An Alternating Algorithm for Finding Linear Arrow-Debreu Market Equilibria

Po-An Chen · Chi-Jen Lu · Yu-Sin Lu

Accepted: 31 May 2021 / Published online: 7 July 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract
Motivated by the convergence result of mirror-descent algorithms to market equilibria in linear Fisher markets, it is natural for one to consider designing dynamics (specifically, iterative algorithms) for agents to arrive at linear Arrow-Debreu market equilibria. Jain (SIAM J. Comput. 37(1), 303–318, 2007) reduced equilibrium computation in linear Arrow-Debreu markets to the equilibrium computation in bijective markets, where everyone is a seller of only one good and a buyer for a bundle of goods. In this paper, we design an algorithm for computing linear bijective market equilibrium, based on solving the rational convex program formulated by Devanur et al. The algorithm repeatedly alternates between a step of gradient-descent-like updates and a distributed step of optimization exploiting the property of such convex program. Convergence can be ensured by a new analysis different from the analysis for linear Fisher market equilibria.

Keywords Arrow-Debreu market equilibria · Convex program · Alternating algorithm

1 Introduction
A market can be thought of as an algorithm or mechanism that implements a social choice of redistribution of various goods between agents (as buyers and sellers for goods) via pricing. In mathematical economics, exchange market models were first proposed by Walras in 1874 and later by Arrow and Debreu in 1954 along with the
concept of market equilibrium [3]. These exchange market models are used to capture the essence of complicated real-world markets.

In the Arrow-Debreu market model, each agent has an initial endowment of divisible goods and a utility function for purchasing a bundle of goods that maximizes her utility when every agent uses the revenue from selling her initial endowment. The Fisher market model [5] can be seen as a special case of the Arrow-Debreu model. In it, each buyer is subject to her budget constraint instead of the revenue from selling her initial endowment. There is a market equilibrium\(^1\) if two conditions are satisfied:

- **Buyer Optimality**, in which every agent uses the revenue from selling her initial endowment to purchase a bundle of goods to maximize her own utility
- **Market Clearance**, in which the total demand for every good equals the total supply

The celebrated theorem by Arrow and Debreu [3] proved the existence of a market equilibrium under some mild necessary conditions for the utility functions. The most common types of utility functions from this class, which are described in more detail in the section of preliminaries, are the linear, Leontief, and Cobb-Douglas functions, and belong to the important class of Constant Elasticity of Substitution (CES) functions [2, 19].

Formally, an Arrow-Debreu market \(M\) consists of a set \(A = \{1, \ldots, n\}\) of agents that trade a set \(G = \{1, \ldots, m\}\) of divisible goods among themselves. Each unit of good \(g \in \{1, \ldots, m\}\) can be bought by an agent at price \(p_g\). The vector of prices is \(p \in \mathbb{R}^m_+ (\mathbb{R}^+ = \{x \geq 0\})\). Each agent \(i\) purchases a consumption plan \(x_i \in \mathbb{R}^m_+\). Each agent \(i\) has an initial endowment \(w_i \in \mathbb{R}^m_+\) of the \(m\) goods, where \(w_{ij}\) is the amount of good \(j\) initially held by \(i\). These initial goods can be sold to other agents and thus provide agent \(i\) with revenue. The revenue can in turn be used to purchase other goods. Every agent \(i\) has a utility function \(U_i : \mathbb{R}^m_+ \rightarrow \mathbb{R}_+, \) where \(U_i(x_i)\) describes how much utility agent \(i\) receives from the consumption plan \(x_i\). An Arrow-Debreu market can then be described as \(M = (A, G, u, w)\). When \(G = A\), i.e., \(w_{ij} = 1\) if \(i = j\) and 0 otherwise, the Arrow-Debreu market becomes a bijective market. The utility of agent \(i\) for the good of agent \(j\) is \(u_{ij} \geq 0\). The directed graph \((A, E)\) has an edge \(ij\) for every pair with \(u_{ij} > 0\) where set \(E\) consists of all the edges. Jain [15] reduced Arrow-Debreu markets to bijective markets as follows. If a good is included in the initial endowment of multiple agents, we give a different name to each good. If an agent has \(k\) goods in the endowment, we split the agent into \(k\) copies that have the same utility function and each owns one of the goods. Consequently, we can focus on the equilibrium computation for bijective markets since any solutions would also work for Arrow-Debreu markets.

Given the convergence of distributed generalized gradient-descent algorithms (i.e., mirror-descent) to market equilibria in the Fisher and CCVF markets [5, 11], it is natural for one to consider designing dynamics (specifically, iterative algorithms)
for agents to arrive at market equilibria in Arrow-Debreu markets.\footnote{2} In this paper, instead of using generic algorithms for solving a convex program, i.e., convex program solvers, we design (partially-distributed) iterative algorithms for solving the following convex program formulated by Devanur et al. in \cite{12} for bijective markets, where everyone is a seller of only one good and a buyer for a bundle of goods:\footnote{3}

\begin{equation}
\begin{aligned}
\min & \sum_j p_j \log \frac{p_j}{\beta_j} - \sum_{i,j} b_{ij} \log u_{ij} \\
\text{subject to} & \; \sum_i b_{ij} = p_j \; \forall j \\
& \; \sum_j b_{ij} = p_i \; \forall i \\
& \; u_{ij}/p_j \leq 1/\beta_i \; \forall i, j \\
& \; p_i \geq 1 \; \forall i \\
& \; b_{ij} \geq 0 \; \forall i, j, \beta_i \geq 0 \; \forall i,
\end{aligned}
\end{equation}

where $b_{ij}$ is paid by agent $i$ to agent $j$ for good $j$, $p_i$ “acts like” the endowed budget $B_i$ of agent $i$ in the Fisher model (yet there is no actual budget but only the initial endowment of goods for agent $i$), and $\beta_i$ is agent $i$’s inverse of “best” bang per buck. Our algorithm for computing linear Arrow-Debreu market equilibria is based on solving this rational convex program that captures buyer optimality and market clearance.

Generic solvers that mostly search through the feasible space for an optimal solution are simply designed for any optimization problem, do not take advantage of the structure of the mathematical program, and may sacrifice scalability, i.e., they may not be fast enough when facing large problems. Since we are dealing with a special convex program characterizing linear bijective market equilibria, update dynamics that exploit the structure of this program may enjoy better scalability and, from a computational perspective, enable potentially (at least partially) distributed implementation rather than entirely centralized computation of generic solvers. Thus, technically generic solvers may not be a good candidate for comparison with our partially-distributed algorithm. On the other hand, we argue the necessity of our design and compare our algorithm mainly with block coordinate gradient descent type methods \cite{4, 18} later in Section 1.1.

\subsection*{1.1 Mirror Descents for Fisher Markets}

Recall that when designing distributed algorithms via mirror descents for Fisher markets \cite{5}, each $\beta_i$ is endogenously set to 1 in the convex program above (there is no concept of variable $\beta_i$ so there is no need for to update $\beta_i$). The convex program is not

\footnote{2}We can even require the dynamics to be of the no-regret property to provide incentives for the agents if possible \cite{7–9, 17, 15}.

\footnote{3}Jain \cite{15} reduced the equilibrium computation in linear Arrow-Debreu markets to the equilibrium computation in bijective markets.
then as follows.

$$\begin{align*}
\min & \quad \sum_j p_j \log p_j - \sum_{i,j} b_{ij} \log u_{ij} \\
\text{s.t.} & \quad \sum_i b_{ij} = p_j \forall j \\
& \quad \sum_j b_{ij} = B_i \forall i \\
& \quad b_{ij} \geq 0 \forall i, j.
\end{align*}$$

The objective function becomes

$$\varphi(b) = \sum_j p_j \log p_j - \sum_{i,j} b_{ij} \log u_{ij} = \sum_{i,j} b_{ij} \log \frac{p_j}{u_{ij}},$$

and the feasible space is

$$S = \left\{ b \in \mathbb{R}^{n \times m} : \sum_j b_{ij} = B_i \forall i, b_{ij} \geq 0 \forall i, j \right\}$$

where $p_j = \sum_i b_{ij}$. The components of the gradient of $\varphi$ are thus

$$(\nabla \varphi(b))(i, j) = 1 - \log \frac{u_{ij}}{p_j}.$$ 

The projection back to the feasible space is applied on vector $b'' = (b''_i)_i = (b''_{ij})_{ij}$ after the update such that $b'' = b - \eta \nabla \varphi(b)$ for a learning rate $\eta$ and $b'$ is the feasible point after projection. The choice of the regularizer and thereby the corresponding Bregman divergence decides the actual distance measure for projection. The Proportional Response (PR) dynamics \cite{20} are the weight updates that result from the Bregman divergence being the KL-divergence by choosing the relative entropy as the regularizer and $\eta = 1$:

$$b'_{ij} = \frac{1}{Z_i} b_{ij} \left( \frac{u_{ij}}{p_j} \right)$$

where $Z_i$ is chosen such that $\sum_j b_{ij} = B_i$. Due to the property of the PR dynamics, the update of $b_{ij}$ does not need to depend on the information of any other agent (given that any price $p_j$ is publicly known), so it can be performed in a distributed way. When the Bregman divergence is defined to be half the square of $l_2$ norm, the algorithm becomes the gradient descent algorithm.

### 1.2 Our Results: Alternating Algorithms for Bijective Markets

In the above formulation (1), other than $b_{ij}$, there is one more set of variables that we need to consider for Arrow-Debreu markets: $\beta_i$. One can simply choose to perform standard gradient updates (with projection) with respect to the concatenation of $(b_{ij})_{ij}$ and $(\beta_i)_i$. However, this approach would preclude any possibility of distributed updates of $(b_{ij})_{ij}$ or/and $(\beta_i)_i$. Since distributed computation is a desirable property to have as in \cite{5} and \cite{7–9}, we would like to utilize the structure of this
special convex program to at least let some variables be updated distributedly. To argue the merits of (partially) distributed updates that involve only (some of) each agent’s own variables, we need to know the implementation subtlety and details behind “centralized” updates of some variables in multi-agent markets. Agents in markets may be physically located in different places instead of residing all in the same place, and need to communicate by message passing to obtain the information about some variables of other agents, thereby enabling each agent’s identical and synchronous computation for the part of centralized updates. Thus, the more reduction in the proportion of centralized updates in variables, the less redundancy in message passing and computation that will occur. Furthermore, distributed updates can be conducted in parallel to increase efficiency in computation time. Even partially distributed computation can be beneficial.

In our case, the bijective market consists of multiple agents, each of which controls its own part of variables \((b_{ij})_{ij}\) and \((\beta_i)_{i}\) and may reside in different physical places. We are thus motivated to design partially distributed updates of these numerous variables in convex program (1) for the bijective market that features each important variable \(\beta_i\), which represents agent \(i\)’s inverse of best bang per buck, and fortunately the structure of this formulation, especially the constraints involving \(\beta_i\), allows the distributed (and possibly paralleled) update of \(\beta_i\) without entanglement with the updates of other \(\beta_j\), which works as insights for our algorithm design sketched in the following with ideas explained.

If we separate the updates of \(b_{ij}\) and \(\beta_i\), we can alternate between updating \(b_{ij}\) and updating \(\beta_i\). Fixing one set of variables temporarily, we can update the other variable set to decrease the objective value. This can be done repeatedly. One natural attempt is to update \(b_{ij}\) using mirror-descent types of approaches and updating \(\beta_i\) simply optimizing the objective value. Careless updates of \(\beta_i\) would defeat the purpose of optimizing the objective over \(b_{ij}\) (in the manner described above).

Observe that \(\beta_i \leq \frac{p_j}{u_{ij}}\) for all \(i, j\) and that the objective is non-increasing with \(\beta_i\) (since the derivative with respect to \(\beta_i\) is \(-\frac{p_i}{\beta_i} \leq 0\)). Thus, we know that \(\beta_i = \min_j p_j / u_{ij}\) is the best possible value if we fix the value of \(b_{ij}\), and this should be the update rule for \(\beta_i\), which can be performed distributedly since it does not involve any information from other agents and every price is just a common information. Note that mirror descents and the convergence analysis in [5, 7] will not work directly since we will modify \(\beta_i\) before each modification of \(b_{ij}\). Nevertheless, we show that a convergence analysis mainly for gradient descents with certain condition will do the job. Thus, our approach in this paper is to alternate between

1. a gradient-descent step of updating \(b_{ij}\) (with the previously chosen \(\beta_i\)) with projection,\(^4\) and
2. an optimization step (instead of the gradient-descent step) of updating \(\beta_i\) (with \(p_j\) determined by the previously chosen \(b_{ij}\) in Step (1)).

\(^4\)The projection would be applied to the whole vector \(b\) as in [5]. Alternatively, one may apply mirror descents with respect to \(b_i\) separately and distributedly, and show that, jointly, the objective still converges to minimum (as has been done for congestion games [7–9]). This can be left as future work.
Specifically, Step (2) is done by each agent $i$ computing $\beta_i = \min_j p_j/u_{ij}$ (the inverse of the best bang per buck given the previously fixed $p_j$ in Step (1)). This can be done distributedly for each agent $i$.

To prove the convergence, we have to argue the following.

- Each gradient step decreases the current objective value by some amount when using a new analysis for gradient descents with a convex objective other than the analyses in [5] and [7].
- Step (2) does not increase the objective value given the previously chosen $b_{ij}$ in Step (1).

Note that we need such a new analysis because the convex function for which we perform the gradient-descent step keeps changing. It is the convex objective in which each $\beta_i$ is fixed to the value chosen in previous iteration.

Our approach bears a resemblance to the well-known block coordinate gradient descent type methods [4, 18], including the Block Coordinate Gradient Descent/Projection (BCGD and BCGP) algorithm and the alternating minimization algorithm [4]. Our $(b_{ij})_{ij}$ and $(\beta_i)_i$ are like blocks in the BCGP algorithm yet there are two main reasons that prevent us from directly using BCGP: first, the BCGP algorithm assumes the global domain of the vector of all variables is a Cartesian product of all blocks’ domains (which are independent) and there is no constraint involving more than one block of variables, but our constraints $u_{ij} p_j \leq \frac{1}{\beta_i} \forall i, j$ violates this; secondly, we cannot perform the update of $(\beta_i)_i$ distributedly if we also do gradient-descent updates for $(\beta_i)_i$. Thus, we do need to design a mixture of the gradient-descent update (with projection) and optimization update in this paper in order to best take advantage of the characteristics of the convex program modeling linear bijective market equilibria although our current analysis can only provide a lower convergence rate than the sublinear rate of the BCGP and two-block alternating minimization algorithms in [4] (see our quantitative result in Theorem 2). Nevertheless, looking at the convergence results, our bound $\lambda$ regarding the Hessian plays a similar role as the Lipschitz constant bounds in the BCGP algorithm.

1.3 Related Work

1.3.1 Connection to Block Coordinate Gradient Descent Type Methods

The BCGD/BCGP method [4] has each iteration consisting of a gradient step without/with projection with respect to some block of variables taken in a cyclic (or random) order. A global sublinear rate of convergence is established and even accelerated when the problem is unconstrained. Our convergence analysis in this paper shares a high-level similarity with those in decrease of the objective value for every iteration (in terms of some sort of recurrence relation) and then showing the global convergence rate using the recurrence relation. Since blocks of variables are updated sequentially, their convergence analysis requires the independence of the blocks’ domains, which is not the case in our convex program for linear bijective market equilibria. We have constraints that involve two blocks of variables so the domains of
blocks are not independent. Also, our algorithm has a learning rate, which is a common concept in machine learning when design algorithms and deriving convergence results, while the BCDG/BCDP algorithms has no such concept.

Specifically, these BCGD/BCGP type methods achieve sublinear convergence rates, for instance in [4], \( O \left( \frac{1}{t+\epsilon} \right) \) for the unconstrained case and \( O \left( \frac{1}{t} \right) \) for the constrained case, where \( t \) is the number of iterations and \( c, c' \) are constants containing the number of blocks; the alternating minimization method achieves a convergence rate of \( O \left( \frac{1}{t-1} \right) \) for \( t \geq 2 \). By setting the learning rate \( \eta \) proportional to \( \frac{1}{t^{2/5}} \), our approach enjoys distributed updates of variables \( \beta_i \) that involve correlated constraints and gives a convergence rate of \( O \left( \frac{1}{\eta^2 t} \right) = O \left( \frac{1}{t^{1/5}} \right) \) by our current analysis (Theorem 2). We leave the job of giving a better convergence rate to immediate future work.

### 1.3.2 Dynamics and Computation for Market Equilibria

For Fisher market equilibria, the Eisenberg-Gale convex program [14] can capture the equilibrium allocation for buyers with utility functions from the same class in the CES family. The problem of equilibrium computation was introduced to the theoretical computer science community by Devanur et al. [13]. The proportional response dynamic is equivalent to a generalized gradient-descent algorithm with Bregman divergences on a convex program that captures the equilibria of Fisher markets with linear utilities [5, 20]. The tatonnement process is a simple and natural rule for updating prices in markets. Cheung et al. [11] established that tatonnement is like a generalized gradient descent that uses the Bregman divergence for the class of Convex Conservative Vector Field (CCVF) markets.

Jain [15] reduced the equilibrium computation in linear Arrow-Debreu markets to the equilibrium computation in bijective markets. Chen et al. [10] showed that the problem of computing an Arrow-Debreu market equilibrium with additively separable utilities is PPAD-complete. Devanur et al. [12] formulated a rational convex program for the linear Arrow-Debreu model to characterize a market equilibrium.

### 1.3.3 Markets on Networks

In almost all the market models we discussed previously, every buyer can shop for goods from every seller. This implies that the agents (buyers/sellers) reside in an underlying network that is a complete graph. In reality, trading may only happen between two immediate neighbors in a graph that is not necessarily complete. Kakade et al. [16] considered Arrow-Debreu markets with such generality and designed algorithms that compute approximate equilibria.

### 1.3.4 Market Games

If buyers are allowed to strategically report their interests in different goods in order to maximize their own utilities, the market becomes a market game [1, 6] in which
Nash equilibrium can be used as the equilibrium concept. The Fisher market game was first studied by Adsul et al. [1] for buyers with linear utility functions. Adsul et al. showed the existence of pure Nash equilibria under mild assumptions and provided the conditions necessary for a pure Nash equilibrium to exist. Brânzei et al. [6] showed that a Fisher market game for buyers with linear, Leontief, and Cobb-Douglas utility functions always has a pure Nash equilibrium. Brânzei et al. also bounded the price of anarchy for Fisher market games.

2 Preliminaries

We now introduce the general Constant Elasticity of Substitution (CES) utility function family as follows:

$$U_i(x_i) = \left( \sum_{j=1}^{m} u_{ij} \cdot x_{ij}^{\rho} \right)^{\frac{1}{\rho}}$$  (2)

where $-\infty < \rho \leq 1$, $\rho \neq 0$. The Leontief, Cobb-Douglas, and linear utility functions shown below are given when $\rho$ approaches $-\infty$, approaches 0, and equals 1, respectively.

- Leontief: $U_i(x_i) = \min_{j \in m} \left\{ \frac{x_{ij}}{u_{ij}} \right\}$  (3)
- Cobb–Douglas: $U_i(x_i) = \prod_{j \in m} x_{ij}^{u_{ij}}$  (4)
- Linear: $U_i(x_i) = \sum_{j \in m} u_{ij}x_{ij}$  (5)

The Leontief function captures the utility of items that are perfect complements. The Linear function captures the utility of items that are perfect substitutes. The Cobb-Douglas function represents a perfect balance between complements and substitutes.

In a market equilibrium, we have a set of prices $p : A \rightarrow \mathbb{R}_+$ and allocations $x : E \rightarrow \mathbb{R}_+$ that satisfy the following conditions [12]. First, agents use an optimal plan at price $p$ if every agent is allocated a utility-maximizing bundle subject to their constraints. This reduces to the following.

- For every $i \in A$, if $x_{ij} > 0$, then $u_{ij} / p_j$ is the maximal value over $j \in A$.
- $p_i > 0$ for every $i \in A$.

The market clears if $\sum_{i \in A} x_{ij} = 1^5$ for every $j \in A$, i.e., every good is fully sold, and $p_i = \sum_{j \in A} x_{ij} p_j$ for every $i \in A$, i.e., the money spent by agent $i$ equals her income $p_i$.

---

5W.l.o.g., we assume that $\sum_{i \in A} w_{ij} = 1$ for every $j \in A$ since only $w_{ii} = 1$ and $w_{ij} = 0$ for $i \neq j$ in bijective markets.
2.1 Convex Program for Linear Bijective Markets

Our algorithm for computing linear Arrow-Debreu market equilibria is based on solving the rational convex program formulated by Devanur et al. in [12]. Devanur et al. presented a rational convex program for the linear bijective market that guarantees the existence of a market equilibrium. We need to formally introduce the convex program here before giving our alternating algorithm in the next section.

\[
\begin{align*}
\min & \sum_j p_j \log \frac{p_j}{\beta_j} - \sum_{ij} b_{ij} \log u_{ij} \\
\sum_i b_{ij} &= p_j \forall j \in A \\
\sum_j b_{ij} &= p_i \forall i \in A \\
u_{ij}/p_j &\leq 1/\beta_i \forall i, j \in E \\
p_i &\geq 1 \forall i \in A \\
b, \beta &\geq 0
\end{align*}
\]

(6)

Note that \(u_{ij}\) is the given utility of agent \(i\) for the item of agent \(j\). Variable \(b_{ij}\) is the money paid by agent \(i\) to agent \(j\). Variable \(\beta_i\) represents the inverse best bang-per-bucks of agent \(i\). Here, variable \(p_i\), which is treated as a function of variables \(b_{ij}\) for all \(j\), corresponds to the value of agent \(i\)'s initial endowment since \(\sum_j b_{ij} = p_i\).

Let \(b = (b_{ij})_{ij}, \beta = (\beta_i)_i\), and \(y = (b, \beta)\). The money spent by agent \(i\) equals \(i\)'s income \(p_i\) (revenue from selling her initial endowment) at market equilibrium, i.e., \(p_i = \sum_j x_{ij} p_j\). The allocation, which is dependent on \(b_{ij}\) and \(p_j\), can be derived by \(x_{ij} = b_{ij}/p_j\). Therefore, we can focus on \(y\). The objective function is

\[
\Phi(y) = \sum_j p_j \log \frac{p_j}{\beta_j} - \sum_{ij} b_{ij} \log u_{ij}.
\]

The space of feasible solutions is

\[
S = \left\{ b \in \mathbb{R}^{n \times m}, \beta \in \mathbb{R}^n : b_{ij} \geq 0 \forall i, j, u_{ij}/\sum_i b_{ij} \leq 1/\beta_i \forall i, j, \sum_j b_{ij} \geq 1 \forall i \right\}.
\]

The space of feasible \(b\)'s fixing the values of \(\beta\) is

\[
S_\beta = \left\{ b \in \mathbb{R}^{n \times m} : b_{ij} \geq 0 \forall i, j, u_{ij}/\sum_i b_{ij} \leq 1/\beta_i \forall i, j, \sum_j b_{ij} \geq 1 \forall i \right\}.
\]

From [12], the following condition is necessary for the existence of an equilibrium. (The argument is provided in the Appendix.)

(*) For every strongly connected component \(S \subseteq E\) of the digraph \((A, E)\), if \(|S| = 1\), then there is a loop incident to the node in \(S\).

The result from [12] shows the feasibility of such a convex program.

**Theorem 1** (THM 1.1 of [12]) Consider an instance of the linear Arrow-Debreu market given by graph \((A, E)\) and the utilities \(u : E \to \mathbb{R}_+\). The convex program (6) is feasible if and only if (*) holds, and, in this case, the optimum value is 0,
and the prices \( p_i \) in an optimal solution give a market equilibrium with allocations \( x_{ij} = b_{ij} / p_j \). Further, if all utilities are rational numbers, then there exists a market equilibrium with all prices and allocations also rational and of a bitsize polynomially bounded in the input size.

We aim to design partially distributed algorithms to solve this rational convex program.

### 3 Alternating Algorithm

Let \( \mathcal{K} \) be the convex feasible space with diameter \( d \), and the objective \( \phi : \mathcal{K} \to \mathbb{R} \) is a convex function satisfying the property that there exists positive \( \lambda, \gamma \in \mathbb{R} \) such that for any \( \xi \in \mathcal{K} \), \( 0 \leq \nabla^2 \phi(\xi) \leq \lambda I \), and \( \|\nabla \phi(\xi)\|_2 \leq \gamma \). Note that \( d \) and \( \gamma \) are bounds that are commonly needed for convergence analysis while how to choose \( \lambda \) can be shown in a given problem.

In our case, assume \( 0 \leq \Phi(y) \leq 1 \) for any \( y \) w.l.o.g. (by Theorem 1), and define \( \phi_\beta(b) = \Phi(b, \beta) \), which is the objective function fixing a value of \( \beta \); \( \mathcal{K} \) consists of the feasible solutions

\[
\mathcal{S}_\beta = \left\{ b \in \mathbb{R}^{n \times m} : b_{ij} \geq 0 \forall i, j, \frac{u_{ij}}{\sum_i b_{ij}} \leq \frac{1}{\beta} \forall i, \sum_j b_{ij} \geq 1 \forall i \right\},
\]
given the values of \( \beta \). For all \( t \), we have that \( \mathcal{S}_{\beta_t} \) satisfy the diameter bound \( d \); \( \phi_{\beta_t} \) satisfy the property with appropriate settings of \( \gamma \) and \( \lambda \) such that \( \gamma \) is an upper bound on each \( \|\nabla \phi_{\beta_t}(\cdot)\|_2 \) while setting \( \lambda \) is non-trivial: specifically, we have the following proposition, whose proof is in Appendix B.

**Proposition 1** For all \( \mathcal{K}_t := \mathcal{S}_{\beta_t} \) and \( \xi \in \mathcal{K}_t \), \( 0 \leq \nabla^2 \phi_{\beta_t}(\xi) \leq \lambda I \) with \( \lambda := n \).

We can obtain the gradient \( \nabla \phi_{\beta_t}(b) \) with respect to \( b \) by taking the derivative of \( b_{ij} \) for each \( i, j \), i.e.,

\[
\nabla \phi_{\beta_t}(b) = \left( \frac{\partial \phi_{\beta_t}(b)}{\partial b_{ij}} \right)_{ij} = \left( 1 - \log \frac{\beta_j u_{ij}}{p_j} \right)_{ij}.
\]

Let \( b_t \in \mathcal{K}_t, b' = b_t - \eta \nabla \phi_{\beta_t}(b_t) \), and \( b_{t+1} = \Pi_{\mathcal{K}_t}(b') = \arg \min_{z \in \mathcal{K}_t} \| z - b' \|_2^2 \), for a learning rate \( \eta \leq \frac{\lambda d^2}{\gamma^2} \), which affects the convergence rate and can be chosen later. The idea is to keep the current values of \( \beta_t \) but update \( b \) using the gradient with respect to \( b \). We denote component \( i, j \) of \( b \) at \( t \) as \( b_t(i, j) \) and component \( i \) of \( \beta \) at \( t \) as \( \beta_t(i) \).

**Algorithm 1** Alternating algorithm.

1. \( b_{t+1} = \Pi_{\mathcal{K}_t} \left( b_t - \eta \nabla \phi_{\beta_t}(b_t) \right) \).
2. \( \beta_{t+1}(i) = \min_j \frac{p_{t+1}(j)}{u_{ij}} \) for each \( i \) where \( p_{t+1}(j) = \sum_i b_{t+1}(i, j) \).
Thus, this is: (1) first moving from point \((b_t, \beta_t)\) to point \((b_t + \beta_t, \beta_t)\), and (2) then moving from point \((b_t + \beta_t, \beta_t)\) to point \((b_t + 1, \beta_t + 1)\). Repeat these two steps, and we need to show that the objective value converges. That is, we have to show that Step (2) does not increase the objective value given \(b_{t+1}\), and that the gradient step decreases the current objective value by some amount. We do this by using a new analysis for gradient descents with a convex objective. We derive a market equilibrium for a linear Arrow-Debreu market.

### 3.1 Convergence

We introduce the following lemma that is for any immediate objective values from standard gradient-descent updates. Using this lemma whose technical part will be proved in the next subsection, we then obtain the convergence result for our alternating algorithm, i.e., Theorem 2. Intuitively, this critical lemma means given a fixed value of \(\beta\), under some condition the gradient descents guarantee that the update of \(b\) decreases the objective value by a certain amount. The update of \(\beta\) only speeds up the convergence process.

**Lemma 1** Let \(b_t \in K_t, b' = b_t - \eta \nabla \phi_{\beta_t}(b_t)\), and \(b_{t+1} = \Pi_{K_t}(b') = \arg\min_{z \in K_t} \|z - b'\|^2_2\), for a learning rate \(\eta \leq \frac{\lambda d^2}{\Gamma \gamma^2}\). Then, whenever \(\phi_{\beta_t}(b_t) - \phi_{\beta_t}(q_t) \geq \frac{d \Gamma \sqrt{6 \lambda \eta}}{\gamma}\) for \(q_t = \arg\min_{z \in K_t} \phi_{\beta_t}(z)\),

\[
\phi_{\beta_t}(b_{t+1}) \leq \phi_{\beta_t}(b_t) - \frac{\eta}{4d} (\phi_{\beta_t}(b_t) - \phi_{\beta_t}(q_t))^2,
\]

which implies that

\[
\phi_{\beta_t}(b_{t+1}) \leq \phi_{\beta_t}(b_t) - \frac{3\lambda \eta^2 \gamma^2}{2}.
\]

**Proof** By Taylor’s expansion at point \((b_{t+1}, \beta_t)\) and the property of \(\phi_{\beta_t}\) in terms of \(\lambda\),

\[
\phi_{\beta_t}(b_{t+1}) \leq \phi_{\beta_t}(b_t) + \langle \nabla \phi_{\beta_t}(b_t), b_{t+1} - b_t \rangle + \frac{\lambda}{2} \|b_{t+1} - b_t\|^2_2.
\]

The last term in the above inequality is at most, by \(b' = b_t - \eta \nabla \phi_{\beta_t}(b_t)\) and the definition of projection,

\[
\frac{\lambda}{2} \|b_{t+1} - b_t\|^2_2 \leq \frac{\lambda}{2} \eta^2 \|\nabla \phi_{\beta_t}(b_t)\|_2^2 \leq \frac{\lambda}{2} \eta^2 \gamma^2.
\]

To bound the second term in (10), note that by the convexity of \(\phi_{\beta_t}\),

\[
\phi_{\beta_t}(q_t) \geq \phi_{\beta_t}(b_t) + \langle \nabla \phi_{\beta_t}(b_t), q_t - b_t \rangle,
\]

which implies that

\[-\langle \nabla \phi_{\beta_t}(b_t), q_t - b_t \rangle \geq \phi_{\beta_t}(b_t) - \phi_{\beta_t}(q_t) \geq 0.\]

Let \(\Delta = -\langle \nabla \phi_{\beta_t}(b_t), q - b_t \rangle\), and we know from above that \(\Delta^2 \geq (\phi_{\beta_t}(b_t) - \phi_{\beta_t}(q_t))^2\). Now we have the following lemma whose proof is in the next subsection.
Lemma 2 When $\Delta \geq 2\eta \gamma^2$, 
\[
\langle \nabla \phi_{\beta_t}(b_t), b_{t+1} - b_t \rangle \leq -\frac{\eta}{3d^2} (\phi_{\beta_t}(b_t) - \phi_{\beta_t}(q_t))^2. \tag{11}
\]

Suppose that $\phi_{\beta_t}(b_t) - \phi_{\beta_t}(q_t) \geq d\gamma \sqrt{6\lambda \eta}$, implying that $\Delta \geq d\gamma \sqrt{6\lambda \eta}$. By $\eta \leq \frac{\lambda d^2}{\gamma^2}$, i.e., $d \geq \gamma \frac{\eta}{\lambda}$. This implies that 
\[
\Delta \geq d\gamma \sqrt{6\lambda \eta} \geq 2\eta \gamma^2,
\]
which satisfies the condition of Lemma 2. Therefore,
\[
\phi_{\beta_t}(b_{t+1}) \leq \phi_{\beta_t}(b_t) \geq \phi_{\beta_t}(q_{t}) + \frac{3\lambda \eta^2 \gamma^2}{2}.
\]

Finally, the guarantee for Step (2) of Algorithm 1 is not hard to see: for each $i$, we know that $\beta_{t+1}(i)$ is the best possible value if we fix values of $b_{ij}$ to $b_{t+1}(i, j)$ so $\beta_{t+1}$ never gives a larger objective value than $\beta_t$.

Lemma 3 By Step (2) of Algorithm 1,
\[
\Phi(b_{t+1}, \beta_{t+1}) \leq \Phi(b_t, \beta_t). \tag{12}
\]

The following main theorem gives the convergence rate based on the results of Lemmas 1 and 3.

Theorem 2 For $\eta = O \left( \frac{1}{t^{7/5}} \right)$,
\[
\Phi(y_t) - \Phi(y^*) \leq \frac{2}{3\lambda \eta^2 \gamma^2 t} = O \left( \frac{1}{\eta^2 t} \right). \tag{13}
\]

In particular, if $\eta = \frac{1}{t^{1/5}}$, the convergence rate is 
\[
O \left( \frac{1}{t^{1/5}} \right).
\]

 Springer
Note that by setting the learning rate $\eta$ proportional to $\frac{1}{t^{2/5}}$, our approach enjoys distributed updates of variables $\beta_i$ that involve correlated constraints and achieves a convergence rate of $O\left(\frac{1}{t^{2/5}}\right)$ while the BCGD/BCGP type methods without correlated constraints have sublinear convergence rates (see more quantitative details in Section 1.3).

**Proof** By Lemmas 1 and 3 combined, we have that

$$
\Phi(y_{t+1}) \leq \Phi(y_t) - \frac{3\lambda \eta^2 \gamma^2}{2},
$$

whenever $\phi_{\beta_i}(b_t) - \phi_{\beta_i}(q_t) \geq d \sqrt{6\lambda \eta}$ for $q_t = \arg \min_{z \in K} \phi_{\beta_i}(z)$, from which we obtain the following claim by $\Phi(\cdot) \leq 1$:

$$
\Phi(y_{t+1}) \leq \Phi(y_t) - \frac{3\lambda \eta^2 \gamma^2}{2} \Phi^2(y_t). \quad (14)
$$

Thus, we have that

$$
\frac{1}{\Phi(y_t)} - \frac{1}{\Phi(y_{t-1})} \geq \frac{3\lambda \eta^2 \gamma^2}{2} \cdot \frac{\Phi^2(y_{t-1})}{\Phi(y_{t-1}) \Phi(y_t)} \geq \frac{3\lambda \eta^2 \gamma^2}{2},
$$

where the first inequality is Inequality (14). It can be concluded that

$$
\frac{1}{\Phi(y_t)} \geq \frac{3\lambda \eta^2 \gamma^2}{2} - 1,
$$

from which along with the condition we have that

$$
d \sqrt{6\lambda \eta} \leq \Phi_{\beta_i}(b_t) - \Phi_{\beta_i}(q_t) \leq \Phi(y_t) - \Phi(y^*) \leq \frac{2}{3\lambda \eta^2 \gamma^2}. 
$$

Therefore, $\eta$ should be set as at most

$$
\left(\frac{2}{3\sqrt{6d}d^{3/2}t}\right)^{2/5} = O\left(\frac{1}{t^{2/5}}\right).
$$

\[\square\]

### 3.2 Proof of Lemma 2

We first rewrite the statement in (11) as follows for convenience and conciseness: $\phi_{\beta_i}$ is rewritten as $\phi$; $b_i$ is rewritten as $b$; $b'$ is rewritten as $\tilde{b}$; $b_{t+1}$ is rewritten as $b'$; $q_t$ is rewritten as $q$. Now we are ready to show that when $\Delta \geq 2\eta \gamma^2$,

$$
\langle \nabla \phi(b), b' - b \rangle \leq -\frac{\eta \Delta^2}{3d^2}. \quad (15)
$$

Note that $\phi(b) - \phi(q) \geq d \sqrt{6\lambda \eta}$ implies that $\Delta \geq d \sqrt{6\lambda \eta}$. By $\eta \leq \frac{\lambda d^2}{\gamma^2}$, we have $d \sqrt{6\lambda} \geq 2\gamma \sqrt{\eta}$, and thus $\Delta \geq d \gamma \sqrt{6\lambda \eta} \geq 2\eta \gamma^2$, which satisfies the condition for Inequality (15).

Consider the triangle formed by $b, b', \tilde{b}$. Let $\alpha_1, \alpha_2, \alpha_3$ denote the angles that correspond to the points $b, b', \tilde{b}$, respectively. Since $b' = \Pi_K(b)$ and $b \in K$, we
know that $\alpha_2 \geq \frac{\pi}{2}$ and thus $\alpha_1 \leq \frac{\pi}{2}$. (When $\bar{b} = b'$, we use the convention that $\alpha_2 = \alpha_3 = \frac{\pi}{2}$.) Use the following figure for illustration throughout the proofs.

As $\bar{b} = b - \eta \nabla \phi(b)$, we have that

$$\langle \nabla \phi(b), b' - b \rangle = \| \nabla \phi(b) \|_2 \| b' - b \|_2 \cos(\pi - \alpha_1)$$

$$= -\| \nabla \phi(b) \|_2 \| b' - b \|_2 \cos \alpha_1,$$

which is at most 0 since $\cos \alpha_1 \geq 0$. Moreover, because $\alpha_2 \geq \frac{\pi}{2}$, we have that

$$\| b' - b \|_2 \geq \| b - \bar{b} \|_2 \sin \alpha_3 = \eta \| \nabla \phi(b) \|_2 \sin \alpha_3.$$

Thus, we derive that

$$\langle \nabla \phi(b), b' - b \rangle \leq -\eta \| \nabla \phi(b) \|_2^2 \cos \alpha_1 \sin \alpha_3.$$

What is left to be shown are the lower bounds for $\cos \alpha_1$ and $\sin \alpha_3$.

Consider the triangle formed by $b, \bar{b}, q$, and let $\alpha_4$ denote the angle at point $b$. Then, we need the following two claims, whose proofs will be provided in the following two subsections.

**Lemma 4** If $\Delta \geq 2\eta \gamma^2$, $\alpha_4 \geq \alpha_1$.

**Lemma 5** If $\Delta \geq 2\eta \gamma^2$, $\sin \alpha_3 \geq \frac{\cos \alpha_4}{3}$.

With these two claims, we obtain

$$\langle \nabla \phi(b), b' - b \rangle \leq -\eta \| \nabla \phi(b) \|_2^2 \cos \alpha_4 \frac{\cos \alpha_4}{3}.$$

Recall that $-\Delta = \langle \nabla \phi(b), q - b \rangle = \| \nabla \phi(b) \|_2 \| q - b \|_2 \cos(\pi - \alpha_4) = -\| \nabla \phi(b) \|_2 \| q - b \|_2 \cos \alpha_4$. Thus,

$$\cos \alpha_4 = \frac{\Delta}{\| \nabla \phi(b) \|_2 \| q - b \|_2} \geq \frac{\Delta}{d \| \nabla \phi(b) \|_2},$$

and we have proven Inequality (15).
3.2.1 Proof of Lemma 4

Consider the triangle formed by $\mathbf{b}, \bar{\mathbf{b}}, \mathbf{q}$. Let $\bar{\mathbf{b}}'$ be the point on plane $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{q}$ such that the triangles $\mathbf{b} - \bar{\mathbf{b}} - \bar{\mathbf{b}}'$ and $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$ are identical. ($\bar{\mathbf{b}}' = \mathbf{b}'$ if the triangles $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{q}$ and $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$ are on the same plane.) Clearly, angle $\mathbf{b} - \bar{\mathbf{b}} - \bar{\mathbf{b}}'$ equals angle $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$, which is $\alpha_1$. Thus, we can consider triangle $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$ instead of $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$.

Assume for the sake of contradiction that $\alpha_4 < \alpha_1$. Note that since $\eta < \frac{\Delta}{2y^2}$, we have

$$\frac{3}{2} \leq \frac{\eta}{\Delta} \leq \frac{\Delta}{y^2}.$$

This implies that line $\mathbf{b} - \mathbf{q}$ must intersect the line $\bar{\mathbf{b}}' - \bar{\mathbf{b}}$ at some point $\mathbf{z}$ with $\| \mathbf{z} - \bar{\mathbf{b}} \|_2 < \| \mathbf{b}' - \bar{\mathbf{b}} \|_2$. However, since $\mathbf{b}, \mathbf{q} \in \mathcal{K}$ and $\mathcal{K}$ is convex, we know that $\mathbf{z} \in \mathcal{K}$, and thus $\| \mathbf{z} - \bar{\mathbf{b}} \|_2 \geq \| \mathbf{b}' - \bar{\mathbf{b}} \|_2$ by the definition of $\mathbf{b}'$. This is a contradiction as $\| \mathbf{b}' - \bar{\mathbf{b}} \|_2 = \| \mathbf{b}' - \bar{\mathbf{b}} \|_2$. We can conclude that $\alpha_4 \geq \alpha_1$.

3.2.2 Proof of Lemma 5

Since $\alpha_4 \geq \alpha_1$ and $\| \mathbf{y}_t - \mathbf{q} \|_2 \geq \| \mathbf{y}_t - \mathbf{y}' \|_2$, $\bar{\mathbf{b}}'$ must lie inside triangle $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{q}$, which implies that the line passing through $\mathbf{q}$ and $\bar{\mathbf{b}}'$ must intersect line segment $\mathbf{b} - \bar{\mathbf{b}}$ at some point $\mathbf{w}$. Let $a_1, a_2, a_3$ denote angles $\mathbf{b} - \mathbf{q} - \mathbf{b}', \mathbf{q} - \mathbf{b}' - \bar{\mathbf{b}}$, and $\mathbf{w} - \bar{\mathbf{b}} - \mathbf{b}'$, respectively.

We first claim that $a_2 \geq \frac{\pi}{2}$. To see that, we compare it with angle $\mathbf{b} - \mathbf{b}' - \mathbf{q}$, denoted as $a_2'$, which is at least $\frac{\pi}{2}$. Since triangles $\mathbf{b} - \bar{\mathbf{b}} - \bar{\mathbf{b}}'$ and $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$ are identical, angle $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$ (denoted as $\theta$) is the same as angle $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{b}'$ (denoted as $\theta$), which implies that angle $\mathbf{b}' - \mathbf{b} - \mathbf{q}$ is at most angle $\mathbf{b}' - \mathbf{b} - \mathbf{q}$ since $\mathbf{b}'$ lies on the same plane as $\mathbf{b} - \bar{\mathbf{b}} - \mathbf{q}$. Since $\| \mathbf{b}' - \bar{\mathbf{b}} \|_2 = \| \mathbf{b}' - \bar{\mathbf{b}} \|_2$, if we rotate triangles $\mathbf{b}' - \bar{\mathbf{b}} - \mathbf{q}$ and $\bar{\mathbf{b}}' - \bar{\mathbf{b}} - \mathbf{b}'$ along line $\mathbf{q} - \bar{\mathbf{b}}$ to make them lie on the same plane, $\bar{\mathbf{b}}'$ must lie inside of triangle $\mathbf{b}' - \mathbf{b} - \mathbf{q}$. As $a_2' \geq \frac{\pi}{2}$, we must have that $a_2 \geq a_2' \geq \frac{\pi}{2}$.

Next, note that $\alpha_3 = a_2 - a_3 \geq \frac{\pi}{2} - a_3$ and $a_3 = a_1 + a_4$. So, $\sin \alpha_3 \geq \sin(\frac{\pi}{2} - a_1 - a_4) = \cos(a_1 + a_4) = \cos a_1 \cos a_4 - \sin a_1 \sin a_4 \geq \cos a_4 \sqrt{1 - \sin^2 a_1} - \sin a_1$ as $\sin a_4 \leq 1$ and $\sin a_1 \geq 0$. Moreover,

$$\sin a_1 \leq \frac{\| \mathbf{b} - \mathbf{w} \|_2}{\| \mathbf{b} - \mathbf{q} \|_2} \leq \frac{\| \mathbf{b} - \bar{\mathbf{b}} \|_2}{\| \mathbf{b} - \mathbf{q} \|_2} \leq \frac{\eta \| \nabla \phi(\mathbf{b}) \|_2}{\eta y^2 \cos a_4} \leq \frac{\eta y^2 \cos \alpha_4}{\Delta} \leq \cos \frac{\alpha_4}{2}.$$

As a result, we have that $\sin a_3 \geq \cos \alpha_4 \sqrt{1 - \frac{1}{4}} \geq \frac{\cos \alpha_4}{3}$. 

\[ \square \]
4 Future Work

Our current analysis can only provide a lower convergence rate than the BCGP and two-block alternating minimization algorithms in [4]. Deriving a better convergence rate is an immediate future work. In this article, the projection is applied to the whole vector \( b \) as in [5]. As a future direction, to design a fully distributed iterative algorithm, the mirror descent algorithm may need to be applied with respect to \( b_i \) (with projection) separately, and the analysis has to show that the objective still jointly converges to minimum (as has been done for congestion games [7–9]).

Agents may strategically report their interests in different services in order to maximize their own utilities in markets. When we allow this and treat the market equilibrium computation as a mapping from a given report, we are studying market games. As far as we know, it is still an open question Nash equilibria exist in “Arrow-Debreu market games” (as in Fisher market games [6]). It would be also interesting to consider different classes of underlying social networks that describe crowd structures [16]. There are some price-of-anarchy results in Fisher market games [6]. Another interesting research direction might be better bounding the price of anarchy for Fisher/Arrow-Debreu market games.

Appendix A: Argument of Condition (*) for Theorem 1

We restate the argument of [12] here. Assume that \( \{k\} \) is a singleton strongly connected component without a loop. Let \( T \) denote the set of nodes different from \( \{k\} \) that can be reached on a directed path in \( E \) from \( \{k\} \). In an equilibrium allocation, the agents in \( T \cup \{k\} \) spend all their money on the goods of the agents in \( T \), which implies that \( p_k = 0 \), contrary to our assumption that \( p_j > 0 \) for every \( j \in A \).

Appendix B: Proof of Proposition 1

Let \( b \in S_\beta \). Consider any \( i, k \in A, j, l \in G \). First, we have

\[
\frac{\partial \phi_\beta(b)}{\partial b_{ij}} = -\log u_{ij} + \log \frac{p_j}{\beta_j} + 1 = 1 - \left( \log u_{ij} - \log \frac{p_j}{\beta_j} \right) = 1 - \log \frac{u_{ij} \beta_j}{p_j}.
\]

Let \( p_j = \sum_i b_{ij} \). Thus, we have

\[
\frac{\partial \phi_\beta(b)}{\partial b_{ij}} = 1 - \log \frac{u_{ij} \beta_j}{\sum_i b_{ij}}.
\]

Next, note that except for \( i = k \), we have

\[
\frac{\partial^2 \phi_\beta(b)}{\partial b_{ij} \partial b_{kl}} = 0,
\]

and if \( i = k \), we have

\[
\frac{\partial^2 \phi_\beta(b)}{\partial b_{ij} \partial a_{kl}} = \frac{1}{\sum_i b_{ij}} = \frac{1}{p_j}.
\]
This means that each entry of the Hessian matrix $\nabla^2 \phi_\beta(b)$ is at most $n$. Then for any $z \in \mathbb{R}^{n \times n}$, we have

$$z^T \cdot (\nabla^2 \phi_\beta(b)) \cdot z = \left( \sum_i z_{ij} \frac{1}{p_j} \right)_{ij} \cdot z$$

$$= \sum_{i,j} z_{ij} \left( \sum_i z_{ij} \frac{1}{p_j} \right)$$

$$= \sum_j \frac{1}{p_j} \left( \sum_i z_{ij} \right)^2$$

$$\leq \sum_j \left( z_{1j} + z_{2j} + \ldots + z_{nj} \right)^2$$

$$\leq \sum_j \left( |z_{1j}| + |z_{2j}| + \ldots + |z_{nj}| \right)^2$$

$$\leq \sum_j \left( 1^2 + 1^2 + \ldots + 1^2 \right) \left( z_{1j}^2 + z_{2j}^2 + \ldots + z_{nj}^2 \right)$$

$$= n \left( \sum_{i,j} z_{ij}^2 \right).$$

The first inequality is by $p_i \geq 1$. The third inequality is by Cauchy-Schwarz inequality. This implies that $\nabla^2 \phi_\beta(b) \preceq \alpha I$ with $\alpha = n$.

Acknowledgements  We would like to thank Ling-Wei Wang for useful discussions.

References

1. Adsul, B., Babu, C.S., Garg, J., Mehta, R., Sohoni, M.: A simplex-like algorithm for fisher markets. In: Proc. of Algorithmic Game Theory (2010)
2. Arrow, K.J., Chenery, H.B., Minhas, B.S., Solow, R.M.: Capital-labor substitution and economic efficiency. The Review of Economics and Statistics (1961)
3. Arrow, K.J., Debreu, G.: Existence of an equilibrium for a competitive economy. Econometrica: Journal of the Econometric Society (1954)
4. Beck, A., Tetruashvili, L.: On the convergence of block coordinate descent type methods. SIAM J. Optim. 23(4), 2037–2060 (2013)
5. Birnbaum, B., Devanur, N.R., Xiao, L.: Distributed algorithms via gradient descent for fisher markets. In: Proceedings of the 12th ACM Conference on Electronic Commerce (2011)
6. Brânzei, S., Chen, Y., Deng, X., Filos-Ratsikas, A., Frederiksen, S.K.S., Zhang, J.: The fisher market game: Equilibrium and welfare. In: Proc. of 28th Conference on Artificial Intelligence (2014)
7. Chen, P.-A., Lu, C.-J.: Generalized mirror descents in congestion games with splittable flows. In: Proc. 13th International Conference on Autonomous Agents and Multiagent Systems (2014)
8. Chen, P.-A., Lu, C.-J.: Playing congestion games with bandit feedbacks (extended abstract). In: Proc. 14th International Conference on Autonomous Agents and Multiagent Systems (2015)
9. Chen, P.-A., Lu, C.-J.: Generalized mirror descents in congestion games. Artif. Intell. 241, 217–243 (2016)
10. Chen, X., Dai, D., Du, Y., Teng, S.-H.: Settling the complexity of arrow-debreu equilibria in markets with additively separable utilities. arXiv:0904.0644 (2009)
11. Cheung, Y.K., Cole, R., Devanur, N.: Tatonnement beyond gross substitutes?: gradient descent to the rescue. In: Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing (2013)
12. Devanur, N.R., Garg, J., Végh, L.A.: A rational convex program for linear arrow-debreu markets. ACM Trans. Econ. Comput. 5(1), 1–13 (2016)
13. Devanur, N.R., Papadimitriou, C.H., Saberi, A., Vazirani, V.V.: Market equilibrium via a primal-dual algorithm for a convex program. J. ACM (JACM) 55(5), 1–18 (2008)
14. Eisenberg, E., Gale, D.: Consensus of subjective probabilities: The pari-mutuel method. The Annals of Mathematical Statistics (1959)
15. Jain, K.: A polynomial time algorithm for computing an arrow-debreu market equilibrium for linear utilities. SIAM J. Comput. 37(1), 303–318 (2007)
16. Kakade, S.M., Kearns, M., Ortiz, L.E.: Graphical economics. In: Proc. of Conference on Learning Theory, pp. 17–32 (2004)
17. Kleinberg, R., Piliouras, G., Tardos, E.: Load balancing without regret in the bulletin board model. Distrib. Comput. 24, 21–29 (2011)
18. Luo, Z.Q., Tseng, P.: On the convergence of the coordinate descent method for convex differentiable minimization. J. Optim. Theory Appl. 72(1), 7–35 (1992)
19. Solow, R.M.: A contribution to the theory of economic growth. The quarterly journal of economics (1956)
20. Zhang, L.: Proportional response dynamics in the fisher market. In: Proc. of Automata Languages and Programming (2009)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Po-An Chen¹ · Chi-Jen Lu² · Yu-Sin Lu³

Chi-Jen Lu
cjlu@iis.sinica.edu.tw

Yu-Sin Lu
s8700937@gmail.com

1 Institute of Information Management, National Yang Ming Chiao Tung University, 1001 University Rd, 300, Hsinchu City, Taiwan
2 Institute of Information Science, Academia Sinica, 128 Academia Road, Section 2, Nankang, 11529, Taipei, Taiwan
3 Industrial Technology Research Institute, 195, Section 4, Chung Hsing Rd, Chutung, Hsinchu County, Taiwan