Beyond Classical Fisher Markets: Non-convexity and Uncertainty

Yinyu Ye
Stanford University

Joint work with Devansh Jalota, Marco Pavone, and Qi Qi

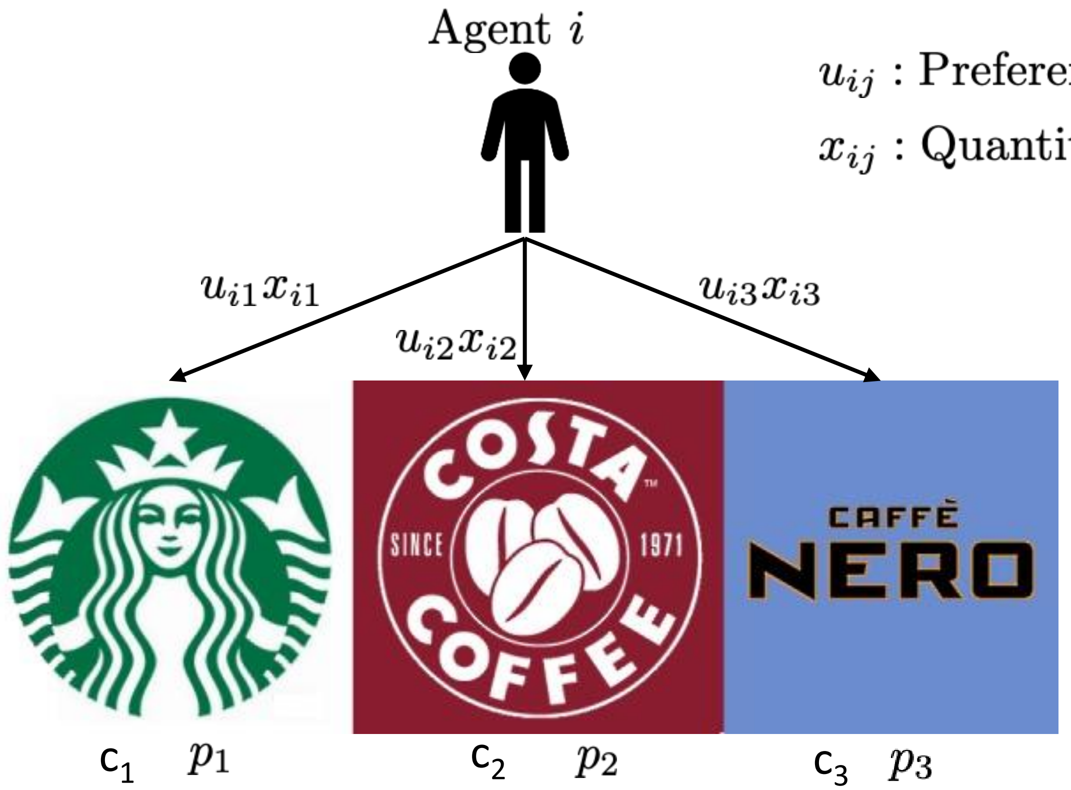
Algorithms, Approximation, and Learning in Market and Mechanism Design Workshop

There are many settings when we need to (fairly) allocate shared resources/goods to users

Public Good Allocation

Vaccine Allocation

A key framework to achieve a (fair or envy-free) allocation of resources/goods is Fisher Markets



u_{ij} : Preference of Agent i for one unit of good j

x_{ij} : Quantity of good j purchased by person i

p_j : Price of Good j

w_i : Budget of Agent i

Individual Optimization Problem:

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$

$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$

$$\mathbf{x}_i \geq \mathbf{0}$$

The prices can be derived from a centralized Eisenberg-Gale social optimization problem

Individual Optimization Problem:

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$

$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$

$$\mathbf{x}_i \geq \mathbf{0}$$

$\Longleftrightarrow$

Social Optimization Problem:

$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t. } \sum_i x_{ij} \leq c_j, \forall j \in [M]$$

Capacity Constraints

$$x_{ij} \geq 0, \forall i, j$$

p_j : Price of Good j = Dual Variable of Constraint j

The social problem can be solved in polynomial time (Jain, Vazirazi, Ye 2005; Jalota, Qi et al. GEB 2023)

However, the applicability of Fisher Markets may be limited

Classical Fisher Markets

Individual's choice under *only* budget constraint

Require complete information on utilities and budgets to compute prices

We extend classical Fisher Markets to take into account practical considerations

Classical Fisher Markets		Our Work
Resource Allocation under budget and capacity constraints	→	Individuals under budget and other **physical (e.g., knapsack)** constraints Jalota, Pavone, Qi, **Ye** GEB'23
Require complete information on deterministic utilities and budgets to compute prices	→	Set prices in online and incomplete/uncertain information environment of Fisher Markets Jalota, **Ye** WINE'23

Organization

- Fisher Markets with Additional Constraints: Non-convexity
- Distributed Algorithms for Fisher Markets
- Online Algorithms in Stochastic Fisher Markets: Uncertainty
- Conclusion/Takeaways

Organization

- **Fisher Markets with Additional Constraints: Non-convexity**
- Distributed Algorithms for Fisher Markets
- Online Algorithms in Stochastic Fisher Markets: Uncertainty
- Conclusion/Takeaways

We consider convex IOP where agents have additional linear constraints beyond budgets

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$

$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i$$

$$\mathbf{x}_i \geq \mathbf{0}$$

Physical Constraints

Constraint Matrix

Fisher markets with additional constraints have different properties from classical Fisher markets

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$

$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i$$

$$\mathbf{x}_i \geq \mathbf{0}$$

Physical Constraints

Constraint Matrix

1. Competitive or Market Equilibrium may not Exist

2. Market Equilibrium may not be Unique

3. **[Giffen Goods]** An increase in the price of a good may result in an increased demand of those goods

4. The set of equilibrium prices is **non-convex**

Under mild conditions, however, the market equilibrium exists

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$

$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i$$

$$\mathbf{x}_i \geq \mathbf{0}$$

Constraint Matrix

Physical Constraints

Theorem 1: Market Equilibrium Exists if there under some technical assumptions, such as there is a good that does not belong to any physical constraint

Theorem 2: Market Equilibrium Exists if b_{it} = 0 for all i, t.

Can we develop a method to compute equilibria with additional constraints when they exist?

Can the Convex Fisher Market social optimization problem with additional constraints be used to set equilibrium prices?

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$

$$\text{s.t.} \quad \mathbf{p}^T \mathbf{x}_i \leq w_i$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i$$

$$\mathbf{x}_i \geq \mathbf{0}$$

Constraint Matrix

? $\Longleftrightarrow$

Physical Constraints

Social Optimization Problem:

SOP-I

$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t.} \quad \sum_i x_{ij} = c_j, \forall j \in [M]$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i, \forall i \in [N]$$

$$x_{ij} \geq 0, \forall i, j$$

Theorem: The dual variables of the capacity constraint of **SOP-I** is an equilibrium for homogeneous constraints

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$
$$\text{s.t.} \;\; \mathbf{p}^T \mathbf{x}_i \le w_i$$
$$A_t^{(i)} \mathbf{x}_i \le 0, \forall t \in T_i$$
$$\mathbf{x}_i \ge \mathbf{0}$$

Constraint Matrix

YES $\Longleftrightarrow$

Social Optimization Problem:

SOP-I

$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$
$$\text{s.t.} \;\; \sum_i x_{ij} = c_j, \forall j \in [M]$$
$$A_t^{(i)} \mathbf{x}_i \le 0, \forall t \in T_i, \forall i \in [N]$$
$$x_{ij} \ge 0, \forall i, j$$

Physical Constraints

This gives a polynomial time algorithm to compute market equilibria

Theorem: However, in general, the dual variables of the capacity constraint of **SOP-I** may not be equilibrium prices

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$
$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$
$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i$$
$$\mathbf{x}_i \geq \mathbf{0}$$

Constraint Matrix

$\not\Leftrightarrow$

Physical Constraints

Social Optimization Problem:

SOP-I

$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$
$$\text{s.t. } \sum_i x_{ij} = c_j, \forall j \in [M]$$
$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i, \forall i \in [N]$$
$$x_{ij} \geq 0, \forall i, j$$

A plausible approach to account for physical constraints in Fisher Markets can be achieved through Budget Perturbations

SOP-I

$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t.} \quad \sum_i x_{ij} = c_j, \forall j \in [M]$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i, \forall i \in [N]$$

$$x_{ij} \geq 0, \forall i, j$$

A plausible approach to account for physical constraints in Fisher Markets can be achieved through Budget Perturbations

SOP-I

$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t.} \quad \sum_i x_{ij} = c_j, \forall j \in [M]$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i, \forall i \in [N]$$

$$x_{ij} \geq 0, \forall i, j$$

BA-SOP

Budget Perturbation

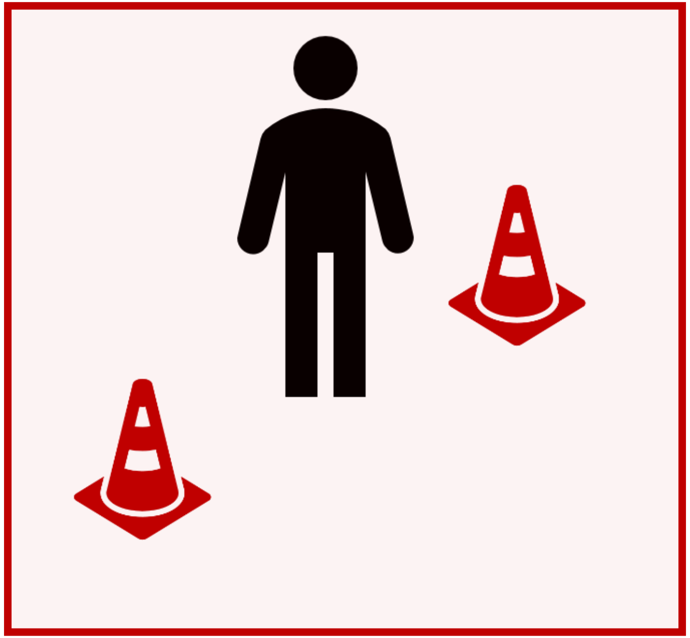
$$\max_{\mathbf{x}_i} \sum_i (w_i + \lambda_i) \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t.} \quad \sum_i x_{ij} = c_j, \forall j \in [M]$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i, \forall i \in [N]$$

$$x_{ij} \geq 0, \forall i, j$$

Budget Perturbations allow more constrained agents to have "higher priority" to get their goods



Low λ_i

High λ_i

Theorem 4: The dual variables of the capacity constraint of **BP-SOP** are the market equilibrium price iff $\lambda_i = \sum_t r_{it} b_{it}$

Individual Optimization Problem:

IOP

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$
$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \le w_i$$
$$A_t^{(i)} \mathbf{x}_i \le b_{it}, \forall t \in T_i$$
$$\mathbf{x}_i \ge \mathbf{0}$$

$$\Longleftrightarrow$$

Social Optimization Problem:

BA-SOP

$$\max_{\mathbf{x}_i} \sum_i (w_i + \lambda_i) \log \left(\sum_j u_{ij} x_{ij} \right)$$
$$\text{s.t. } \sum_i x_{ij} = c_j, \forall j \in [M]$$
$$A_t^{(i)} \mathbf{x}_i \le b_{it}, \forall t \in T_i, \forall i \in [N]$$
$$x_{ij} \ge 0, \forall i, j$$

r_{it}: Dual Variable of Physical Constraint

p_j : Price of Good j = Dual Variable of Constraint j

However, determining budget perturbations is PPAD-hard

The problem of finding a market equilibrium in Fisher Markets with linear constraints is **PPAD-hard**

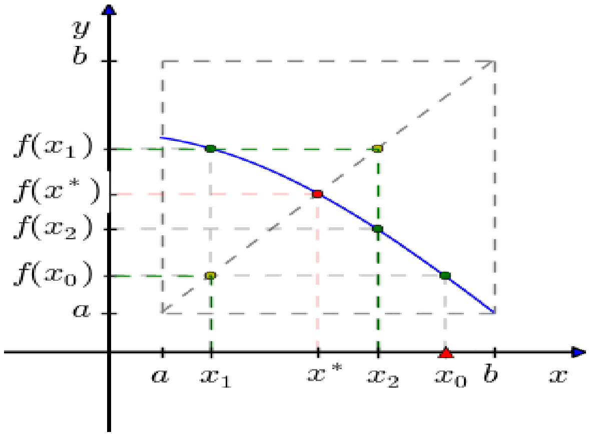
↓

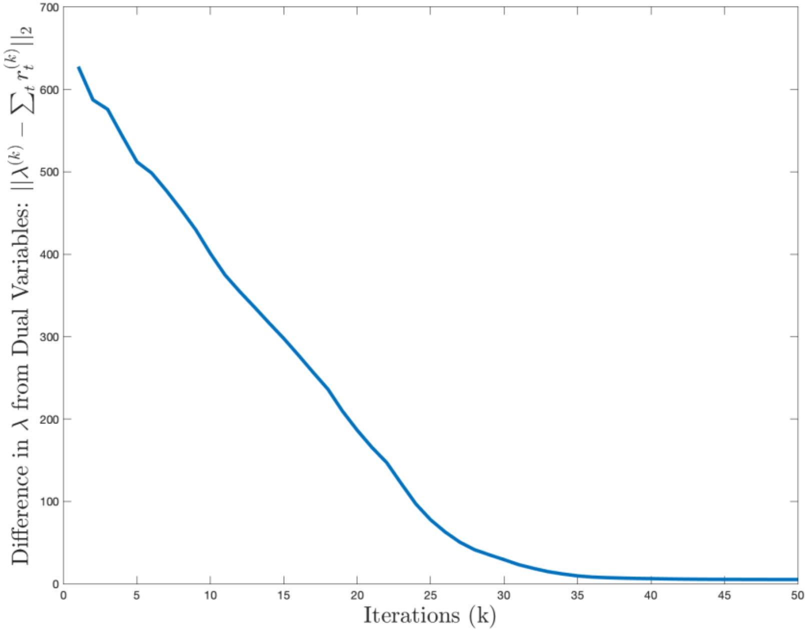
Thus, determining budget perturbations, in general, is a challenging problem

To determine the perturbation constants we test a fixed-point iterative procedure

Algorithm 1: Fixed Point Scheme

Input : Tolerance ϵ, Function $G(\cdot)$ to calculate dual variables
Output: Budget Perturbation Parameters λ
$\lambda \leftarrow \mathbf{0}$;
$\mathbf{R} \leftarrow G(\lambda)$;
$q_i \leftarrow \sum_{t=1}^{l_i} r_{it} b_{it}, \forall i$;
while $\|\lambda - \mathbf{q}\|_2 > \epsilon$ **do**
  $\lambda_i \leftarrow \sum_{t=1}^{l_i} r_{it} b_{it} \ \forall i$;
  $\mathbf{R} \leftarrow G(\lambda)$;
  $q_i \leftarrow \sum_{t=1}^{l_i} r_{it} b_{it}, \forall i$;
end



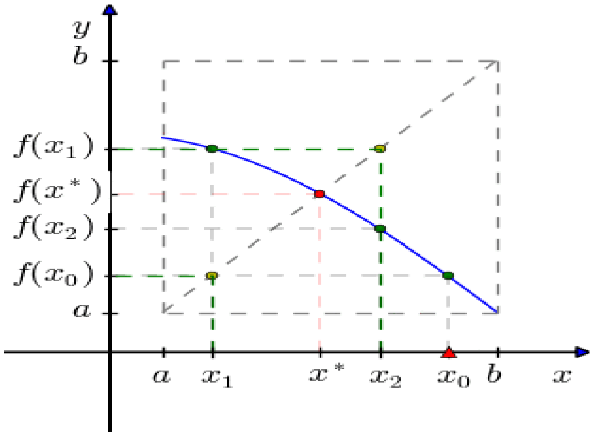


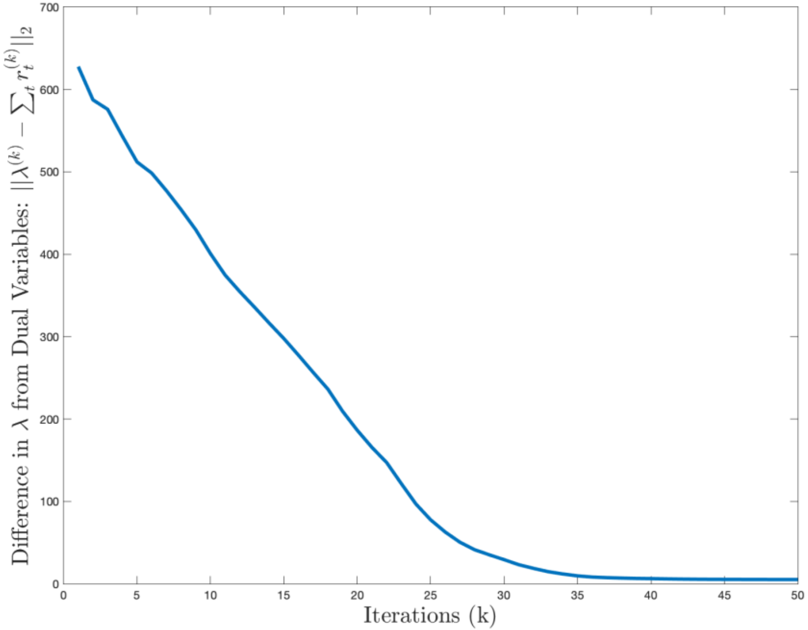
However, determining the budget perturbations requires solving a large-scale centralized optimization

Algorithm 1: Fixed Point Scheme

Input : Tolerance ϵ, Function $G(\cdot)$ to calculate dual variables
Output: Budget Perturbation Parameters λ

$\lambda \leftarrow \mathbf{0}$;
$\mathbf{R} \leftarrow G(\lambda)$;
$q_i \leftarrow \sum_{t=1}^{l_i} r_{it} b_{it}, \forall i$;
while $\|\lambda - \mathbf{q}\|_2 > \epsilon$ **do**
  $\lambda_i \leftarrow \sum_{t=1}^{l_i} r_{it} b_{it} \; \forall i$;
  $\mathbf{R} \leftarrow G(\lambda)$;
  $q_i \leftarrow \sum_{t=1}^{l_i} r_{it} b_{it}, \forall i$;
end





Solving a centralized optimization requires complete information on agents' budgets and utilities

Organization

- Fisher Markets with Additional Constraints: Non-convexity
- **Distributed Algorithms for Fisher Markets**
- Online Algorithms in Stochastic Fisher Markets: uncertainty
- Conclusion/Takeaways

Alternating Direction Method of Multipliers

$$\min_{\mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y}} \quad h(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + g(\mathbf{y})$$
$$\text{s.t.} \quad A\mathbf{x} + B\mathbf{y} = \mathbf{c}$$

$$\mathcal{L}_\beta(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + g(\mathbf{y}) + \lambda^T(A\mathbf{x} + B\mathbf{y} - \mathbf{c}) + \frac{\beta}{2}\|A\mathbf{x} + B\mathbf{y} - \mathbf{c}\|^2$$

Algorithm 1: Two Block ADMM

Input : Initial dual multiplier $\lambda^{(0)}$, and initial vector $\mathbf{y}^{(0)}$
for $k = 0, 1, 2, \ldots$ **do**
  $\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x} \in \mathcal{X}} \mathcal{L}_\beta(\mathbf{x}, \mathbf{y}^{(k)})$;
  $\mathbf{y}^{(k+1)} = \arg\min_{\mathbf{y} \in \mathcal{Y}} \mathcal{L}_\beta(\mathbf{x}^{(k+1)}, \mathbf{y})$;
  $\lambda^{(k+1)} \leftarrow \lambda^{(k)} - \beta(A\mathbf{x}^{(k+1)} + B\mathbf{y}^{(k+1)} - \mathbf{c})$;
end

ADMM helps break down a large problem into small tractable sub-problems

Enables a market Implementation where users solve individual objective

Distributed optimization enables a natural market implementation where users optimize individual objectives under given prices

Centralized Problem

Solve 1 big optimization problem

Split variables to transform the social optimization problem and apply ADMM to this transformed problem

All users solve an individual optimization objective at each iteration

Iteratively solve this problem to update budget perturbation and learn equilibrium prices

We obtain a natural market implementation through ADMM with Classical Fisher Markets

We apply ADMM to the following transformed problem (BA-SOP-ADMM) where we add a variable y

$$\max_{\mathbf{x}_i \in \mathcal{X}_i, \mathbf{y}_i \in \mathcal{Y}_i} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t.} \quad \sum_i y_{ij} = c_j, \forall j \in [M]$$

$$\mathbf{x}_i = \mathbf{y}_i, \forall i \in [N]$$

$$x_{ij} \geq 0, \forall i, j$$

We get a natural market implementation

Repeat until convergence to Equilibrium Price:

1. Agents distributedly solve regularized version of IOP based on market price
2. Market designer updates baseline demand **y** based on observed demands **x**
3. Prices are updated in the market using a tatonnement style update with a fixed step-size

We also obtain a natural market implementation through ADMM with Additional Constraints

We apply ADMM to the following transformed problem (BA-SOP-ADMM) where we add a variable y

$$\max_{\mathbf{x}_i \in \mathcal{X}_i, \mathbf{y}_i \in \mathcal{Y}_i} \sum_i \widetilde{w}_i \log \left(\sum_j u_{ij} x_{ij} \right)$$

$$\text{s.t.} \quad \sum_i y_{ij} = c_j, \forall j \in [M]$$

$$\mathbf{x}_i = \mathbf{y}_i, \forall i \in [N]$$

$$A_t^{(i)} \mathbf{x}_i \leq b_{it}, \forall t \in T_i$$

$$x_{ij} \geq 0, \forall i, j$$

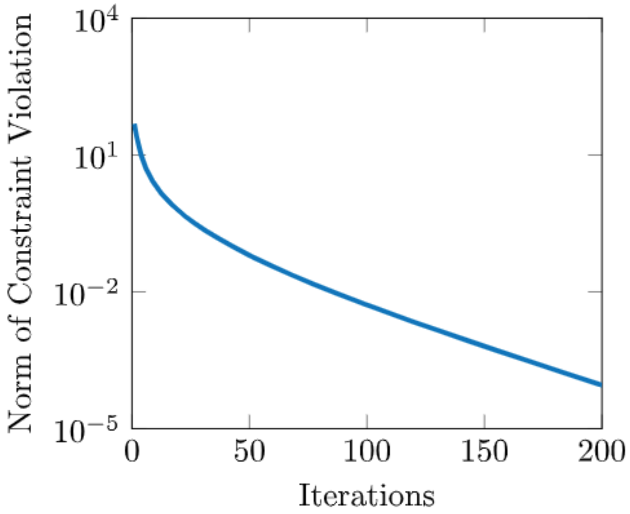
We get a natural market implementation

Repeat until convergence to Equilibrium Price:

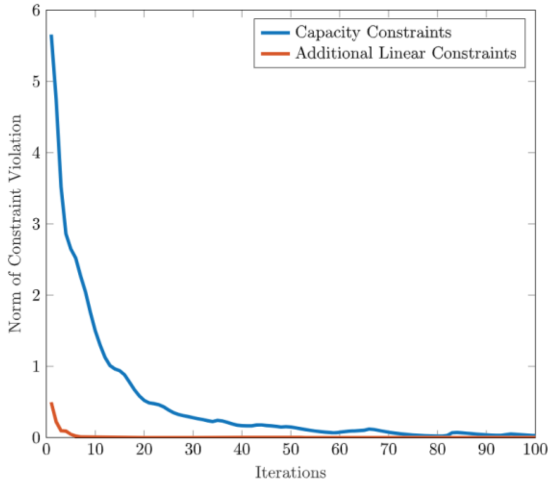
1. Agents distributedly solve regularized version of IOP based on market price
2. Market designer updates baseline demand **y** based on observed demands **x**
3. Prices and perturbations are updated in the market using a tatonnement style update with a fixed step-size

Applying ADMM to our setting achieves good convergence guarantees

Homogenous Constraints



Non-Homogenous Constraints



Provable Convergence Guarantees for classical Fisher markets and Fisher markets with homogeneous linear constraints

Can this distributed implementation be made online where users arrive into the market sequentially with uncertainty?

Yes! For classical Fisher markets

Ongoing Work: Extending online implementation to Fisher markets with linear constraints

Organization

- Fisher Markets with Additional Constraints: Non-convexity
- Distributed Algorithms for Fisher Markets
- **Online Algorithms in Stochastic Fisher Markets: Uncertainty**
- Conclusion/Takeaways

Recall the prices can be derived from a centralized optimization problem that requires complete information

Individual Optimization Problem:

$$\max_{\mathbf{x}_i} \sum_j u_{ij} x_{ij}$$
$$\text{s.t. } \mathbf{p}^T \mathbf{x}_i \leq w_i$$
$$\mathbf{x}_i \geq \mathbf{0}$$

$\Longleftrightarrow$

Social Optimization Problem:

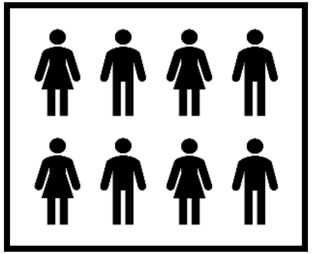
$$\max_{\mathbf{x}_i, \forall i \in [N]} \sum_i w_i \log \left(\sum_j u_{ij} x_{ij} \right)$$
$$\text{s.t. } \sum_i x_{ij} \leq c_j, \forall j \in [M]$$
$$x_{ij} \geq 0, \forall i, j$$

Capacity Constraints

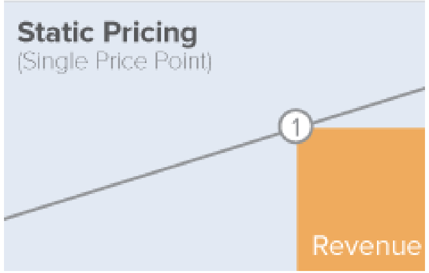
p_j : Price of Good j = Dual Variable of Constraint j

We start by focusing on the problem of online arrivals with incomplete information in a classical Fisher market

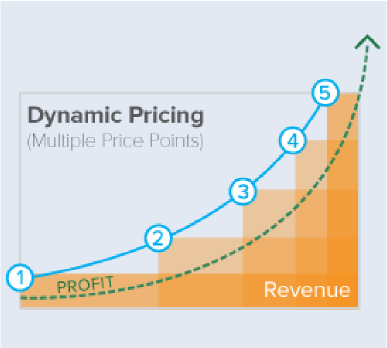
We study an online incomplete information variant of Fisher markets



Buyers arrive sequentially with utility and budget parameters drawn i.i.d. from a distribution



Establish performance limits of static pricing algorithms, including one that sets expected equilibrium prices



Develop an adaptive expected equilibrium pricing approach with strong performance guarantees

Develop a revealed preference algorithm with sub-linear regret and capacity violation

Online Pricing Market: evaluate algorithms through the absolute regret of social welfare and capacity violation

Regret (Optimality Gap)

Difference in the Optimal Social Objective of the online policy $\boldsymbol{\pi}$ *to that of the optimal offline social value*

$$R_n(\boldsymbol{\pi}) = \mathbb{E}\left[\sum_i w_i \log\left(\sum_j u_{ij} x^*_{ij}\right) - \sum_i w_i \log\left(\sum_j u_{ij} x_{ij}(\boldsymbol{\pi})\right)\right]$$

Optimal Offline Objective

Objective of online policy

Constraint Violation or Market Clearance

Norm of the violation of capacity constraints of the online policy $\boldsymbol{\pi}$

$$V_j(\boldsymbol{\pi}) = \sum_j x_{ij}(\boldsymbol{\pi}) - c_j$$

Violation of Capacity Constraint of good *j*

$$V_n(\boldsymbol{\pi}) = ||\mathbb{E}[V(\boldsymbol{\pi})^+]||_2$$

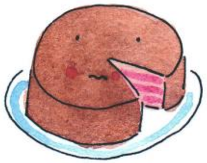
Norm of the expected constraint violation

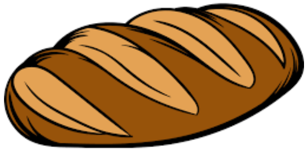
Limitations of Static Pricing

Theorem: No static pricing algorithm can achieve either a regret or capacity violation of better than $\Omega(\sqrt{n})$, where n is the number of arriving users

Problem with static pricing: Using optimal expected prices, the capacity violation is $\Omega(\sqrt{n})$, with n agents

2 goods, each with a capacity of $\boldsymbol{n}$





Two agent types specified by (Utility for Good 1, Utility for Good 2)

Type I: (1, 0)

Arrival Probability =

Type II: (0, 1)

Arrival Probability =

Static Expected Prices: (0.5, 0.5)

While $\frac{n}{2}$ users of Type I arrive in expectation, the realized arrivals of type I users deviates by $O(\sqrt{n})$

Can we develop adaptive pricing algorithms with improved performance guarantees?

We overcome problem of static expected equilibrium pricing by dynamically adjusting prices of over or under consumed goods

Our adaptive expected equilibrium pricing approach achieves constant constraint violation and log regret

Algorithm 1: Adaptive Expected Equilibrium Pricing

Input : Initial Good Capacities $\mathbf{c}$, Number of Users n, Threshold Parameter Vector Δ, Support of Probability Distribution $\{\tilde{w}_k, \tilde{\mathbf{u}}_k\}_{k=1}^K$, Occurrence Probabilities $\{q_k\}_{k=1}^K$

Initialize $\mathbf{c}_1 = \mathbf{c}$ and the average remaining good capacity to $\mathbf{d}_1 = \frac{\mathbf{c}}{n}$;

for $t = 1, 2, ..., n$ **do**

Phase I: Set Price

if $\mathbf{d}_{t'} \in [\mathbf{d} - \Delta, \mathbf{d} + \Delta]$ *for all* $t' \leq t$ **then**

Set price $\mathbf{p}^t$ as the dual variables of the capacity constraints of the certainty equivalent problem $CE(\mathbf{d}_t)$ with capacity $\mathbf{d}_t$;

else

Set price $\mathbf{p}^t$ using the dual variables of the capacity constraints of the certainty equivalent problem $CE(\mathbf{d})$ with capacity $\mathbf{d} = \mathbf{d}_1$;

end

Phase II: Observed User Consumption and Update Available Good Capacities

User purchases optimal bundle of goods $\mathbf{x}_t$ given price $\mathbf{p}^t$;

Update the available good capacities $\mathbf{c}_{t+1} = \mathbf{c}_t - \mathbf{x}_t$;

Compute the average remaining good capacities $\mathbf{d}_{t+1} = \frac{\mathbf{c}_{t+1}}{n-t}$;

end

Set price based on dual variable of capacity constraints of certainty equivalent problem

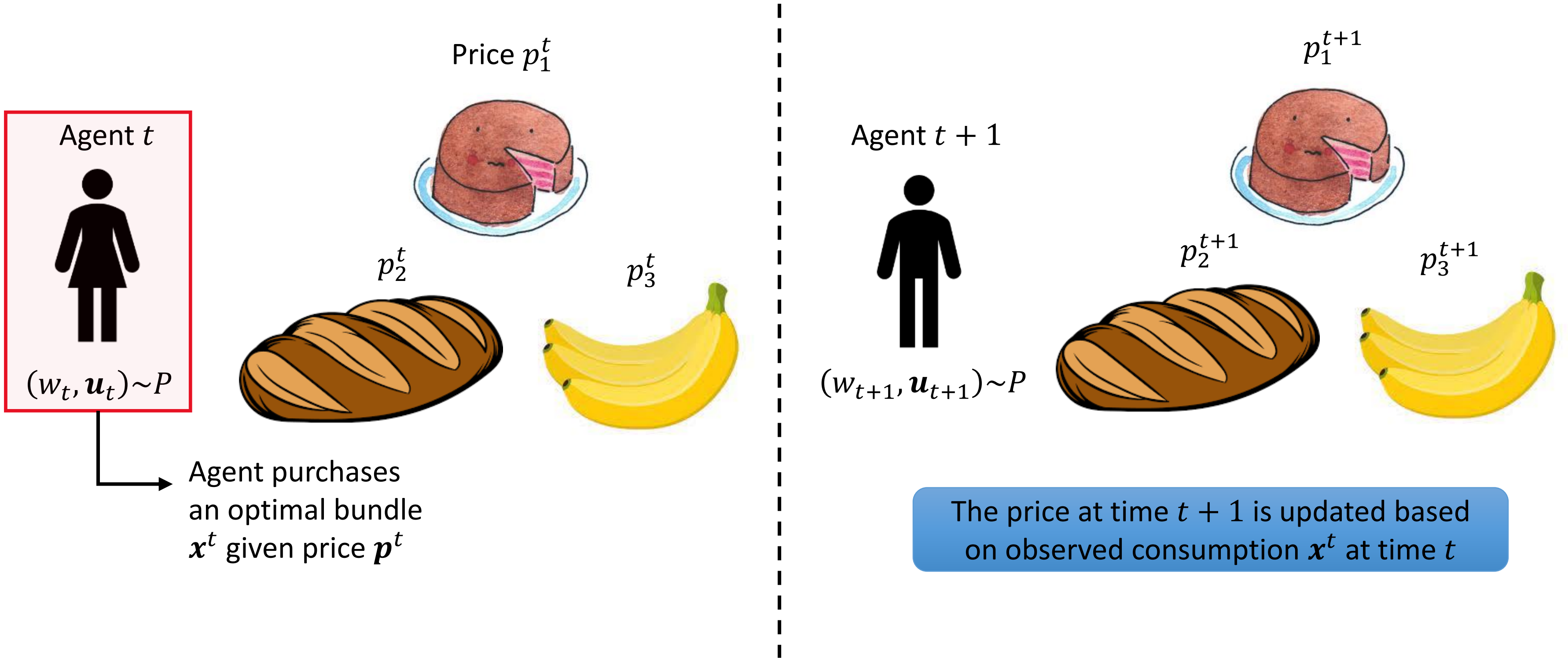
Users consume optimal bundle of goods

Update average remaining resource capacities

Theorem: Under i.i.d. budget and utility parameters with a discrete probability distribution and when good capacities are $O(n)$, Algorithm 1 achieves an expected regret of $R_n(\boldsymbol{\pi}) \leq O(\log(n))$ and expected constraint violation of $V_n(\boldsymbol{\pi}) \leq O(1)$

However, this algorithm required knowledge of the distribution from which users’ utility and budgets are drawn

We design a dual based algorithm, wherein users see prices at each time they arrive



Applying gradient descent to the dual of the social optimization problem motivates a natural algorithm

Dual of social optimization problem with Lagrange multiplier of the capacity constraints p_j

$$\min_{\mathbf{p}} \quad \sum_{t=1}^{n} w_t \log(w_t) - \sum_{t=1}^{n} w_t \log\left(\min_{j\in[m]} \frac{p_j}{u_{tj}}\right) + \sum_{j=1}^{m} p_j c_j - \sum_{t=1}^{n} w_t$$

Equivalent Sample Average Approximation (SAA) of Dual Problem

$$\min_{\mathbf{p}} \quad D_n(\mathbf{p}) = \sum_{j=1}^{m} p_j \frac{c_j}{n} + \frac{1}{n}\sum_{t=1}^{n}\left(w_t \log(w_t) - w_t \log(\min_{j\in[m]} \frac{p_j}{u_{tj}}) - w_t\right)$$

(Sub)-gradient descent of dual problem for each agent: $O(m)$ complexity of price update

$$\partial_{\mathbf{p}}\left(\sum_{j\in[m]} p_j \frac{c_j}{n} + w\log(w) - w\log\left(\min_{j\in[m]}\frac{p_j}{u_j}\right) - w\right)\Bigg|_{\mathbf{p}=\mathbf{p}^t} = \frac{1}{n}\mathbf{c} - \mathbf{x}_t$$

Difference between market share of each agent and goods purchased

We develop a revealed preference algorithm with sub-linear regret and constraint violation guarantees

Algorithm 2: Revealed Preference Algorithm for Online Fisher Markets

Input : Number of users n, Vector of good capacities per user $\mathbf{d} = \frac{\mathbf{c}}{n}$
Initialize $\mathbf{p}^1 > \mathbf{0}$;
for $t = 1, 2, ..., n$ **do**
  Phase I: ;
  User purchases an optimal bundle of goods $\mathbf{x}_t$ given the price $\mathbf{p}^t$;
  Phase II (Price Update): ;
  $\mathbf{p}^{t+1} \leftarrow \mathbf{p}^t - \gamma_t (\mathbf{d} - \mathbf{x}_t)$;
end

Difference between market share of each agent and goods purchased

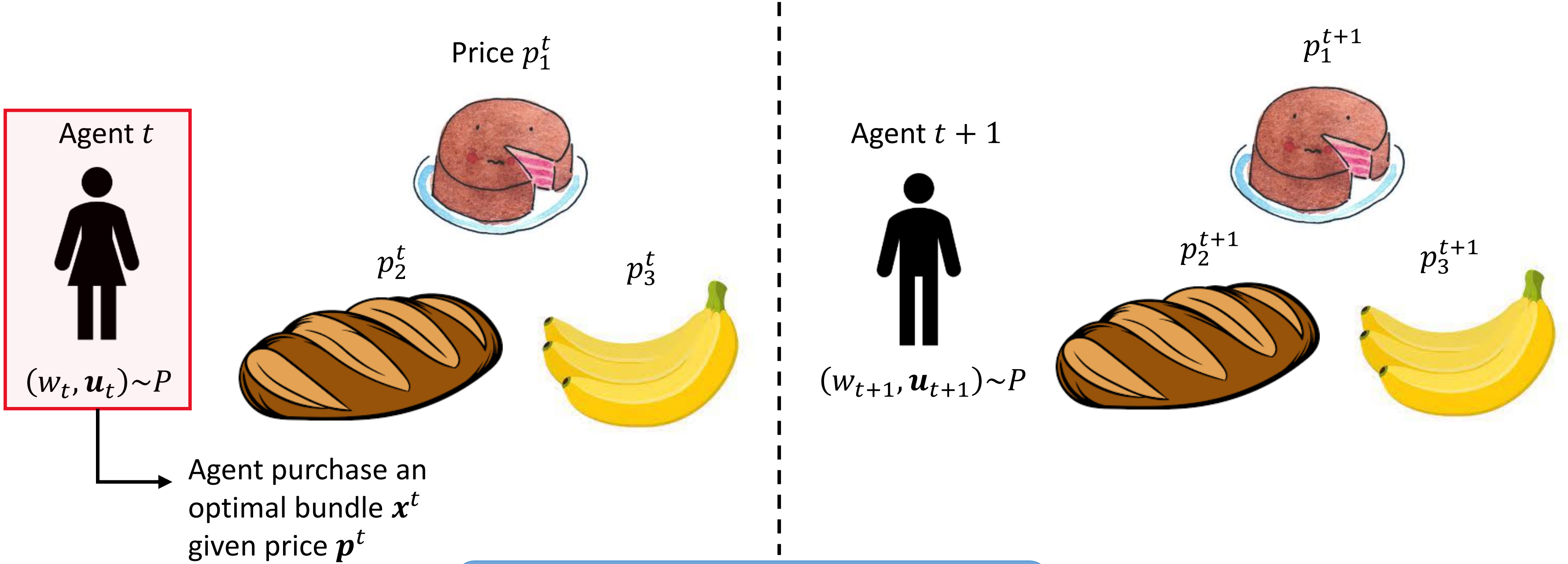
Step-size: $O\left(\frac{1}{\sqrt{n}}\right)$

Only requires knowledge of user consumption (and not their budgets or utilities) to update prices

We believe our results in the online setting for classical Fisher markets may also hold for homogenously constrained Fisher markets

Theorem: Under i.i.d. budget and utility parameters with strictly positive support and when good capacities are $O(n)$, Algorithm 2 achieves an expected regret of $R_n(\boldsymbol{\pi}) \leq O(\sqrt{n})$ and expected constraint violation of $V_n(\boldsymbol{\pi}) \leq O(\sqrt{n})$, where n is the number of arriving users.

Again, the price of a good is increased if the arriving user purchases more than its market share of the good and vice versa



Increase Prices: $p_j^{t+1} > p_j^t$ if $x_j^{t+1} > \frac{c_j}{n}$

Decrease Prices: $p_j^{t+1} < p_j^t$ if $x_j^{t+1} < \frac{c_j}{n}$

Organization

- Fisher Markets with Additional Constraints: Non-convexity
- Distributed Algorithms for Fisher Markets
- Online Algorithms in Stochastic Fisher Markets: Uncertainty
- **Conclusion/Takeaways**

Takeaways: we extended classical Fisher Markets to take into account practical considerations

Resource Allocation under budget, capacity **and physical (e.g., knapsack)** constraints

Jalota, Pavone, Qi, **Ye** GEB'23

Additional constraints introduce **non-convexities**

Yet we derive a **social optimization problem** and **distributed algorithms** to compute prices

Set prices in **Online and Uncertain** variants of Fisher Markets

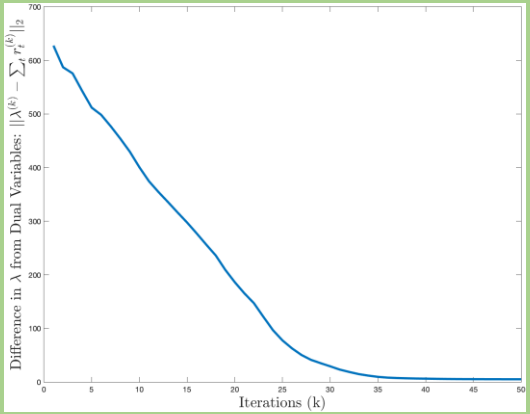
Jalota, **Ye** WINE'23

Static Pricing has performance limitations

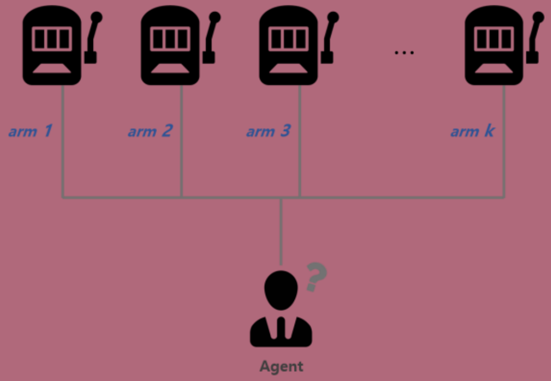
We derive **adaptive/dynamic pricing** approaches with improved performance guarantees

Ongoing and Future Work

Convergence of Fixed Point Scheme



Online Algorithms with Linear Constraints and a Batch Size



Integral Allocations