlap, where by "overlap" we mean intersect when neither is a subset of the other. For instance, we might use the LMSR, which is submodular, and restrict limit orders to singleton securities.

Supermodularity of F gives us the following result about S, the proof of which uses similar ideas as the proof of Topkis' Monotonicity Theorem; see Topkis [1978] or the excellent notes of Featherstone [2008].

LEMMA 4.2. *Let* F *be supermodular, and let* S *be defined as in (13). For any* $\alpha, \alpha' \in [0, 1]$ *with* $\alpha \leq \alpha'$, *for any* $\mathbf{x} \in S(\alpha)$ *and* $\mathbf{x}' \in S(\alpha')$, $\mathbf{x} \wedge \mathbf{x}' \in S(\alpha)$ *and* $\mathbf{x} \vee \mathbf{x}' \in S(\alpha')$.

PROOF. For $\alpha \in [0,1]$, define $D(\alpha) = \{\mathbf{x} \mid \mathbf{0} \leq \mathbf{x} \leq \mathbf{n}, x_0 \leq \alpha n_0\}$. We will make use of the following three facts about D, which can be verified easily:

(1) $\alpha \leq \alpha' \implies D(\alpha) \subseteq D(\alpha')$.
(2) $\mathbf{x} \in D(\alpha), \mathbf{x}' \in D(\alpha) \implies \mathbf{x} \vee \mathbf{x}' \in D(\alpha)$.
(3) $\mathbf{x} \in D(\alpha), \mathbf{x}' \in \bigcup_{\alpha' \in [0,1]} D(\alpha') \implies \mathbf{x} \wedge \mathbf{x}' \in D(\alpha)$.

Now, consider any $\alpha, \alpha' \in [0,1]$ with $\alpha \leq \alpha'$, and any $\mathbf{x} \in S(\alpha)$ and $\mathbf{x}' \in S(\alpha')$. We have

$$0 \geq F(\mathbf{x} \vee \mathbf{x}') - F(\mathbf{x}') \geq F(\mathbf{x}) - F(\mathbf{x} \wedge \mathbf{x}') \geq 0.$$

The first inequality follows from Facts 1 and 2 about D and the optimality of $\mathbf{x}'$ at α'. The second inequality follows from the supermodularity of F. The third follows from Fact 3 about D and the optimality of $\mathbf{x}$ at α.

Since this chain of inequalities starts and ends with 0, the inequalities must all hold with equality, implying that $F(\mathbf{x} \vee \mathbf{x}') = F(\mathbf{x}')$ and $F(\mathbf{x}) = F(\mathbf{x} \wedge \mathbf{x}')$. Since $\mathbf{x}' \in S(\alpha')$ and $\mathbf{x} \vee \mathbf{x}' \in D(\alpha')$ by Fact 2, $\mathbf{x} \vee \mathbf{x}' \in S(\alpha')$. Since $x \in S(\alpha)$ and $\mathbf{x} \wedge \mathbf{x}' \in D(\alpha)$ by Fact 3, $\mathbf{x} \wedge \mathbf{x}' \in S(\alpha)$. $\square$

We are now ready to state our result for the existence of efficient trading paths. The main construction in its proof is based on the correspondence (13). Lemma 4.2 implies that under supermodularity, each $S(\alpha)$ is a *sublattice* of the feasible set, allowing us to form a path by choosing the unique maximum lattice element for each α. The main details of the proof consist of establishing monotonicity and continuity of this path.

THEOREM 4.3. *Let C be a cost function obtained via (1) from a pseudo-barrier function and suppose that F is supermodular in $\mathbf{x}$. Then there exists a fair trading path $\mathcal{P}$ for $(\boldsymbol{\Theta}, \mathbf{n}, \boldsymbol{\ell})$ at state $\mathbf{q}$ such that $\mathcal{P}(1)$ is efficient.*

PROOF. We define a particular function $\mathcal{P}$ and show that it satisfies the definition of a trading path $\mathcal{P}$ for $(\boldsymbol{\Theta}, \mathbf{n}, \boldsymbol{\ell})$ at state $\mathbf{q}$ and that its endpoint $\mathcal{P}(1)$ is efficient. We use Lemma 4.2 to show the path is non-decreasing and the Maximum Theorem and Lemma 2.1 to show continuity. Finally, we use the optimality conditions (11–12) to show that $\mathcal{P}$ is a fair trading path.

By Lemma 4.2, for any α, $S(\alpha)$ must be a *sublattice*, i.e., for any $\mathbf{x}, \mathbf{x}' \in S(\alpha)$, $\mathbf{x} \vee \mathbf{x}' \in S(\alpha)$ and $\mathbf{x} \wedge \mathbf{x}' \in S(\alpha)$. This implies that there is a unique maximum element in each set $S(\alpha)$. We can therefore define the path $\mathcal{P}(\alpha) = \max\{\mathbf{x} \mid \mathbf{x} \in S(\alpha)\}$. We first show that $\mathcal{P}$ is a trading path with $\mathcal{P}(1)$ efficient. Assumption 1 immediately implies $\mathcal{P}(0) = 0$. By construction, $\mathcal{P}(1) \in S(1)$ and thus $\mathcal{P}(1) \leq \mathbf{n}$ and is efficient. To show $\mathcal{P}$ is a trading path, it remains to show that it is non-decreasing and continuous.

By Lemma 4.2, for any $\alpha, \alpha' \in [0,1]$ with $\alpha \leq \alpha'$, for any $\mathbf{x} \in S(\alpha)$ and $\mathbf{x}' \in S(\alpha')$, $\mathbf{x} \vee \mathbf{x}' \in S(\alpha')$. Since $\mathbf{x} \vee \mathbf{x}' \geq \mathbf{x}$, this implies that $\max\{\mathbf{y} \mid \mathbf{y} \in S(\alpha')\} \geq \mathbf{x}$ for all $\mathbf{x} \in S(\alpha)$, which implies that $\mathcal{P}$ is non-decreasing.

To show that $\mathcal{P}$ is continuous at any point α, it suffices to show that for all monotonically increasing or decreasing sequences $\{\alpha^i\}_{i=1}^{\infty}$ such that $\lim_{i \to \infty} \alpha^i = \alpha$, $\lim_{i \to \infty} \mathcal{P}(\alpha^i) = \mathcal{P}(\alpha)$. Suppose this were not the case. In particular, suppose that for some α and some increasing or decreasing sequence $\{\alpha^i\}_{i=1}^{\infty}$, $\lim_{i \to \infty} \alpha^i = \alpha$, but either $\lim_{i \to \infty} \mathcal{P}(\alpha^i)$ does not exist or $\lim_{i \to \infty} \mathcal{P}(\alpha^i) \neq \mathcal{P}(\alpha)$. Note that it cannot be the case that $\lim_{i \to \infty} \mathcal{P}(\alpha^i)$ does not exist since $\mathcal{P}$ is increasing and bounded. So suppose that $\mathbf{z} \equiv \lim_{i \to \infty} \mathcal{P}(\alpha^i) \neq \mathcal{P}(\alpha)$.

By Berge's Maximum Theorem (see Appendix A.3), S is upper hemicontinuous. Since $\mathcal{P}(\alpha^i) \in S(\alpha^i)$ for all i, this implies that $\mathbf{z} \in S(\alpha)$ and therefore $\mathbf{z} < \mathcal{P}(\alpha)$ by construction of the path. (Here and throughout we use $\mathbf{a} < \mathbf{b}$ to mean that $a_i \leq b_i$ for all i, and $a_i < b_i$ for some i.) Let $\boldsymbol{\delta} = \mathcal{P}(\alpha) - \mathbf{z}$. Note that $\boldsymbol{\delta} > 0$. If the sequence $\{\alpha^i\}_{i=1}^{\infty}$ is decreasing,

then for all α^i, $\mathcal{P}(\alpha^i) \geq \mathcal{P}(\alpha) = \mathbf{z} + \boldsymbol{\delta}$. But $\mathbf{z}$ cannot be bounded away from every $\mathcal{P}(\alpha^i)$ if $\mathbf{z} = \lim_{i \to \infty} \mathcal{P}(\alpha^i)$, a contradiction.

Suppose then that $\{\alpha^i\}_{i=1}^\infty$ is increasing. We first argue it must be the case that $\delta_0 = 0$. Suppose $\delta_0 > 0$. Since the objective F is concave, for any two points $\mathbf{x}, \mathbf{x}' \in S(\alpha)$ and any $\lambda \in [0,1]$, $\lambda \mathbf{x} + (1-\lambda)\mathbf{x}' \in S(\alpha)$. Therefore, $\mathbf{z} + \boldsymbol{\delta}/2 \in S(\alpha)$. For $\alpha \in [0,1]$, let $D(\alpha) = \{\mathbf{x} \mid \mathbf{0} \leq \mathbf{x} \leq \mathbf{n}, x_0 \leq \alpha n_0\}$ be the feasible set. Since $\mathbf{z} + \boldsymbol{\delta} \in D(\alpha)$, it must be the case that there exists some τ such that for all $i > \tau$, $\mathbf{z} + \boldsymbol{\delta}/2 \in D(\alpha^i)$. Furthermore, since $D(\alpha^i) \subseteq D(\alpha)$ for all i and $\mathbf{z} + \boldsymbol{\delta}/2$ is optimal for α, it must be the case that for all i, for all $\mathbf{x} \in D(\alpha^i)$, $F(\mathbf{x}) \leq F(\mathbf{z} + \boldsymbol{\delta}/2)$. Together these imply that for all $i > \tau$, $\mathbf{z} + \boldsymbol{\delta}/2 \in S(\alpha^i)$, and so $\mathcal{P}(\alpha^i) \geq \mathbf{z} + \boldsymbol{\delta}/2$. But if this is true, we again cannot have $\mathbf{z} = \lim_{i \to \infty} \mathcal{P}(\alpha^i)$, another contradiction, so δ_0 must be 0.

Finally, consider the last case in which $\{\alpha^i\}_{i=1}^\infty$ is increasing and $\delta_0 = 0$. By the same argument above, $\mathbf{z} + \boldsymbol{\delta}/2 \in S(\alpha)$. Thus $F(\mathbf{x}) = F(\mathbf{x} + \boldsymbol{\delta}/2)$, and expanding out the definition of F, $C(\mathbf{q} + \Theta\mathbf{x} + \Theta\boldsymbol{\delta}/2) - C(\mathbf{q} + \Theta\mathbf{x}) = \boldsymbol{\ell}^\top \boldsymbol{\delta}/2$. Similarly, $F(\mathbf{x} + \boldsymbol{\delta}/2) = F(\mathbf{x} + \boldsymbol{\delta})$, so $C(\mathbf{q} + \Theta\mathbf{x} + \Theta\boldsymbol{\delta}) - C(\mathbf{q} + \Theta\mathbf{x} + \Theta\boldsymbol{\delta}/2) = \boldsymbol{\ell}^\top \boldsymbol{\delta}/2$. By Lemma 2.1, for any state $\mathbf{q}'$, $C(\mathbf{q}' + \Theta\boldsymbol{\delta}) - C(\mathbf{q}') = \boldsymbol{\ell}^\top \boldsymbol{\delta}$. Fix some i and let $\mathbf{y} = \mathcal{P}(\alpha_i)$. We must have $\mathbf{z} \geq \mathbf{y}$, and since $\mathbf{z} + \boldsymbol{\delta} \in D(\alpha)$ and $\delta_0 = 0$, $\mathbf{y} + \boldsymbol{\delta} \in D(\alpha_i)$. From the previous paragraph, we know that

$$F(\mathbf{y} + \boldsymbol{\delta}) = \boldsymbol{\ell}^\top (\mathbf{y} + \boldsymbol{\delta}) - C(\mathbf{q} + \Theta\mathbf{y} + \Theta\boldsymbol{\delta}) = \boldsymbol{\ell}^\top \mathbf{y} - C(\mathbf{q} + \Theta\mathbf{y}) = F(\mathbf{y}).$$

Since $\mathbf{y} \in S(\alpha_i)$, $\mathbf{y} + \boldsymbol{\delta} \in S(\alpha_i)$, and $\mathbf{y}$ could not be $\mathcal{P}(\alpha_i)$, another contradiction. This means that $\mathcal{P}$ must be continuous.

We have established that $\mathcal{P}$ is a trading path. The fact that the trading path $\mathcal{P}$ is a fair trading path follows immediately by comparing optimality conditions (11–12) with those in Definition 3.2. $\quad\square$

4.2. When There Is No Fair Trading Path

In the previous section, we showed that an efficient trading path can be constructed under certain favorable conditions. We now show that in general this is not the case. The following applies to a broad class of cost functions, including the LMSR cost function when traders request overlapping bundles.

THEOREM 4.4. *Consider a complete market over $N \geq 3$ states of the world with N Arrow-Debreu securities. Let C be a cost function obtained via (1) with R bounded on $\mathcal{H}(\phi(\Omega))$. There exists a set of orders $(\Theta, \mathbf{n}, \boldsymbol{\ell})$, market state $\mathbf{q}$, and $\epsilon > 0$, such that no ϵ-fair trading path $\mathcal{P}$ for $(\Theta, \mathbf{n}, \boldsymbol{\ell})$ at state $\mathbf{q}$ has an efficient endpoint $\mathcal{P}(1)$.*

The high-level idea behind the proof, which appears in the appendix, is as follows. Suppose that $N = 3$ and consider two traders with orders on bundles $\boldsymbol{\theta}_1 = (1/2, 1/2, 0)$ and $\boldsymbol{\theta}_0 = (0, 2/3, 1/3)$. Intuitively, purchasing $\boldsymbol{\theta}_0$ should simultaneously cause the price of security 1 to drop and the price of security 2 to (quickly) rise. Given an arbitrary cost function C, we use general properties of cost functions (*expressiveness* and *bounded loss*) to construct a specific start state $\mathbf{q}$, quantity n_0, and price ℓ_1 such that if the market state moves in a straight-line path from $\mathbf{q}$ to $\mathbf{q} + n_0\boldsymbol{\theta}_0$, the price of bundle $\boldsymbol{\theta}_1$ starts out above ℓ_1 due to a high price of security 1, temporarily drops (far) below ℓ_1 as the price of security 1 falls, and finally exceeds ℓ_1 again as the price of security 2 rises. By setting ℓ_0 very high, effectively making trader 0's order a market order, we force there to be a unique efficient fill that results in full execution of trader 0's order while trader 1's order is not executed at all. The corresponding (unique) trading path moves the market state in a straight line between $\mathbf{q}$ and $\mathbf{q} + n_0\boldsymbol{\theta}_0$, which, as described above, violates condition (6) for trader 1.
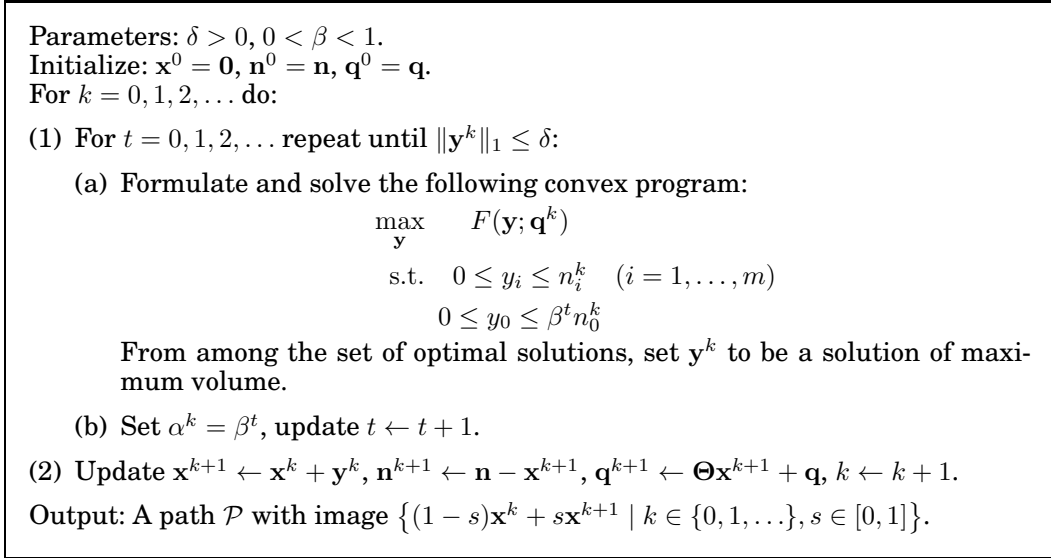
Parameters: $\delta > 0$, $0 < \beta < 1$.
Initialize: $\mathbf{x}^0 = \mathbf{0}$, $\mathbf{n}^0 = \mathbf{n}$, $\mathbf{q}^0 = \mathbf{q}$.
For $k = 0, 1, 2, \ldots$ do:

(1) For $t = 0, 1, 2, \ldots$ repeat until $\|\mathbf{y}^k\|_1 \leq \delta$:

    (a) Formulate and solve the following convex program:

$$\max_{\mathbf{y}} \quad F(\mathbf{y}; \mathbf{q}^k)$$
$$\text{s.t.} \quad 0 \leq y_i \leq n_i^k \quad (i = 1, \ldots, m)$$
$$0 \leq y_0 \leq \beta^t n_0^k$$

    From among the set of optimal solutions, set $\mathbf{y}^k$ to be a solution of maximum volume.

    (b) Set $\alpha^k = \beta^t$, update $t \leftarrow t + 1$.

(2) Update $\mathbf{x}^{k+1} \leftarrow \mathbf{x}^k + \mathbf{y}^k$, $\mathbf{n}^{k+1} \leftarrow \mathbf{n} - \mathbf{x}^{k+1}$, $\mathbf{q}^{k+1} \leftarrow \boldsymbol{\Theta}\mathbf{x}^{k+1} + \mathbf{q}$, $k \leftarrow k + 1$.

Output: A path $\mathcal{P}$ with image $\left\{ (1 - s)\mathbf{x}^k + s\mathbf{x}^{k+1} \mid k \in \{0, 1, \ldots\}, s \in [0, 1] \right\}$.

Fig. 1. **The Fair Trading Path Algorithm (FTPA)**

5. CONSTRUCTING A MAXIMALLY COMPLETE, APPROXIMATELY FAIR TRADING PATH

We now present an algorithm which, given a new order and an order book satisfying Assumption 1, constructs an approximately fair trading path terminating at a complete order fill. As in the construction of Section 4.1, the algorithm forms a path by gradually filling the new order, and filling standing orders when possible while respecting the fair trading path conditions (5–6).

The algorithm, given in Figure 1, consists of an outer and an inner loop. Each iteration of the outer loop solves for an efficient order fill of maximal volume, starting at the current state, subject to the requirement that the additional order volume (i.e., 1-norm) not exceed input parameter δ. The error ϵ by which the output path violates the fair trading path conditions is proportional to δ and properties of the underlying cost function, as shown below. The order fill of bounded volume is itself obtained through one or more iterations of the inner loop. At each iteration of the inner loop, a convex program is solved to find an efficient fill where the fill of the new order (order 0) is bounded according to β^t, where β is the other input parameter. (Note the slight overload in notation: in the algorithm superscripts are used as round indices for iterates, with the exception of β^t where the t represents an exponent.) By only restricting the new order, we can guarantee the (approximate) fair trading path conditions—recall that (6) does not involve the new order. However, this means several inner iterations may be needed for the total order volume to fall below δ as a consequence of continuity. When it is efficient to no longer execute any trades, the algorithm has reached a complete order fill. The variable α^k is for book-keeping purposes in the proofs only.

In step (1a), we require not just an optimal solution to the convex program, but an optimal solution of maximal volume. The following result explains how to achieve this efficiently. Note that it depends on certain special properties of the class of cost functions we consider. The proofs for all results in this section are collected in the appendix.

LEMMA 5.1. *Given an optimal solution to the convex program in (1a), an optimal solution of maximal volume can be obtained via linear programming.*

Let us provide some intuition for the lemma and describe the linear program in more detail. A consequence of Lemma 2.1 is that the directions along which the cost function C is linear form a subspace. We can therefore form a matrix $\mathbf{D} \in \mathbf{R}^{K \times k}$, where $k \leq K$, whose columns form a basis for this subspace. If the cost change along some combination of these directions equals the change in the limit-price term of the objective, then we obtain directions along which the objective is constant. Given an optimal solution $\mathbf{y}'$ to the convex program in step (1a), the LP for extending it to an optimal solution of maximum volume takes the following form:

$$\max_{\mathbf{y}, \alpha} \ \|\mathbf{y}' + \mathbf{y}\|_1$$
$$\text{s.t.} \quad \mathbf{0} \leq \mathbf{y}' + \mathbf{y} \leq \mathbf{n}, \quad \boldsymbol{\Theta}\mathbf{y} = \mathbf{D}\alpha, \quad \boldsymbol{\ell}^\top \mathbf{y} = \mathbf{1}^\top \alpha$$

The latter two constraints ensure that adding $\mathbf{y}$ does not change the objective value, and thus $\mathbf{y}' + \mathbf{y}$ remains an optimal solution. The fact that an LP can be formulated to maximize volume is used in proving correctness of the algorithm.

We now turn to the algorithm's properties. The first element of the proof is to show that the algorithm always progresses from one iteration to the next. For this it is important that in step (1a) we choose an optimal solution of maximal volume.

LEMMA 5.2. *At every iteration $k = 0, 1, \ldots$ the inner loop in step (1) halts.*

From this lemma we obtain that the algorithm will generate an infinite sequence of iterates $\{\mathbf{x}^k\}$. The following gives the main property of this sequence.

THEOREM 5.3. *The sequence of iterates $\{\mathbf{x}^k\}$ generated by the algorithm converges to a maximally complete order fill.*

There are several parts to this result. First, it states that the sequence of iterates converges. Often with numerical algorithms only convergence in objective value holds, not convergence in iterates (e.g., this is common in optimization methods when there are multiple optima). Here we obtain convergence in iterates because we are monotonically filling the orders. Second, it states that the limit point is a complete order fill. There are two sub-cases: 1) at some point the sequence of iterates becomes constant, meaning that the algorithm has effectively converged in a finite number of steps, or 2) the algorithm generates an infinite sequence of distinct iterates. The first case occurs whenever the optimal solution obtained in step (1a) is $\mathbf{0}$, which gives one stopping criterion. The second case is not unusual for numerical algorithms (e.g., consider gradient descent), and allows for several stopping criteria: small enough change in iterates, in efficiency, etc. Under continuity of the objective's gradient, the result implies that condition (12) will eventually hold to any desired tolerance, which is perhaps the most natural stopping criterion for our purposes.

The next result concerns the path constructed by the algorithm. Note that even if the sequence of iterates is infinite, the output path is well-defined. Furthermore, because the iterates converge, it is straightforward to normalize the domain to [0,1] and take $\mathcal{P}(1) = \bar{\mathbf{x}}$, the limit point of the sequence.

THEOREM 5.4. *Suppose the price function π is L-Lipschitz. Then the path $\mathcal{P}$ output by the algorithm is ϵ-fair for $\epsilon = \delta L$.*

The result requires Lipschitz continuity of the order price function. This holds if the cost function C has Lipschitz continuous gradients, and for cost functions of the form (1) that we consider, this is equivalent to R being strongly convex [Hiriart-Urruty and Lemaréchal 2001, Thm 4.2.1].[2] The Lipschitz constant L for the price function

---

[2]For instance, the LMSR cost function is Lipschitz continuous with constant 1 with respect to the 1-norm.

depends on both the underlying constant for C, and the order bundle matrix $\boldsymbol{\Theta}$ (more specifically, its matrix norm). Therefore L will depend on the way bundles are normalized. Throughout, the Lipschitz parameter L refers to continuity with respect to the 1-norm.[3] The correctness of the algorithm and the previous result establish the following.

COROLLARY 5.5. *Suppose the price function π is L-Lipschitz. Then for any $\epsilon > 0$, there exists an ϵ-fair trading path for $(\boldsymbol{\Theta}, \mathbf{n}, \ell)$ at $\mathbf{q}$ terminating at a maximally complete order fill.*

There are still some limitations to these results. We cannot guarantee that the algorithm will reach an efficient endpoint when an ϵ-fair trading path to one exists, let alone parametrize the algorithm to balance path fairness and endpoint efficiency. It is also not straightforward to apply a limiting argument to Corollary 5.5 and obtain existence of an exact ($\epsilon = 0$) and complete trading path.

Let us stress again that all our results rely on Assumption 1 holding at the outset. By Theorem 5.3, the assumption holds after running our algorithm upon each new order arrival. Thus we only need to ensure that it holds when the market is started, by using a call market or starting with an empty order book as suggested earlier.

6. SIMULATIONS

In this section, we investigate the empirical performance of the Fair Trading Path Algorithm (FTPA) against a simple baseline. The baseline takes as input the set of orders $(\boldsymbol{\Theta}, \mathbf{n}, \ell)$, market state $\mathbf{q}$, and a parameter $\delta > 0$, and tries to fill a small quantity of each order in cycle, as long as the trade is profitable. More specifically, for each order $i = 0, 1, \ldots, m$ that has not yet fully executed, let $\tau_i = \min\{\delta, n_i\}$ and check whether $C(\mathbf{q} + \tau_i \boldsymbol{\theta}_i) - C(\mathbf{q}) \leq \tau_i \ell_i$. If so, sell τ_i shares of $\boldsymbol{\theta}_i$ to trader i at this cost, update the current market state $\mathbf{q}$ to $\mathbf{q} + \tau_i \boldsymbol{\theta}_i$, and update n_i to $n_i - \tau_i$. Remove order i from the book whenever n_i reaches 0. Each execution of τ_i shares is filled in a straight line, forming a trading path. The baseline represents a simple and natural way to handle limit orders by executing them against the market maker in small pieces, and is similar to what some practitioners have done [Berg and Proebsting 2009].

We evaluate the FTPA and baseline along four metrics:

— **Trade volume:** The volume (total number of executed shares) of the overall order fill after all orders have been processed.
— **Social welfare:** The efficiency after all orders have been processed.
— **Violation of condition (6):** The maximum amount by which the limit price of any unfilled order exceeds its market price along the trading path, potentially violating condition (6). Estimated by sampling points on the path.
— **Violation of condition (5):** The maximum amount by which the limit price of any order (excluding the new order) falls below its market price while the order is currently being executed, potentially violating condition (5). Estimated by sampling points on the path.

In our experiments we varied the maximum trade volume δ allowed at each step, as well as the liquidity parameter b. All our plots have δ on the x-axis in log scale, and one of the four metrics on the y-axis.

---

[3]Because the number of dimensions is finite, Lipschitz continuity with respect to some norm implies the same for any norm. However, the parameter L depends on the norm chosen.
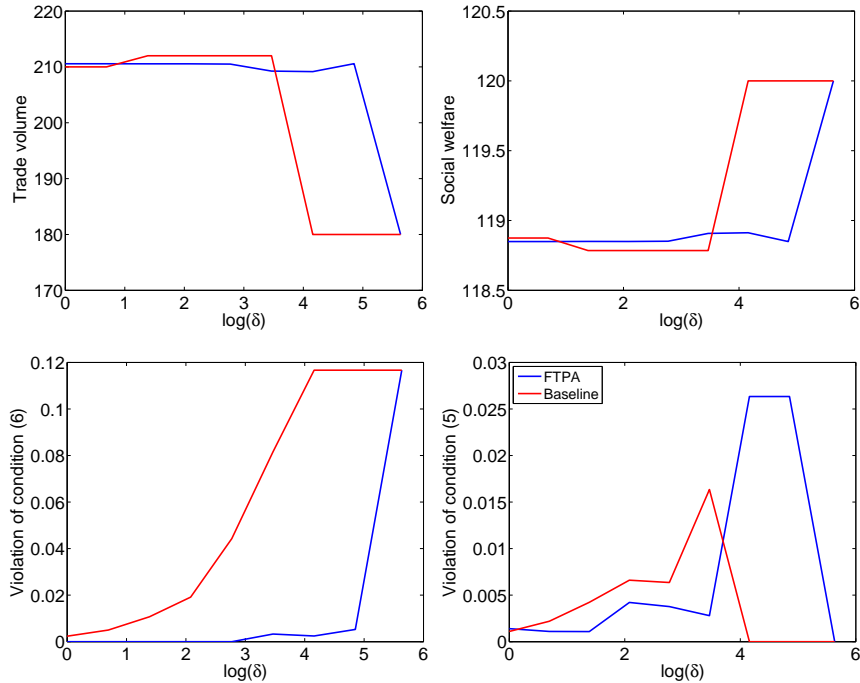
Fig. 2. Algorithm performance on an instantiation of the two trader example from the proof of Theorem 4.4.

Two Trader Instance

We first consider one instance of the general counterexample to efficient trading paths constructed in the proof of Theorem 4.4. We use an LMSR cost function over three securities with liquidity parameter $b = 10$ and initial market state $q = (0, -60, -30)$. Trader 1 places a limit order with $\boldsymbol{\theta}_1 = (1/2, 1/2, 0)$, $n_1 = 100$, and $\ell_1 = 0.45$, which cannot be executed. Trader 0 then places a market order with $\boldsymbol{\theta}_0 = (0, 2/3, 1/3)$, $n_0 = 180$, and $\ell_0 = 1$. By the argument in the proof of Theorem 4.4, there is no ϵ-fair trading path with an efficient endpoint for small ϵ.

Results are shown in Figure 2. First examine the volume of trade. Running the FTPA with $\delta = n_0 + n_1 = 280$ (the rightmost point in the curve) is equivalent to calculating a straight-line path to the unique efficient fill. Only trader 0's order is executed in this case, resulting in a total of 180 shares executed. For all but the highest values of δ, the FTPA and the baseline execute a significant portion of trader 1's order, about 30 shares. As shown in the top right plot, this leads to very little change in efficiency. This is because trader 1 pays exactly his limit price for these shares. The small drop in social welfare for smaller values of δ is therefore due only to the fact that trader 0 pays slightly more when trader 1's order is partially filled.

The bottom left plot shows that for the FTPA, the maximum amount by which an order's limit price exceeds its market price along the trading path grows very slowly in δ, until δ reaches its maximum value of 280 when (as we know from the proof of Theorem 4.4) the market price of trader 1's bundle drops below $1/3$ while $\ell_1 = 0.45$. This quantity grows more quickly in δ for the baseline. Finally, the bottom right plot shows that the maximum amount by which an order's market price exceeds its limit price during execution is relatively small for both algorithms and all values of δ.
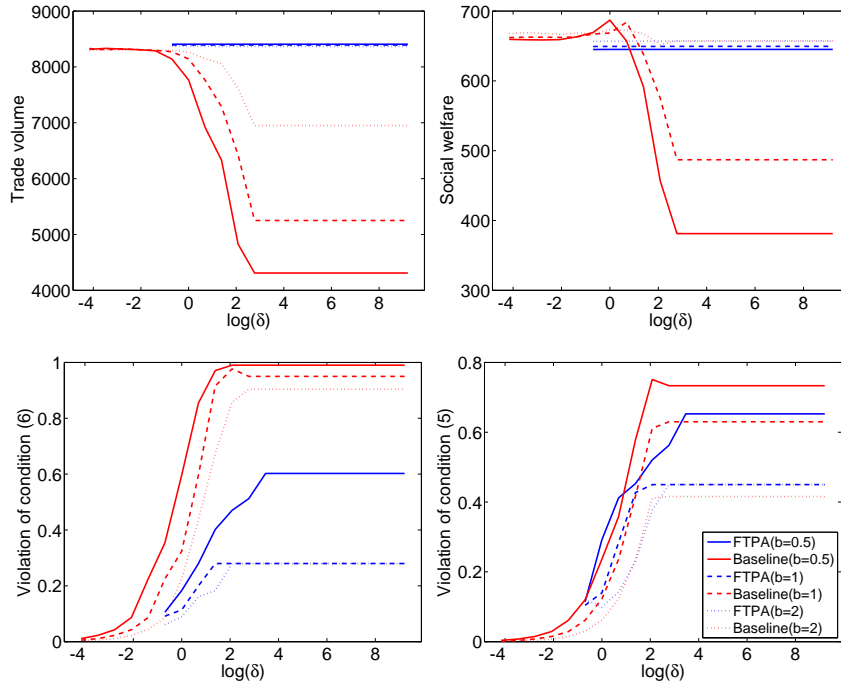
Fig. 3. Algorithm performance on trades generated from survey data on the 2008 elections.

Presidential Elections Market

We next generate trading data from survey results from the 2008 U.S. Presidential election gathered by the Princeton Election Consortium [Wang et al. 2011]. This data consists of probability forecasts on the outcome of the election in a single U.S. state (e.g., how likely it is that McCain wins in OH) or on a conjunction or disjunction of outcomes in up to three U.S. states (e.g., how likely it is that Obama wins in both OH and PA) provided by individuals across the U.S. The data was used by Dudík et al. [2012] to generate synthetic trades to evaluate their constraint-generation based market maker.

We restrict attention to forecasts that cover a set of 5 U.S. states on which there were a large number of predictions. We use the LMSR cost function for a complete market over the $N = 32 \, (= 2^5)$ states of the world corresponding to the 32 possible election outcomes for the two major political parties in these five U.S. states. We generate one limit order from each forecast in the data. A probability forecast on the likelihood of an event E is converted into a request to purchase $n_i = 10$ shares of the bundle $\boldsymbol{\theta}_i$ consisting of one unit each of all securities corresponding to states $\omega \in E$. As the limit price ℓ_i we use the forecast probability of E. We discard forecasts of 0 from the data since orders with a limit price of 0 would never be executed. We run the FTPA and baseline on 1000 orders sub-sampled from this dataset.

It is well known that the performance of LMSR is sensitive to the choice of liquidity parameter b. In initial experiments, we found that with $n_i = 10$ for all i, $b = 1$ led to a reasonable amount of price movement: Prices moved quickly enough that orders were not fully executed immediately upon arrival, but slowly enough that orders in the book were often able to execute later. We chose to run our simulations with three different values of b near to this: 0.5, 1, and 2. The results, which are qualitatively similar for these three values, are shown in Figure 3. Although we plot all results together, results from runs using different liquidity parameters are not directly comparable. As

for comparing the FTPA and baseline, both construct a piece-wise linear path that aims to faithfully approximate a fair trading path, and in both the parameter δ controls the maximum length of each piece, albeit in different ways. We therefore compare the two under the same setting of δ, varying δ. However, the baseline is also able to run for small values of δ where the FTPA takes prohibitively long to finish; for completeness, we also include baseline results in this range to indicate the trends.

From the top left plot we observe that the FTPA leads to a higher trade volume for all comparable δ. Its trade volume is very stable, while trade volume for the baseline quickly degrades beyond a certain threshold. The same general pattern holds for social welfare, as seen in the top right plot. However, here we find that the baseline achieves the highest social welfare of both options at sufficiently small δ.

Observe that the baseline run with small values of δ achieves higher efficiency than the FPTA run with the largest value of δ, even though the latter executes arriving orders efficiently. This is not a contradiction: the FPTA is executing each arriving order efficiently in an online fashion, but this greedy approach is not necessarily efficient for the entire set of orders in batch.

The bottom two plots examine violations of the trading path constraints. The bottom right plot shows that both approaches perform similarly when it comes to condition (5), except for larger δ where the FTPA has an advantage. The biggest discrepancy is for condition (6) in the bottom left plot—the baseline has a tendency to let market prices drift further below limit prices without executing the associated orders. This is arguably the most challenging condition to enforce, because it requires a global control of prices rather than just a local control of the price on the order being executed.

While the FTPA compares favorably to the baseline on the range of δ for which results are available, the trends clearly indicate that, for small enough δ, the baseline can achieve excellent social welfare and trade volume with minimal violation of the trading path conditions. Improving the FTPA so that it can scale to this range is an important avenue for future work.

7. CONCLUSION

Even in a prediction market with a heavily subsidized market maker, a trader may not accept current prices. In that case, a limit order allows the trader to inject information in the market in the form of a constraint on prices, another form of liquidity. By merging cost-function based market makers with limit orders, we design a prediction market mechanism that is well suited for attracting trade. The first traders will interact mainly with the market maker at a low subsidy rate. As trade volume grows, more and more trades will effectively occur as bilateral matches between traders.

We formalized the conventions of continuous-trade markets in a general framework through the notion of a fair trading path, which monotonically fills orders and ensures that standing orders trade as the market prices reach their limit prices. We showed that arriving orders can be filled efficiently (in the sense of social welfare, or total trader utility) under supermodularity conditions, but that this is not possible more generally. We then provided a Fair Trading Path Algorithm which is guaranteed to construct a path respecting limit prices to within any required tolerance, and terminates at a state where no order is profitable.

The most immediate avenue for future work is to develop an algorithm that combines theoretical guarantees with efficient runtime in practice. It would also be worthwhile to perform more advanced simulations with traders that strategize in terms of their reported limit price or the timing of their trades. Another line of research would be to gain a deeper understanding, through theory and simulations, of the algorithm's price discovery properties when there is a common component to the traders' probability forecasts, or when there is a mix of informed and noise traders.

REFERENCES

J. Abernethy, Y. Chen, and J. W. Vaughan. 2013. Efficient Market Making via Convex Optimization, and a Connection to Online Learning. *ACM TEAC* 1, 2 (2013).

S. Agrawal, E. Delage, M. Peters, Z. Wang, and Y. Ye. 2011. A Unified Framework for Dynamic Pari-Mutuel Information Market Design. *Oper Res* 59, 3 (2011).

H. Berg and T. A. Proebsting. 2009. Hanson's Automated Market Maker. *Journal of Prediction Markets* 3, 1 (2009), 45–59.

J. Berg, R. Forsythe, F. Nelson, and T. Rietz. 2008. Results from a Dozen Years of Election Futures Markets Research. *Handbook of Experimental Economics Results* 1 (2008), 742–751.

P. Bossaerts, L. Fine, and J. Ledyard. 2002. Inducing Liquidity in Thin Financial Markets through Combined-value Trading Mechanisms. *European Economic Review* 46 (2002), 1671–1695.

S. Boyd and L. Vandenberghe. 2009. *Convex Optimization*. Cambridge Uni. Press.

M. Chakraborty, S. Das, and J. Peabody. 2015. Price Evolution in a Continuous Double Auction Prediction Market with a Scoring-Rule Based Market Maker. In *AAAI*.

Y. Chen and D.M. Pennock. 2007. A Utility Framework for Bounded-Loss Market Makers. In *UAI*.

M. Dudík, S. Lahaie, and D. Pennock. 2012. A Tractable Combinatorial Market Maker Using Constraint Generation. In *ACM EC*.

C. Featherstone. 2008. A Closer Look at Topkis' Theorem. (2008). Course notes, accessed at http://stanford.io/1C9zQ00.

H. Föllmer and A. Schied. 2002. Convex Measures of Risk and Trading Constraints. *Finance and Stochastics* 6, 4 (2002), 429–447.

L. Fortnow, J. Kilian, D. M. Pennock, and M. P. Wellman. 2005. Betting Boolean-Style: A Framework for Trading in Securities Based on Logical Formulas. *Decision Support Systems* 39, 1 (2005), 87–104.

R. Hanson. 2003a. Book Orders for Market Scoring Rules. (2003). Manuscript.

R. Hanson. 2003b. Combinatorial Information Market Design. *Information Systems Frontiers* 5, 1 (2003), 105–119.

R. Hanson. 2007. Logarithmic Market Scoring Rules for Modular Combinatorial Information Aggregation. *Journal of Prediction Markets* 1, 1 (2007), 3–15.

L. Harris. 2002. *Trading and Exchanges: Market Microstructure for Practitioners*. Oxford University Press.

J. Hiriart-Urruty and C. Lemaréchal. 2001. *Fundamentals of Convex Analysis*. Springer Science & Business Media.

J. Lange and N. Economides. 2005. A Parimutuel Market Microstructure for Contingent Claims. *European Financial Management* 11, 1 (2005).

A. Othman and T. Sandholm. 2011. Liquidity-sensitive Automated Market Makers via Homogeneous Risk Measures. In *WINE*. Springer, 314–325.

M. Peters, A. Man–Cho So, and Y. Ye. 2006. A Convex Parimutuel Formulation for Contingent Claim Markets. In *ACM EC*.

D. M. Rothschild and D. M. Pennock. 2014. The Extent of Price Misalignment in Prediction Markets. *Algorithmic Finance* 3, 1–2 (2014), 2–20.

L. Shapley and M. Shubik. 1977. Trade Using One Commodity as a Means of Payment. *Journal of Political Economy* (1977), 937–968.

D. M. Topkis. 1978. Minimizing a Submodular Function on a Lattice. *Operations Research* 26, 2 (1978).

G. Wang, S. R. Kulkarni, H. V. Poor, and D.N. Osherson. 2011. Aggregating Large Sets of Probabilistic Forecasts by Weighted Coherent Adjustment. *Decision Analysis* 8 (2011), 128—144.

A. APPENDIX

A.1. Proof of Lemma 2.1

Let $\mathbf{p}(\mathbf{q}) = \arg\max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \mathbf{p}^\top \mathbf{q} - R(\mathbf{p})$ as in Equation 2. Expanding out the definition of the cost function, since $C(\mathbf{q} + \mathbf{r}) - C(\mathbf{q}) = C(\mathbf{q}) - C(\mathbf{q} - \mathbf{r})$, we have

$$
\begin{aligned}
2 \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top \mathbf{q} - R(\mathbf{p}) \right) &= \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top (\mathbf{q} - \mathbf{r}) - R(\mathbf{p}) \right) + \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top (\mathbf{q} + \mathbf{r}) - R(\mathbf{p}) \right) \\
&\geq \mathbf{p}(\mathbf{q})^\top (\mathbf{q} - \mathbf{r}) - R(\mathbf{p}(\mathbf{q})) + \mathbf{p}(\mathbf{q})^\top (\mathbf{q} + \mathbf{r}) - R(\mathbf{p}(\mathbf{q})) \\
&= 2 \left[\mathbf{p}(\mathbf{q})^\top \mathbf{q} - R(\mathbf{p}(\mathbf{q})) \right].
\end{aligned}
$$

By definition of $\mathbf{p}(\mathbf{q})$, the first expression and last expression are equivalent, so the inequality must hold with equality. This implies that $\mathbf{p}(\mathbf{q})$ must also be a maximizer of $\mathbf{p}^\top (\mathbf{q} - \mathbf{r}) - R(\mathbf{p})$ and $\mathbf{p}^\top (\mathbf{q} + \mathbf{r}) - R(\mathbf{p})$. Since R is strictly convex, this maximizer is unique, and since R is a pseudo-barrier function, it must lie in the relative interior of $\mathcal{H}(\phi(\Omega))$ (see Section 4.3 of Abernethy et al. [2013]). In order for this to be true, it must be the case that $\mathbf{p}^\top \mathbf{r}$ has the same constant value for all $\mathbf{p} \in \phi(\Omega)$. In particular, since $C(\mathbf{q} + \mathbf{r}) - C(\mathbf{q}) = \mathbf{p}(\mathbf{q})^\top (\mathbf{q} + \mathbf{r}) - R(\mathbf{p}(\mathbf{q})) - \mathbf{p}(\mathbf{q})^\top \mathbf{q} + R(\mathbf{p}(\mathbf{q})) = \mathbf{p}(\mathbf{q})^\top \mathbf{r}$, it must be the case that $\mathbf{p}^\top \mathbf{r} = \gamma$ for all $\mathbf{p} \in \phi(\Omega)$. This immediately implies that $\phi(\omega)^\top \mathbf{r} = \gamma$ for all outcomes $\omega \in \Omega$.

Now, fix any $\mathbf{q}' \in \mathbf{R}^K$ and $\lambda \in \mathbf{R}$. We have

$$
\begin{aligned}
C(\mathbf{q}' + \lambda \mathbf{r}) - C(\mathbf{q}') &= \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top (\mathbf{q} + \lambda \mathbf{r}) - R(\mathbf{p}) \right) - \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top \mathbf{q} - R(\mathbf{p}) \right) \\
&= \lambda \gamma + \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top \mathbf{q} - R(\mathbf{p}) \right) - \max_{\mathbf{p} \in \mathcal{H}(\phi(\Omega))} \left(\mathbf{p}^\top \mathbf{q} - R(\mathbf{p}) \right) = \lambda \gamma. \quad \square
\end{aligned}
$$

A.2. Optimality Conditions (11) and (12)

The following lemma and its proof provide more detail on the derivation of the optimality conditions (11) and (12).

LEMMA A.1. *Suppose $\mathbf{x}^*$ is an optimal solution to*

$$
\max_{\mathbf{x}} \quad F(\mathbf{x}; \mathbf{q})
$$
$$
\text{s.t. } 0 \leq x_i \leq n_i \;\; (i = 0, \ldots, m)
$$

where F is defined as in (8). Then for each $i = 0, 1, \ldots, m$, conditions (11) and (12) hold.

PROOF. By the optimality criteria for differentiable convex optimization problems [Boyd and Vandenberghe 2009, p. 142], a solution $\mathbf{x}^*$ maximizes the objective function F over a convex set X if and only if

$$
\nabla F(\mathbf{x}^*; \mathbf{q})^\top (\mathbf{x} - \mathbf{x}^*) \leq 0 \tag{14}
$$

for all $\mathbf{x} \in X$. In our case X consists of the box constraints $0 \leq x_i \leq n_i$, and these conditions are equivalent to

$$
\begin{aligned}
x_i^* > 0 &\Rightarrow \nabla F(\mathbf{x}^*; \mathbf{q})_i \geq 0, \\
x_i^* < n_i &\Rightarrow \nabla F(\mathbf{x}^*; \mathbf{q})_i \leq 0.
\end{aligned}
$$

To see this, consider and vary each coordinate x_i^* independently, and enforce condition (14). In our case, since $\nabla F(\mathbf{x}; \mathbf{q})_i = \ell_i - \boldsymbol{\theta}_i^\top \nabla C(\boldsymbol{\Theta} \mathbf{x} + \mathbf{q})$, we obtain that for all $i \in \{0, 1, \ldots, m\}$,

$$
\begin{aligned}
x_i^* > 0 &\Rightarrow \pi_i(\mathbf{x}^*) = \boldsymbol{\theta}_i^\top \nabla C(\boldsymbol{\Theta} \mathbf{x}^* + \mathbf{q}) \leq \ell_i, \\
x_i^* < n_i &\Rightarrow \pi_i(\mathbf{x}^*) = \boldsymbol{\theta}_i^\top \nabla C(\boldsymbol{\Theta} \mathbf{x}^* + \mathbf{q}) \geq \ell_i.
\end{aligned}
$$

These are the same conditions as (11) and (12). □

A.3. Berge's Maximum Theorem

Several of our results, including the proof of Theorem 4.3, rely on Berge's Maximum Theorem, which is stated here for completeness. We first review the definition of upper hemicontinuity.

Definition A.2 (*Upper hemicontinuity*). A correspondence $f : X \to Y$ is *upper hemicontinuous* at a point $\mathbf{x} \in X$ if for any sequences $\{\mathbf{x}^i\}_{i=1}^{\infty}$ in X and $\{\mathbf{y}^i\}_{i=1}^{\infty}$ in Y, if $\lim_{i\to\infty} \mathbf{x}^i = \mathbf{x}$, $\lim_{i\to\infty} \mathbf{y}^i = \mathbf{y}$, and $\mathbf{y}^i \in f(\mathbf{x}^i)$ for all i, then $\mathbf{y} \in f(\mathbf{x})$. We say f is upper hemicontinuous if it is upper hemicontinuous at all $\mathbf{x} \in X$.

THEOREM A.3 (MAXIMUM THEOREM). *For any $a, b \in \mathbb{N}$ let $f : \mathbf{R}^a \times \mathbf{R}^b \to \mathbf{R}$ be a continuous function and let $g : \mathbf{R}^b \to \mathbf{R}^a$ be a compact-valued and continuous correspondence. Let*

$$h(\mathbf{y}) = \arg \max_{\mathbf{x} \in g(\mathbf{y})} f(\mathbf{x}, \mathbf{y}), \qquad m(\mathbf{y}) = \max_{\mathbf{x} \in g(\mathbf{y})} f(\mathbf{x}, \mathbf{y}).$$

Then h is non-empty, compact-valued, and upper hemicontinuous, and m is continuous.

A.4. Proof of Theorem 4.4

This proof makes use of two standard properties, *bounded loss* and *expressiveness*, that are guaranteed to be satisfied by cost functions expressible as in Equation 1 with R bounded [Abernethy et al. 2013]. Bounded loss says that the worst case monetary loss of the market maker over all possible trades and all possible states of the world is bounded by a constant. For a complete market, expressiveness says that for any probability distribution $\mathbf{p} \in \Delta_N$ and any $\epsilon > 0$, there exists a market state $\mathbf{q} \in \mathbf{R}^K$ such that $\|\mathbf{p}(\mathbf{q}) - \mathbf{p}\| < \epsilon$. In fact, for any $\mathbf{p}$ that puts positive weight on all states, we have the stronger property that there exists $\mathbf{q}$ such that $\mathbf{p}(\mathbf{q}) = \mathbf{p}$.

We consider the case in which $N = K = 3$. The proof extends easily to $N > 3$ by clustering the securities into three arbitrary non-empty sets and letting the purchase of a cluster correspond to the purchase of every underlying security in that cluster.

Define two bundles of interest, $\boldsymbol{\theta}_1 = (1/2, 1/2, 0)$ and $\boldsymbol{\theta}_0 = (0, 2/3, 1/3)$.

By expressiveness, there must exist a market state $\mathbf{q}_{eq} \in \mathbf{R}^3$ such that $\mathbf{p}(\mathbf{q}_{eq}) = (1/3, 1/3, 1/3)$. We first argue that there must exist a $k > 0$ such that $p_1(\mathbf{q}_{eq} - k\boldsymbol{\theta}_0) > 0.9$. If this were not the case, then if the market reached state $\mathbf{q}_{eq}$, a trader could short sell arbitrarily many shares of $\boldsymbol{\theta}_0$, receiving a payment of at least $1/30$ for each and keeping it all for profit if the true state of the world were ω_1, violating bounded loss.[4]

Similarly, there must exist a $j > 0$ such that $p_2(\mathbf{q}_{eq} + j\boldsymbol{\theta}_0) > 0.9$. If this were not the case, then starting at state $\mathbf{q}_{eq}$, a trader could buy arbitrarily many shares of $\boldsymbol{\theta}_0$ and make a profit of at least 0.1 on each if the true state of the world were ω_2, again violating bounded loss.[5] Since C is convex, the price of bundle $\boldsymbol{\theta}_1$ cannot decrease as $\boldsymbol{\theta}_1$ is purchased, so $\boldsymbol{\theta}_1^{\top} \mathbf{p}(\mathbf{q}_{eq} + j\boldsymbol{\theta}_0 + \tau\boldsymbol{\theta}_1) \geq \boldsymbol{\theta}_1^{\top} \mathbf{p}(\mathbf{q}_{eq} + j\boldsymbol{\theta}_0) > 0.45$ for any $\tau > 0$.

Now, suppose that the market starts at state $\mathbf{q} = \mathbf{q}_{eq} - k\boldsymbol{\theta}_0$ with an empty order book, at which point a trader submits a limit order $(\boldsymbol{\theta}_1, n_1, \ell_1)$ with $\ell_1 = 0.45$ and any $n_1 > 0$. Since $\boldsymbol{\theta}_1^{\top} \mathbf{p}(\mathbf{q}_{eq} - k\boldsymbol{\theta}_0) \geq (1/2) p_1(\mathbf{q}_{eq} - k\boldsymbol{\theta}_0) > 0.45$, the order is not executable and is therefore added to the order book. The next trader then submits an order $(\boldsymbol{\theta}_0, n_0, \ell_0)$

---

[4]For intuition, suppose prices were fixed at $\mathbf{p}$ with $p_1 + p_2 + p_3 = 1$. Short selling $\boldsymbol{\theta}_0$ would yield a payment of $(2/3)p_2 + (1/3)p_3 \geq (1/3)(p_2 + p_3) > 1/30$ if $p_1 < 0.9$.

[5]Again, for intuition, if prices were fixed at $\mathbf{p}$, the profit made for purchasing $\boldsymbol{\theta}_0$ in state ω_2 would be $2/3 - (2/3)p_2 - (1/3)p_3 = 2/3 - (1/3)p_2 - (1/3)(p_2 + p3) \geq 1/3 - (1/3)p_2 > 1/30$ if $p_2 < 0.9$.

with $n_0 = k + j$, and $\ell_0 = 1$. Since ℓ_0 is the maximum possible price for the bundle $\boldsymbol{\theta}_0$, this is a market order.

We show that the unique efficient allocation is $x_0^* = k + j$ and $x_1^* = 0$. That is, the market order is fully executed, while the initial limit order remains in the book. First, note that for any fixed value of x_1, the objective $F(\mathbf{x})$ is maximized if $x_0 = k + j$; social welfare is always improved by executing more of the market order. Therefore, we know that $x_0^* = k + j$, and all that remains is to find the value of x_1 that maximizes social welfare, i.e., we need to find $x_1 \in [0, n_1]$ that maximizes

$$0.45x_1 - C(x_1\boldsymbol{\theta}_1 + (k + j)\boldsymbol{\theta}_0 + \mathbf{q}) = 0.45x_1 - C(x_1\boldsymbol{\theta}_1 + \mathbf{q}_{\mathrm{eq}} + j\boldsymbol{\theta}_0).$$

Taking the derivative with respect to x_1, we have

$$0.45 - \boldsymbol{\theta}_1^\top \mathbf{p}(x_1\boldsymbol{\theta}_1 + \mathbf{q}_{\mathrm{eq}} + j\boldsymbol{\theta}_0) < 0.45 - 0.45 = 0,$$

so the objective is maximized when the limit order is not executed at all and $x_1^* = 0$.

Since only the market order is executed, all trading paths with $\mathcal{P}(1) = \mathbf{x}^*$ are equivalent to (i.e., have the same image as) the path $\mathcal{P}(\alpha) = \alpha\mathbf{x}^*$. Letting $\alpha = k/(k + j)$,

$$\boldsymbol{\theta}_1^\top \nabla C(\boldsymbol{\Theta}\mathcal{P}(\alpha) + \mathbf{q}) = \boldsymbol{\theta}_1^\top \nabla C(\mathbf{q}_{\mathrm{eq}}) = \boldsymbol{\theta}_1^\top \mathbf{p}(\mathbf{q}_{\mathrm{eq}}) = 1/3 < 0.45 = \ell_1$$

which violates the condition in Equation 6 for any $\epsilon < 0.45 - 1/3$. Therefore, for any $\epsilon < 0.45 - 1/3$, $\mathcal{P}$ is not an ϵ-fair trading path, and no ϵ-fair trading path exists with $\mathcal{P}(1) = \mathbf{x}^*$. $\square$

A.5. Proofs for Section 5

We will use the following facts in the proofs of correctness. First, note that $\mathbf{0}$ is always a feasible solution to the convex program in step (1a) of Figure 1, and since $F(\mathbf{0}; \mathbf{q}) = 0$ for any $\mathbf{q}$, the optimal value is always at least 0. It is also straightforward to verify the identity:

$$F(\mathbf{y}'' + \mathbf{y}'; \mathbf{q}) = F(\mathbf{y}''; \boldsymbol{\Theta}\mathbf{y}' + \mathbf{q}) + F(\mathbf{y}'; \mathbf{q}). \tag{15}$$

The identity implies that

$$F(\mathbf{y}^k; \mathbf{q}^k) = F(\mathbf{x}^{k+1}; \mathbf{q}) - F(\mathbf{x}^k; \mathbf{q}). \tag{16}$$

In other words, the optimal objective value at each iteration corresponds to the increase in efficiency from the iteration to the next.

— Convergence and the Proof of Lemma 5.2

Lemma 5.2 follows immediately from the following two results.

LEMMA A.4. *At the start of each iteration $k \geq 0$, the order book is maximally filled.*

PROOF. The statement holds at $k = 0$ by Assumption 1. Let $\mathbf{y}^k$ be the solution found in step (1a) at iteration $k \geq 0$. Let $\tilde{\mathbf{y}}$ be a solution to the optimization in step (1a) at iteration $k + 1$ under the additional constraint that $y_0 = 0$. We have $\mathbf{0} \leq \tilde{\mathbf{y}} \leq \mathbf{n}^k - \mathbf{y}^k$ and $\tilde{y}_0 = 0$. Thus we have

$$0 \leq \tilde{y}_i + y_i^k \leq n_i^k \qquad (i = 1, \ldots, m)$$
$$0 \leq \tilde{y}_0 + y_0^k = y_0^k \leq \alpha^k n_0^k.$$

Thus $\mathbf{y}^k + \tilde{\mathbf{y}}$ was a feasible solution to (1a) at iteration k. We have

$$\begin{aligned} F(\tilde{\mathbf{y}} + \mathbf{y}^k; \mathbf{q}^k) &= F(\tilde{\mathbf{y}}; \mathbf{q}^{k+1}) + F(\mathbf{y}^k; \mathbf{q}^k) \\ &\geq F(\mathbf{y}^k; \mathbf{q}^k) \end{aligned}$$

where the first line applies identity (15) and the second follows from the fact that $\tilde{\mathbf{y}}$ is optimal for (1a) at iteration $k+1$, and thus $F(\tilde{\mathbf{y}}; \mathbf{q}^{k+1}) \geq 0$. Hence $\tilde{\mathbf{y}} + \mathbf{y}^k$ is also an optimal solution to (1a) at iteration k. Note that since $\tilde{\mathbf{y}} \geq \mathbf{0}$ and $\mathbf{y}^k \geq \mathbf{0}$, we have $\|\tilde{\mathbf{y}} + \mathbf{y}^k\|_1 = \|\tilde{\mathbf{y}}\|_1 + \|\mathbf{y}^k\|_1$. However, because $\mathbf{y}^k$ is an optimal solution of maximum 1-norm, we have $\|\tilde{\mathbf{y}} + \mathbf{y}^k\|_1 = \|\tilde{\mathbf{y}}\|_1 + \|\mathbf{y}^k\|_1 \leq \|\mathbf{y}^k\|_1$. This implies that $\|\tilde{\mathbf{y}}\|_1 = 0$, and we must have $\tilde{\mathbf{y}} = \mathbf{0}$. $\square$

LEMMA A.5. *If the order book is maximally filled at the start of iteration k, the inner loop in step (1) halts.*

PROOF. Let $H(\alpha)$ denote the feasible set for convex program (1a) with constraint $0 \leq y_0 \leq \alpha n_0^k$ on the new order. Note that the correspondence H is compact-valued and continuous. (The latter is apparent from the fact that the Hausdorff distance between $H(\alpha)$ and $H(\alpha')$, under the 1-norm, is $n_0^k|\alpha - \alpha'|$.) Let $H^*(\alpha)$ denote the set of optimal solutions to the convex program under parameter α. As the objective F is continuous, the Maximum Theorem (see Appendix A.3) implies that H^* is upper-hemicontinuous.

Suppose the loop in (1) does not halt but rather generates an infinite sequence $\{\mathbf{y}^t\}$. As the feasible set is compact there is a convergent subsequence $T \subseteq \mathbf{N}$ with limit point $\bar{\mathbf{y}} = \lim_{t \in T} \mathbf{y}^t$. We have $\mathbf{y}^t \in H^*(\beta^t)$ and $\lim_{t \in T} \beta^t = 0$, thus by upper-hemicontinuity $\bar{\mathbf{y}} \in H^*(0)$. As the order book is maximally filled, $H^*(0) = \{\mathbf{0}\}$, and therefore $\lim_{t \in T} \mathbf{y}^t = \mathbf{0}$. In particular, for any $\delta > 0$, there is a sufficiently large t such that $\|\mathbf{y}^t\| \leq \delta$. We have reached a contradiction, as this causes the loop in (1) to halt $\square$

LEMMA A.6. *The fair trading path algorithm generates an infinite sequence $\{\mathbf{x}^k\}$ that converges to a limit point $\bar{\mathbf{x}}$.*

PROOF. By Lemma A.4, the order book is maximally filled for all $k \geq 0$, assuming we reach the iteration. Lemma A.5 then establishes that we always progress to the next iteration from the current one.

As we always have $\mathbf{y}^k \geq \mathbf{0}$, by construction the sequence $\{\mathbf{x}^k\}$ is monotonically increasing component-wise: $x_i^k \leq x_i^{k+1}$ ($k = 0, 1, 2, \ldots$). As the feasible set is compact, each component i therefore converges to some $\bar{x}_i$ such that $0 \leq \bar{x}_i \leq n_i$. The vector $\bar{\mathbf{x}}$ formed from these limits is feasible, because we have box constraints, and since the dimensionality is finite, component-wise convergence implies convergence in norm to $\bar{\mathbf{x}}$. $\square$

— Path Properties and the Proof of Theorem 5.4

The results of this section establish Theorem 5.4. For this section we use the notation $S_i(\mathbf{x}; \mathbf{q})$ to denote the i-th component of the gradient $\nabla F(\mathbf{x}; \mathbf{q})$. Note that

$$S_i(\mathbf{x}; \mathbf{q}) \equiv \frac{\partial F}{\partial x_i}(\mathbf{x}; \mathbf{q}) = \ell_i - \pi_i(\mathbf{x}; \mathbf{q}),$$

so that $S_i(\mathbf{x}; \mathbf{q})$ corresponds to the profitability or *surplus* of order i at the market prices under order fill $\mathbf{x}$.

LEMMA A.7. *Suppose the price function is L-Lipschitz with respect to the 1-norm. Then condition (6) is satisfied at all points of the output path $\mathcal{P}$ for $\epsilon = \delta L$.*

PROOF. We have that ∇F is L-Lipschitz because it only differs from the price function by a constant term in each component. Consider iteration k and let $\mathbf{y}^k$ be the optimal solution from step (1a). Fix $s \in [0, 1]$ and let $\tilde{\mathbf{x}} = (1 - s)\mathbf{x}^k + s\mathbf{x}^{k+1} = \mathbf{x}^k + s\mathbf{y}^k$.

By Lipschitz continuity, we have

$$\|\nabla F(\tilde{\mathbf{x}}; \mathbf{q}) - \nabla F(\mathbf{x}^k; \mathbf{q})\| \leq sL\|\mathbf{y}^k\|_1$$
$$\leq \delta L,$$

where we have used the fact that $\|\mathbf{y}^k\|_1 \leq \delta$ as guaranteed by the algorithm, and $s \in [0, 1]$. Under the 1- or 2-norm on the left-hand side, the inequality holds component-wise, and recalling that S_i is the i-th component of ∇F, we have

$$|S_i(\tilde{\mathbf{x}}; \mathbf{q}) - S_i(\mathbf{x}^k; \mathbf{q})| \leq \delta L \tag{17}$$

for $i = 0, 1, \ldots, n$.

Now consider an order $i \neq 0$ such that $x_i^k < n_i$, meaning we had $y_i^{k-1} < n_i^{k-1}$ in the previous iteration. By the optimality condition (12), this implies $S_i(\mathbf{x}^k; \mathbf{q}) = S_i(\mathbf{y}^{k-1}; \mathbf{q}^{k-1}) \leq 0$. (If there was no previous iteration, this holds by Assumption 1.) Combining this with (17) leads to $S_i(\tilde{\mathbf{x}}; \mathbf{q}) \leq \delta L$, which is precisely condition (6) for $\epsilon = \delta L$. □

LEMMA A.8. *Suppose the price function is L-Lipschitz with respect to the 1-norm. Then condition (5) is satisfied at all points of $\mathcal{P}$ for $\epsilon = \delta L$.*

PROOF. Fix $s \in [0, 1]$ and let $\tilde{\mathbf{x}} = (1 - s)\mathbf{x}^k + s\mathbf{x}^{k+1} = \mathbf{x}^k + s\mathbf{y}^k$. By an argument almost identical to the one in the proof of Lemma A.7, we have

$$|S_i(\tilde{\mathbf{x}}; \mathbf{q}) - S_i(\mathbf{x}^{k+1}; \mathbf{q})| \leq \delta L \tag{18}$$

for $i = 0, 1, \ldots, n$. Now consider any order i for which $y_i^k > 0$, meaning that $\mathcal{P}$ is strictly increasing between $\mathbf{x}^k$ and $\mathbf{x}^{k+1}$. By the optimality condition (11), this implies $S_i(\mathbf{x}^{k+1}; \mathbf{q}) = S_i(\mathbf{y}^k; \mathbf{q}^k) \geq 0$. Combining this with (18) leads to $S_i(\tilde{\mathbf{x}}; \mathbf{q}) \geq -\delta L$, which is precisely condition (5) for $\epsilon = \delta L$. □

— Limit Properties and the Proofs of Lemma 5.1 and Theorem 5.3

The results of this section taken together establish Theorem 5.3. Given market state $\mathbf{q}$, the set of efficient order fills is a convex set, because the optimization problem is a convex program. Define the subspace

$$M = \text{span}(\{\mathbf{x} - \mathbf{x}' : \mathbf{x}, \mathbf{x}' \text{ are efficient at state } \mathbf{q}\}). \tag{19}$$

We have the following result for the kinds of cost functions we consider.

LEMMA A.9. *Let C be a cost function derived via (1) where R is a pseudo-barrier function. For any $\mathbf{x}$, the objective function is constant on $\mathbf{x} + M$.*

PROOF. Let $\mathbf{y}$ and $\mathbf{y}'$ be two efficient solutions. As the set of optimal solutions to a convex program is convex, their midpoint $\tilde{\mathbf{y}} = \frac{1}{2}\mathbf{y} + \frac{1}{2}\mathbf{y}'$ is also optimal. Equating the objective values for $\mathbf{y}$ and $\tilde{\mathbf{y}}$, and for $\mathbf{y}'$ and $\tilde{\mathbf{y}}$, yields

$$C(\mathbf{\Theta}\tilde{\mathbf{y}} + \mathbf{q} + \mathbf{\Theta}(\mathbf{y} - \tilde{\mathbf{y}})) - C(\mathbf{\Theta}\tilde{\mathbf{y}} + \mathbf{q}) = \boldsymbol{\ell}^\top(\mathbf{y} - \tilde{\mathbf{y}})$$
$$C(\mathbf{\Theta}\tilde{\mathbf{y}} + \mathbf{q}) - C(\mathbf{\Theta}\tilde{\mathbf{y}} + \mathbf{q} - \mathbf{\Theta}(\tilde{\mathbf{y}} - \mathbf{y}')) = \boldsymbol{\ell}^\top(\tilde{\mathbf{y}} - \mathbf{y}')$$

Noting that $\mathbf{y} - \tilde{\mathbf{y}} = \tilde{\mathbf{y}} - \mathbf{y}' = \frac{1}{2}(\mathbf{y} - \mathbf{y}')$, we can apply Lemma 2.1 to obtain that for any $\mathbf{x}$,

$$C(\mathbf{\Theta}\mathbf{x} + \mathbf{q} + \lambda\mathbf{\Theta}(\mathbf{y} - \mathbf{y}')) - C(\mathbf{\Theta}\mathbf{x} + \mathbf{q}) = \lambda\boldsymbol{\ell}^\top(\mathbf{y} - \mathbf{y}'),$$

or equivalently

$$F(\mathbf{x}; \mathbf{q}) = F(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{y}'); \mathbf{q})$$

for all $\lambda \in \mathbf{R}$. As $\mathbf{y}$ and $\mathbf{y}'$ were arbitrary optimal solutions, this proves that the objective is constant on $\mathbf{x} + M$. $\square$

The result immediately implies Lemma 5.1, because given an efficient $\mathbf{x}$ we can then find a solution of maximal volume by solving $\max\{\|\mathbf{x} + \mathbf{y}\|_1 : \mathbf{0} \leq \mathbf{x} + \mathbf{y} \leq \mathbf{n}, \mathbf{y} \in M\}$. The proof above shows that to formulate the program, we can find a basis for M by identifying the directions along which C is linear. In case of LMSR, for example, this is just the direction $\mathbf{1}$.

LEMMA A.10. *The maximum volume over all efficient order fills at a market state* $\mathbf{q}$ *is a continuous function of* $\mathbf{n}$ *and* $\mathbf{q}$.

PROOF. By Lemma A.9, the objective function F is constant along any direction from M, as defined in (19), starting from any efficient order fill. By the orthogonal decomposition theorem, $\mathbf{x} \in \mathbf{R}^{m+1}$ can be uniquely written as $\mathbf{x} = \mathbf{x}^0 + \mathbf{x}^1$ where $\mathbf{x}^0 \in M$ and $\mathbf{x}^1 \in M^\perp$. We thus have $F(\mathbf{x}) = F(\mathbf{x}^0 + \mathbf{x}^1) = F(\mathbf{x}^1)$ for efficient $\mathbf{x}$. Let

$$H(\mathbf{n}) = \{\mathbf{x}^1 \in M^\perp : \mathbf{0} \leq \mathbf{x}^0 + \mathbf{x}^1 \leq \mathbf{n}, \text{ for some } \mathbf{x}^0 \in M\}. \tag{20}$$

It is straightforward to verify that this set is convex. Therefore consider the new convex program

$$\max_{\mathbf{x}^1 \in H(\mathbf{n})} F(\mathbf{x}^1)$$

This is equivalent to the original efficiency maximizing convex program, in the sense that the optimal value is the same, and a solution to one gives a solution to the other. To be more specific, let $\mathbf{x}(\mathbf{n})$ be a solution to the original program. Decomposing as before, $\mathbf{x}^1(\mathbf{n})$ is an optimal solution to the new convex program. Similarly, by the definition of $H(\mathbf{n})$, if we have an optimal solution to the new program we can complete it with some $\mathbf{x}^0 \in M$ to obtain an optimal solution to the original.

We claim that on $M^\perp$, and therefore on $H(\mathbf{n})$, the objective F has a unique optimal solution. To see this, let $\mathbf{x}^1$ and $\mathbf{y}^1$ be two optimal solutions. There are $\mathbf{x}^0, \mathbf{y}^0 \in M$ such that $(\mathbf{x}^1 + \mathbf{x}^0) - (\mathbf{y}^1 + \mathbf{y}^0) \in M$ by definition, and therefore $\mathbf{x}^1 - \mathbf{y}^1 \in M$. But writing $\mathbf{x}^1 = \mathbf{y}^1 + (\mathbf{x}^1 - \mathbf{y}^1)$, we see that we must have $\mathbf{x}^1 - \mathbf{y}^1 = \mathbf{0}$ because $\mathbf{x}^1, \mathbf{y}^1 \in M^\perp$ and the orthogonal decomposition is unique.

As F is continuous and the feasible set $H(\mathbf{n})$ is continuous in $\mathbf{n}$, the mapping $\mathbf{n} \to \mathbf{x}^1(\mathbf{n})$ is upper-hemicontinuous by the Maximum Theorem. But because this is a singleton mapping, upper hemicontinuity is equivalent to continuity in this case.

The maximum volume over all optimal solutions can therefore be obtained as the optimal value of the following linear program.

$$\max_{\mathbf{x}^0 \in M} \quad \|\mathbf{x}^1(\mathbf{n}) + \mathbf{x}^0\|_1$$
$$\text{s.t. } \mathbf{0} \leq \mathbf{x}^1(\mathbf{n}) + \mathbf{x}^0 \leq \mathbf{n}$$

As $\mathbf{x}^1(\mathbf{n})$ is continuous, both the objective and the feasible set here are continuous. Thus by the Maximum Theorem, the optimal value is continuous in $\mathbf{n}$. The argument for continuity in $\mathbf{q}$ is entirely analogous. $\square$

LEMMA A.11. *The limit point of the sequence of iterates* $\{\mathbf{x}^k\}$ *generated by the algorithm is a maximally complete order fill.*

PROOF. Lemma A.6 establishes that the algorithm indeed generates a convergent sequence $\{\mathbf{x}^k\}$ with limit point $\bar{\mathbf{x}}$. First, suppose that at some finite k', the optimal solution in step (1) of the algorithm is $\mathbf{0}$. Then for all $k \geq k'$, the sequence is identically $\mathbf{x}^{k'}$ and thus $\bar{\mathbf{x}} = \mathbf{x}^{k'}$. As $\mathbf{0}$ is the optimal solution of maximal 1-norm at $\mathbf{q} + \Theta\bar{\mathbf{x}}$, it must be the unique optimal solution, and $\mathbf{x}$ is maximally complete by definition.

Suppose now that the optimal solution is never $\mathbf{0}$, so that the algorithm generates an infinite sequence $\{\mathbf{x}^k\}$. Let $v(\mathbf{n}, \mathbf{q}, \alpha)$ denote the maximum volume over all optimal solutions to the convex program in step (1a), under parameters $\mathbf{n}, \mathbf{q}, \alpha$. Here α refers to the scaling factor on coordinate 0. By Lemma A.10, v is continuous in its arguments. Note that at step k, we have $v(\mathbf{n}^k, \mathbf{q}^k, \alpha^k) = \|\mathbf{y}^k\|_1$. As $\mathbf{y}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$, we have $\lim \mathbf{y}^k = \lim(\mathbf{x}^{k+1} - \mathbf{x}^k) = \mathbf{0}$, by the fact that $\{\mathbf{x}^k\}$ converges (Lemma A.6). Let $\bar{\mathbf{n}}$ and $\bar{\mathbf{q}}$ be the limit points of $\{\mathbf{n}^k\}$ and $\{\mathbf{q}^k\}$. We consider two cases.

Case 1: $\limsup \alpha^k = \bar{\alpha} > 0$. In this case there is a subsequence of $\{\alpha^k\}$ that converges to $\bar{\alpha}$. By continuity, at the limit the maximum volume over all optimal solutions is $v(\bar{\mathbf{n}}, \bar{\mathbf{q}}, \bar{\alpha}) = \lim \|\mathbf{y}^k\|_1 = 0$. This implies that $\mathbf{0}$ is the unique optimal solution under those parameters. But since $\bar{\alpha} > 0$, the constraint on coordinate 0 does not bind, and thus $\mathbf{0}$ is the unique optimal solution taking $\alpha = 1$.

Case 2: $\limsup \alpha^k = 0$. Since $\alpha^k > 0$ for all k, in this case we have $\lim \alpha^k = 0$. By the Maximum Theorem, $\lim \mathbf{y}^k = \mathbf{0}$ is an optimal solution under parameters $(\bar{\mathbf{n}}, \bar{\mathbf{q}}, 0)$ to both the efficiency convex program and the secondary volume-filling LP, and thus $v(\bar{\mathbf{n}}, \bar{\mathbf{q}}, 0) = 0$. By continuity (Lemma A.10) there is a sufficiently small $\alpha' > 0$ of the form β^t such that $v(\bar{\mathbf{n}}, \bar{\mathbf{q}}, \alpha') \leq \delta/2$. By continuity of v in its first two arguments, for all sufficiently large k we then have $v(\mathbf{n}^k, \mathbf{q}^k, \alpha') \leq \delta$. This means that at all such steps k, the inner loop halts when reaching α', if not before. However, this implies that $\limsup \alpha^k \geq \alpha' > 0$, a contradiction, so this case cannot arise. $\square$