Markets and Fair Division

Kurt Mehlhorn

based on joint work with Xiaohui Bei, Ran Duan, Jugal Garg, Martin Hoefer

max planck institut
informatik

[1] Saarland Informatics Campus

Outline

- A Simple Market Model: Linear Fisher Markets, Statement and Algorithms.
- Fair Division of Divisible Items: Problem Statement and Connection to Markets.
- Fair Division of Indivisible Items: Approximation Algs and Hardness of Approximation.

Walras' Model of an Economy (Léon Walras 1875)

- Each market participant (agent) owns some goods and
- has preferences over goods, i.e.,

  at a given set of prices, certain bundles of goods will give maximum pleasure (utility).

  Agents are only willing to buy bundles that give maximum utility.
- Question: are there prices such that all goods can be completely sold and agents spend all their income, i.e.

  can a perfect exchange be organized through appropriate prices?

Linear Utilities: A Special Case

- Twice as much is twice as good,

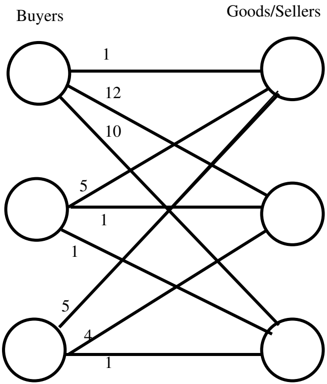
  marginal utilities do no decrease.

- Utilities from different goods add up.
- Example: suppose a bottle of champagne gives me three times the pleasure of a bottle of wine.

  If the price of champagne is more than three times the price of wine, I am **not** willing to buy champagne.

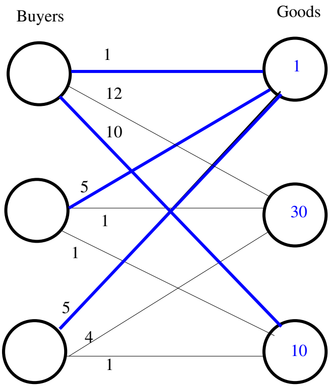
  If the price is exactly three times the price of wine, I am willing to buy champagne and wine and any combination is equally fine.

Example



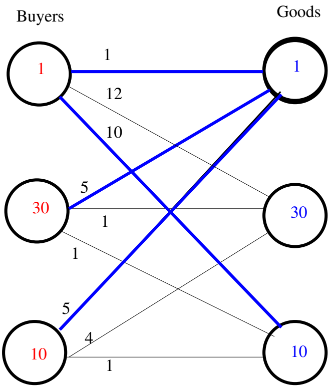
- First agent values second good 12 times as much as first good, . . .
- Assume i-th agent owns i-th good, one unit of each good. Goods are divisible.

Example



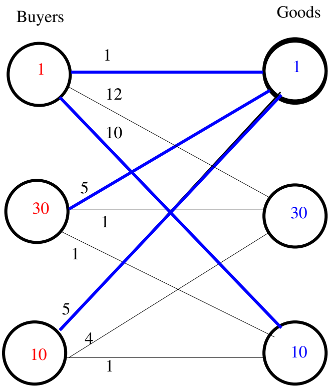
- First agent values second good 12 times as much as first good, ...
- Assume i-th agent owns i-th good, one unit of each good. Goods are divisible.
- If prices are as shown in blue, money will only flow along blue edges (maximum bang-per-buck edges (MBB edges)).

Example



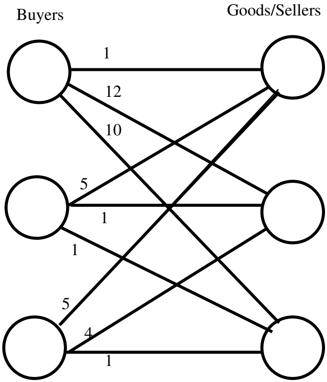
- First agent values second good 12 times as much as first good, ...
- Assume *i*-th agent owns *i*-th good, one unit of each good. Goods are divisible.
- If prices are as shown in blue, money will flow only along the blue edges (MBB edges).
- If goods are completely sold, the red budgets will be available to the agents,

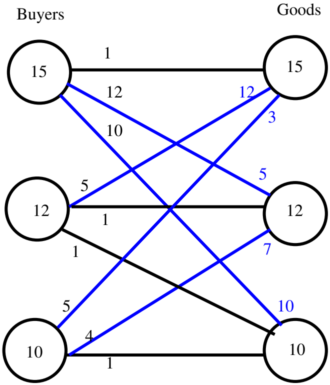
Example



- First agent values second good 12 times as much as first good, . . .
- Assume i-th agent owns i-th good, one unit of each good. Goods are divisible.
- If prices are as shown in blue, money will flow only along the blue edges (MBB edges).
- If goods are completely sold, the red budgets will be available to the agents,
- but the second good will certainly not be completely sold, because nobody is interested in it.

Example (A Solution)



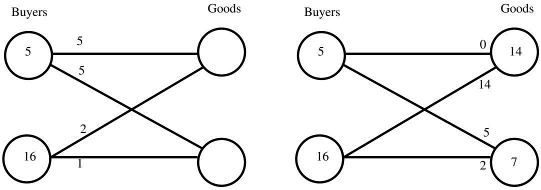


Utilities in black, prices inside nodes, bang-for-buck edges and flow of money in blue.

A Special Case: The Fisher Market (Fisher 1891)

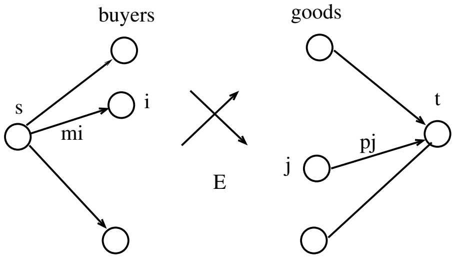
- buyer i has budget m_i, $m_i \geq 0$
- u_{ij} = utility for i if all of good j is allocated to him
- are there prices p_j, $1 \leq j \leq n$, and allocations x_{ij} such that
  - all goods are completely sold: $\sum_i x_{ij} = 1$
  - all money is spent: $\sum_j x_{ij} p_j = m_i$
  - only bang per buck items are bought:

$$x_{ij} > 0 \quad \Rightarrow \quad \frac{u_{ij}}{p_j} = \alpha_i, \text{ where } \alpha_i = \max_\ell \frac{u_{i\ell}}{p_\ell}$$



The MBB-Network G_p (Fisher Market)

Vertices: buyers b_i and goods c_j, source s and sink t



Edges:

(s, b_i) with capacity m_i

(b_i, c_j) iff $u_{ij}/p_j = \alpha_i$, unlimited capacity

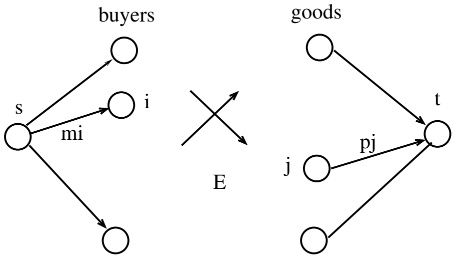
(c_j, t) with capacity p_j

flow on edge (b_i, c_j) = money paid by buyer b_i for his fraction of good c_j

p is an equilbrium iff a max flow saturates all edges out of s and into t.

The MBB-Network G_p (Fisher Market)

Vertices: buyers b_i and goods c_j, source s and sink t



Edges:

(s, b_i) with capacity m_i

(b_i, c_j) iff $u_{ij}/p_j = \alpha_i$, unlimited capacity

(c_j, t) with capacity p_j

flow on edge (b_i, c_j) = money paid by buyer b_i for his fraction of good c_j

p is an equilbrium iff a max flow saturates all edges out of s and into t.

Questions

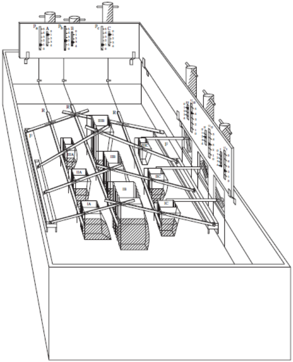
- Do equilibria exist?
- Properties of equilibria: is there a rational equilibrium? do equilibria form a convex set?
- Algorithms:
  - approximation, exact
  - efficient
  - combinatorial or do we need ellipsoid and/or interior point
  - global knowledge versus local knowledge
  - natural updates (tatonnement)

History

Walras introduces the model in 1875 (more general utilities) and argues existence (iterative adaption of prices).

Figure 5

Fisher's apparatus.



Fisher (1891), simpler model (buyers have budgets), alg for three buyers/goods.

Wald (36) shows existence of equ. under strong assumptions.

Arrow/Debreu (54) show existence for a much more general model under mild assumptions.

Existence proofs are non-constructive (fixpoint theorems).

Algorithm Development

- Algorithm development starts in the 60s: Scarf, Smale, Kuhn, Todd, Eaves.
- Early algorithms are inspired by fixed-point proofs or are Newton-based and compute approximations, are exponential time.
- Exact poly-time combinatorial algorithms are known for the Fisher market (Devanur/Padimitriou/Saberi/Varzirani (08) and Orlin (10)) and for the Arrow-Debreu market (Duan/Mehlhorn (12) and Duan/Garg/Mehlhorn (13)). Orlin even gave a strongly polynomial algorithm for the Fisher market.
- I show you a poly-time alg for Fisher markets inspired by an algorithm for Arrow-Debreu markets due to Duan/Mehlhorn.

Overview

intialize all prices to one: $p_j = 1$ for all j

repeat

construct the network G_p for the current prices p and compute a balanced flow f in it;

increase some prices and adjust flow;

until the total surplus is tiny (less than $O(\frac{1}{4n^4U^{3n}})$);

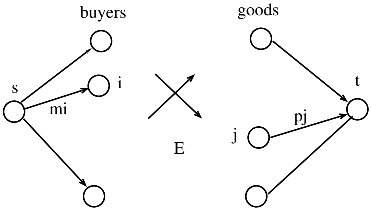
round the current prices to the equilibrium prices;

Details of final rounding: Let p be the current price vector;
let q_i be the rational closest to p_i with denominator $\leq (nU)^n$.

Then $q = (q_1, \ldots, q_n)$ is a vector of equilibrium prices.

The Flow Network G_p, Revisited

- Vertices b_i, c_i, $1 \leq i \leq n$, s and t
- Edges $E_p = \{(b_i, c_j) \mid u_{ij}/p_j = \alpha_i := \max_\ell u_{i\ell}/p_\ell\}$, capacity ∞
- Let f be a maximum flow

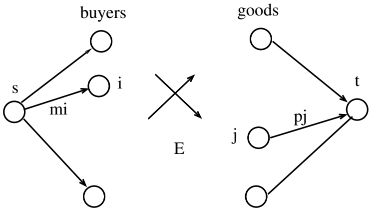


- $r(b_i) = p_i - \sum_j f_{ij}$, surplus of buyer i
- $r(B) = (r(b_1), \ldots, r(b_n))$, surplus vector

- Balanced flow = maxflow minimizing $\|r(B)\| = \sqrt{r(b_1)^2 + \ldots + r(b_n)^2}$;
- Intuiton: As long as there is a good having inflow from two buyers with unequal surplus, balance.
- Can be computed with n maxflow computations (Devanur et al)

The Flow Network G_p, Revisited

- Vertices b_i, c_i, $1 \le i \le n$, s and t
- Edges $E_p = \{(b_i, c_j) \mid u_{ij}/p_j = \alpha_i := \max_\ell u_{i\ell}/p_\ell\}$, capacity ∞
- Let f be a maximum flow



- $r(b_i) = p_i - \sum_j f_{ij}$, surplus of buyer i
- $r(B) = (r(b_1), \ldots, r(b_n))$, surplus vector

- Balanced flow = maxflow minimizing $\|r(B)\| = \sqrt{r(b_1)^2 + \ldots + r(b_n)^2}$;
- Intuiton: As long as there is a good having inflow from two buyers with unequal surplus, balance.
- Can be computed with n maxflow computations (Devanur et al)

Intuition

- Let f be a balanced flow, order buyers $r(b_1) \geq r(b_2) \geq \ldots \geq r(b_n) \geq r(b_{n+1}) := 0$.
- Let ℓ be minimal such $r(b_\ell)/r(b_{\ell+1}) \geq 1 + 1/n$, let
  $S = \{b_1, \ldots, b_\ell\}$, and — big surplus buyers
  $C(S) = \{c_j \mid b_i \in S \text{ and } (i,j) \in E\}$. — goods bought by them
- All flow from the buyers in S goes to the goods in $C(S)$. Hence the goods in $C(S)$ are completely sold.
- Natural action: increase the prices of the goods in $C(S)$ and the flow into them by a common factor x.
- What is the effect of this?

Intuition

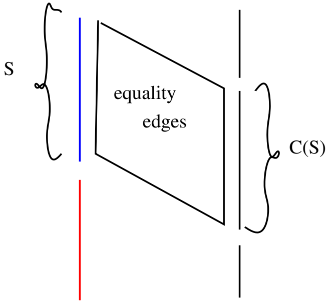
- Let f be a balanced flow, order buyers
  $r(b_1) \geq r(b_2) \geq \ldots \geq r(b_n) \geq r(b_{n+1}) := 0$.
- Let ℓ be minimal such $r(b_\ell)/r(b_{\ell+1}) \geq 1 + 1/n$, let
  $S = \{b_1, \ldots, b_\ell\}$, and — big surplus buyers
  $C(S) = \{c_j \mid b_i \in S \text{ and } (i,j) \in E\}$. — goods bought by them
- All flow from the buyers in S goes to the goods in $C(S)$. Hence the goods in $C(S)$ are completely sold.
- Natural action: increase the prices of the goods in $C(S)$ and the flow into them by a common factor x.
- What is the effect of this?

Price Update

$f =$ a balanced flow

$S = \{b_1, \ldots, b_\ell\}$, and big surplus buyers

$C(S) = \{c_j \mid b_i \in S \text{ and } (i,j) \in E\}$. goods bought by them



- There is no flow from $B \setminus S$ to $C(S)$.
- Goods in $C(S)$ are completely sold.
- Increase prices of goods in $C(S)$ and flow into these vertices by a factor $x > 1$.
- Surplus goes down, surplus unchanged.
- Goods outside $C(S)$ become more attractive for buyers in S.
- Goods in $C(S)$ keep surplus zero; goods with non-zero surplus have price one

Constraints on x

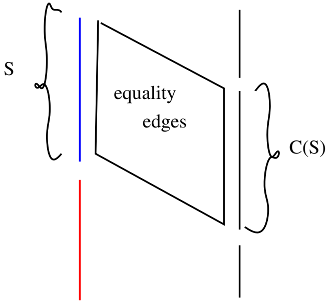
- A new MBB edge arises connecting a buyer in S with a good in $C \setminus C(S)$.
- A blue surplus becomes zero.

Price Update

f = a balanced flow

$S = \{b_1, \ldots, b_\ell\}$, and big surplus buyers

$C(S) = \{c_j \mid b_i \in S \text{ and } (i,j) \in E\}$. goods bought by them



- There is no flow from $B \setminus S$ to $C(S)$.
- Goods in $C(S)$ are completely sold.
- Increase prices of goods in $C(S)$ and flow into these vertices by a factor $x > 1$.
- Surplus goes down, surplus unchanged.
- Goods outside $C(S)$ become more attractive for buyers in S.
- Goods in $C(S)$ keep surplus zero; goods with non-zero surplus have price one

Constraints on x

- A new MBB edge arises connecting a buyer in S with a good in $C \setminus C(S)$.
- A blue surplus becomes zero.

The Complete Algorithm

intialize prices: $p_j = 1$ for all j

repeat

construct the network G for the current prices and compute a balanced flow f in it;

order buyers by surplus and let ℓ be minimal such that $r(b_\ell) > (1 + 1/n)r(b_{\ell+1})$. Let $S = \{b_1, \ldots, b_\ell\}$.

increase prices of goods in $C(S)$ and flows into these goods by gradually increasing factor x until

new equality edge or

surplus of a buyer in S and a buyer in B becomes zero.

until the total surplus is tiny (less than $O(\frac{1}{4n^4 U^{3n}})$);

round the current prices to the equilibrium prices;

Key Lemmas

Prices stay bounded by $M := \sum_i m_i$.

Norm of surplus vector decreases by factor $1 + \Omega(1/n^3)$ in each iteration.

Norm is at most n initially. We stop when norm is less than $1/(4n^4 U^{3n})$. Hence the number of iterations is at most L, where

$$n \cdot (1 + \frac{1}{n^3})^L \leq \frac{1}{n^4 U^{3n}}.$$

Thus

$$L = O(n^3 \log(nU)).$$

Number of arithmetic operations is $O(L \cdot n \cdot n^3) = O(n^7 \log(nU))$.

It suffices to compute with integers with $O(n \log(nU))$ bits.

Key Lemmas

Prices stay bounded by $M := \sum_i m_i$.

Norm of surplus vector decreases by factor $1 + \Omega(1/n^3)$ in each iteration.

Norm is at most n initially. We stop when norm is less than $1/(4n^4 U^{3n})$. Hence the number of iterations is at most L, where

$$n \cdot (1 + \frac{1}{n^3})^L \leq \frac{1}{n^4 U^{3n}}.$$

Thus

$$L = O(n^3 \log(nU)).$$

Number of arithmetic operations is $O(L \cdot n \cdot n^3) = O(n^7 \log(nU))$.

It suffices to compute with integers with $O(n \log(nU))$ bits.

Key Lemmas

Prices stay bounded by $M := \sum_i m_i$.

Norm of surplus vector decreases by factor $1 + \Omega(1/n^3)$ in each iteration.

Norm is at most n initially. We stop when norm is less than $1/(4n^4 U^{3n})$. Hence the number of iterations is at most L, where

$$n \cdot (1 + \frac{1}{n^3})^L \le \frac{1}{n^4 U^{3n}}.$$

Thus

$$L = O(n^3 \log(nU)).$$

Number of arithmetic operations is $O(L \cdot n \cdot n^3) = O(n^7 \log(nU))$.

It suffices to compute with integers with $O(n \log(nU))$ bits.

Key Lemmas

Prices stay bounded by $M := \sum_i m_i$.

Norm of surplus vector decreases by factor $1 + \Omega(1/n^3)$ in each iteration.

Norm is at most n initially. We stop when norm is less than $1/(4n^4U^{3n})$.
Hence the number of iterations is at most L, where

$$n \cdot (1 + \frac{1}{n^3})^L \leq \frac{1}{n^4 U^{3n}}.$$

Thus

$$L = O(n^3 \log(nU)).$$

Number of arithmetic operations is $O(L \cdot n \cdot n^3) = O(n^7 \log(nU))$.

It suffices to compute with integers with $O(n \log(nU))$ bits.

Key Lemmas

Prices stay bounded by $M := \sum_i m_i$.

Norm of surplus vector decreases by factor $1 + \Omega(1/n^3)$ in each iteration.

Norm is at most n initially. We stop when norm is less than $1/(4n^4 U^{3n})$.
Hence the number of iterations is at most L, where

$$n \cdot (1 + \frac{1}{n^3})^L \leq \frac{1}{n^4 U^{3n}}.$$

Thus

$$L = O(n^3 \log(nU)).$$

Number of arithmetic operations is $O(L \cdot n \cdot n^3) = O(n^7 \log(nU))$.

It suffices to compute with integers with $O(n \log(nU))$ bits.

Norm of surplus vector decreases by a factor $1 + \Omega(1/n^3)$ in each iteration.

- The buyers in S have the same surplus up to a factor of e.
- If the surplus of a buyer becomes zero, the norm drops by a factor $1 + \Omega(1/n)$.
- If a new MBB-edge arises, we
  - first push flow into the good until it is completely sold (happens only m times) or surplus of buyer is zero and then
  - balance flow. This will make two surpluses which were at a factor $1 + 1/n$ appart equal.

Assignment of Items to Agents with Valuations

- Set G of m (in)divisible items
- Set A of n agents or users
- u_{ij} = utility of good j to agent i
- Allocation
  x_{ij} = fraction of good j assigned to agent i, $x_{ij} \geq 0$.
- No item is overassigned, i.e., $\sum_i x_{ij} \leq 1$.
- $u_i = \sum_j u_{ij} x_{ij}$, utility of agent i.
- indivisible: $x_{ij} \in \{0, 1\}$.

What constitutes a fair division?

Assignment of Items to Agents with Valuations

- Set G of m (in)divisible items
- Set A of n agents or users
- u_{ij} = utility of good j to agent i

- Allocation
  x_{ij} = fraction of good j assigned to agent i, $x_{ij} \geq 0$.
- No item is overassigned, i.e., $\sum_i x_{ij} \leq 1$.
- $u_i = \sum_j u_{ij} x_{ij}$, utility of agent i.
- indivisible: $x_{ij} \in \{0, 1\}$.

What constitutes a fair division?

Which Objective Catches Fairness?

Maximize the arithmetic mean of valuations
Utilitarian Social Welfare:

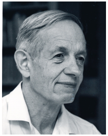
$$\mathrm{SW}(S) = \frac{1}{n} \sum_{i \in A} u_i$$

Maximize the minimum of valuations
Max-Min-Fairness, Egalitarian Welfare:

$$\mathrm{EW}(S) = \min_{i \in A} u_i$$

Maximize the geometric mean of valuations
Proportional Fairness, Nash Social Welfare:

$$\mathrm{NSW}(S) = \left(\prod_{i \in A} u_i \right)^{1/n}$$



What do the numbers u_{ij} mean? Does it make sense to compare u_{ij} and u_{ik}? u_{ij} and u_{hj}?

Which Objective Catches Fairness?

Maximize the arithmetic mean of valuations
Utilitarian Social Welfare:

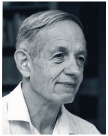
$$\mathrm{SW}(S) = \frac{1}{n} \sum_{i \in A} u_i$$

Maximize the minimum of valuations
Max-Min-Fairness, Egalitarian Welfare:

$$\mathrm{EW}(S) = \min_{i \in A} u_i$$

Maximize the geometric mean of valuations
Proportional Fairness, Nash Social Welfare:

$$\mathrm{NSW}(S) = \left(\prod_{i \in A} u_i \right)^{1/n}$$



What do the numbers u_{ij} mean? Does it make sense to compare u_{ij} and u_{ik}? u_{ij} and u_{hj}?

Which Objective Catches Fairness?

Maximize the arithmetic mean of valuations
Utilitarian Social Welfare:

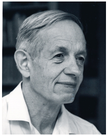
$$\mathrm{SW}(S) = \frac{1}{n} \sum_{i \in A} u_i$$

Maximize the minimum of valuations
Max-Min-Fairness, Egalitarian Welfare:

$$\mathrm{EW}(S) = \min_{i \in A} u_i$$

Maximize the geometric mean of valuations
Proportional Fairness, Nash Social Welfare:

$$\mathrm{NSW}(S) = \left(\prod_{i \in A} u_i \right)^{1/n}$$



What do the numbers u_{ij} mean? Does it make sense to compare u_{ij} and u_{ik}? u_{ij} and u_{hj}?

Which Objective Catches Fairness?

Maximize the arithmetic mean of valuations
Utilitarian Social Welfare:

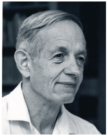
$$\mathrm{SW}(S) = \frac{1}{n} \sum_{i \in A} u_i$$

Maximize the minimum of valuations
Max-Min-Fairness, Egalitarian Welfare:

$$\mathrm{EW}(S) = \min_{i \in A} u_i$$

Maximize the geometric mean of valuations
Proportional Fairness, Nash Social Welfare:

$$\mathrm{NSW}(S) = \left(\prod_{i \in A} u_i \right)^{1/n}$$

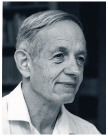


What do the numbers u_{ij} mean? Does it make sense to compare u_{ij} and u_{ik}? u_{ij} and u_{hj}?

Axiomatic Justification of Nash Social Welfare

Maximize the geometric mean of valuations
Proportional Fairness, Nash Social Welfare:

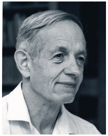
$$\mathrm{NSW}(S) = \left(\prod_{i \in A} u_i\right)^{1/n}$$



Nash (1952): NSW is the only objective that satisfies Pareto optimiality, invariance under scaling, symmetry, and independence of irrelevant alternatives.

Axiomatic Justification of Nash Social Welfare

Maximize the geometric mean of valuations
Proportional Fairness, Nash Social Welfare:



$$\mathrm{NSW}(S) = \left(\prod_{i \in A} u_i\right)^{1/n}$$

Nash (1952): NSW is the only objective that satisfies Pareto optimiality, invariance under scaling, symmetry, and independence of irrelevant alternatives.

Formulation as a Mathematical Program

Formulation as a Non-Linear Optimization Problem:

$$\begin{array}{rrcll} \text{Max.} & \multicolumn{4}{l}{\left(\prod_{i\in A}\sum_{j\in G} u_{ij}x_{ij}\right)^{1/n}} \\ \text{s.t.} & \sum_{i\in A} x_{ij} & \leq & 1 & j\in G \\ & x_{ij} & \geq & 0 & i\in A,\ j\in G \end{array}$$

Formulation as a Mathematical Program

After taking logarithm of the objective (recall that logarithm function is monotone).

$$
\begin{array}{rrcll}
\text{Max.} & \frac{1}{n} \sum_{i \in A} \log \left(\sum_{j \in G} u_{ij} x_{ij} \right) & & & \\
\text{s.t.} & \sum_{i \in A} x_{ij} & \leq & 1 & j \in G \\
& x_{ij} & \geq & 0 & i \in A,\ j \in G
\end{array}
$$

Eisenberg-Gale Convex Program: [Eisenberg, Gale, Ann Math Stat'59] [Gale 1960], [Eisenberg, Mgmt Sci'61]

$$\begin{aligned} \text{Max.} \quad & \frac{1}{n} \sum_{i \in A} \log \left(\sum_{j \in G} u_{ij} x_{ij} \right) \\ \text{s.t.} \quad & \sum_{i \in A} x_{ij} \leq 1 \quad j \in G \\ & x_{ij} \geq 0 \quad i \in A,\ j \in G \end{aligned}$$

First formulated in the connection with Fisher Markets.
Optimal solutions: Competitive Equilibria with Equal Incomes (CEEI), [Varian, JET '74]

Constrained Optimization

$$\begin{array}{rlll} \text{Min.} & f(x) & & \\ \text{s.t.} & h_i(x) \leq 0 & i = 1, 2, 3, \ldots \\ & g_j(x) = 0 & j = 1, 2, 3, \ldots \end{array}$$

KKT conditions, Method of Lagrange Multipliers: In an optimal point, the gradient (vector of partial derivatives) of f is a linear combination of the gradients of the tight constraints. The multipliers have to be non-negative for the inequalities and are unconstrained otherwise. I.e.,

$$\nabla f(x) = \sum_i \lambda_i \nabla h_i(x) + \sum_j \mu_j \nabla g_j(x)$$

$$\begin{array}{lrcll} \text{s.t.} & h_i(x) & \leq & 0 & i = 1, 2, 3, \ldots \\ & g_j(x) & = & 0 & j = 1, 2, 3, \ldots \\ & \lambda_i & \geq & 0 & i = 1, 2, 3, \ldots \\ & \lambda_i > 0 & \text{implies} & h_i(x) = 0 & \end{array}$$

Optimality Conditions for our Mathematical Program

$$
\begin{array}{llllll}
\text{Max.} & \frac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G} u_{ij}x_{ij}\right) & & & & \\
\text{s.t.} & \sum_{i\in A} x_{ij} & \le & 1 & j\in G & \text{dual variable } p_j \\
 & -x_{ij} & \le & 0 & i\in A,\ j\in G & \text{dual variable } q_{ij}
\end{array}
$$

Optimality Conditions: write u_i for $\sum_{j\in G} u_{ij}x_{ij}$.

$$
\begin{array}{rcll}
u_{ij}/u_i & = & p_j - q_{ij} & \text{derivative with respect to } x_{ij} \\
p_j & \ge & 0 & \\
q_{ij} & \ge & 0 & \\
\sum_{i\in A} x_{ij} & \le & 1 & \\
-x_{ij} & \le & 0 & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \\
\sum_{i\in A} x_{ij} < 1 & \text{implies} & p_j = 0 &
\end{array}
$$

Optimality Conditions for our Mathematical Program

$$\begin{array}{llcll}
\text{Max.} & \dfrac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G}u_{ij}x_{ij}\right) & & & \\
\text{s.t.} & \sum_{i\in A}x_{ij} & \leq & 1 & j\in G & \text{dual variable } p_j \\
 & -x_{ij} & \leq & 0 & i\in A,\ j\in G & \text{dual variable } q_{ij}
\end{array}$$

Optimality Conditions: write u_i for $\sum_{j\in G}u_{ij}x_{ij}$.

$$\begin{array}{rcll}
u_{ij}/u_i & = & p_j - q_{ij} & \text{derivative with respect to } x_{ij} \\
p_j & \geq & 0 & \text{price of good } j \\
q_{ij} & \geq & 0 & u_{ij}/u_i \leq p_j \text{ or } u_{ij}/p_j \leq u_i \text{ always} \\
\sum_{i\in A}x_{ij} & \leq & 1 & \\
-x_{ij} & \leq & 0 & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \\
\sum_{i\in A}x_{ij} < 1 & \text{implies} & p_j = 0 &
\end{array}$$

Optimality Conditions for our Mathematical Program

$$
\begin{array}{llll}
\text{Max.} & \frac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G} u_{ij}x_{ij}\right) & & \\
\text{s.t.} & \sum_{i\in A} x_{ij} \leq 1 & j\in G & \text{dual variable } p_j \\
 & -x_{ij} \leq 0 & i\in A,\ j\in G & \text{dual variable } q_{ij}
\end{array}
$$

Optimality Conditions: write u_i for $\sum_{j\in G} u_{ij}x_{ij}$.

$$
\begin{array}{rcll}
u_{ij}/u_i & = & p_j - q_{ij} & \text{derivative with respect to } x_{ij} \\
p_j & \geq & 0 & \text{price of good } j \\
q_{ij} & \geq & 0 & u_{ij}/u_i \leq p_j \text{ or } u_{ij}/p_j \leq u_i \text{ always} \\
\sum_{i\in A} x_{ij} & \leq & 1 & \\
-x_{ij} & \leq & 0 & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \text{if } x_{ij} > 0 \text{ then } u_{ij}/u_i = p_j \text{ and hence } u_{ij}/p_j = u_i \\
\sum_{i\in A} x_{ij} < 1 & \text{implies} & p_j = 0 &
\end{array}
$$

Optimality Conditions for our Mathematical Program

$$
\begin{array}{llllll}
\text{Max.} & \displaystyle\frac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G} u_{ij}x_{ij}\right) & & & & \\
\text{s.t.} & \displaystyle\sum_{i\in A} x_{ij} & \leq & 1 & j \in G & \text{dual variable } p_j \\
 & -x_{ij} & \leq & 0 & i \in A,\ j \in G & \text{dual variable } q_{ij}
\end{array}
$$

Optimality Conditions: write u_i for $\sum_{j\in G} u_{ij}x_{ij}$.

$$
\begin{array}{rcll}
u_{ij}/u_i & = & p_j - q_{ij} & \text{derivative with respect to } x_{ij} \\
p_j & \geq & 0 & \text{price of good } j \\
q_{ij} & \geq & 0 & u_{ij}/u_i \leq p_j \text{ or } u_{ij}/p_j \leq u_i \text{ always} \\
\displaystyle\sum_{i\in A} x_{ij} & \leq & 1 & \\
-x_{ij} & \leq & 0 & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \text{if } x_{ij} > 0 \text{ then } u_{ij}/u_i = p_j \text{ and hence } u_{ij}/p_j = u_i \\
\displaystyle\sum_{i\in A} x_{ij} < 1 & \text{implies} & p_j = 0 & \text{if good } j \text{ is not fully allocated, its price is zero.}
\end{array}
$$

Optimality Conditions for our Mathematical Program

$$
\begin{array}{llllll}
\text{Max.} & \frac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G} u_{ij}x_{ij}\right) & & & & \\
\text{s.t.} & \sum_{i\in A} x_{ij} & \leq & 1 & j\in G & \text{dual variable } p_j \\
 & -x_{ij} & \leq & 0 & i\in A,\ j\in G & \text{dual variable } q_{ij}
\end{array}
$$

Optimality Conditions: write u_i for $\sum_{j\in G} u_{ij}x_{ij}$.

$$
\begin{array}{rcll}
u_{ij}/u_i & = & p_j - q_{ij} & \text{derivative with respect to } x_{ij} \\
p_j & \geq & 0 & \text{price of good } j \\
q_{ij} & \geq & 0 & u_{ij}/u_i \leq p_j \text{ or } u_{ij}/p_j \leq u_i \text{ always} \\
\sum_{i\in A} x_{ij} & \leq & 1 & \\
-x_{ij} & \leq & 0 & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \text{if } x_{ij} > 0 \text{ then } u_{ij}/u_i = p_j \text{ and hence } u_{ij}/p_j = u_i \\
\sum_{i\in A} x_{ij} < 1 & \text{implies} & p_j = 0 & \text{if good } j \text{ is not fully allocated, its price is zero.}
\end{array}
$$

Further consequences: $\sum_j x_{ij}p_j = \sum_j x_{ij}u_{ij}/u_i = 1$, i.e., each agent spends exactly one unit of money and $p_j > 0$ iff $u_{ij} > 0$ for some i.

This is a Fisher Market for Incomes $m_i = 1$ for all Agents

$$\begin{array}{rclll}
u_{ij}/u_i & = & p_j - q_{ij} & & \text{derivative with respect to } x_{ij} \\
p_j & \geq & 0 & & \text{price of good } j \\
q_{ij} & \geq & 0 & & u_{ij}/u_i \leq p_j \text{ or } u_{ij}/p_j \leq u_i \text{ always} \\
\sum_{i \in A} x_{ij} & \leq & 1 & & \\
-x_{ij} & \leq & 0 & & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \text{if } x_{ij} > 0 \text{ then } u_{ij}/u_i = p_j \text{ and hence } u_{ij}/p_j = u_i & \\
\sum_{i \in A} x_{ij} < 1 & \text{implies} & p_j = 0 & \text{if good } j \text{ is not fully allocated, its price is zero.} &
\end{array}$$

Further consequence: $\sum_j x_{ij} p_j = \sum_j x_{ij} u_{ij}/u_i = 1$, i.e., each agent spends exactly one unit of money, and $p_j > 0$ iff $u_{ij} > 0$ for some i. Alternatively:

- buyer i has budget m_i, $m_i = 1$.
- are there prices p_j, $1 \leq j \leq m$, and allocations x_{ij} such that
  - all goods are completely sold: $\sum_i x_{ij} = 1$
  - all money is spent: $\sum_j x_{ij} p_j = m_i$
  - only bang per buck items are bought:

$$x_{ij} > 0 \quad \Rightarrow \quad \frac{u_{ij}}{p_j} = \alpha_i, \text{ where } \alpha_i = \max_\ell \frac{u_{i\ell}}{p_\ell}$$

This is a Fisher Market for Incomes $m_i = 1$ for all Agents

$$\begin{array}{rcll}
u_{ij}/u_i & = & p_j - q_{ij} & \text{derivative with respect to } x_{ij} \\
p_j & \geq & 0 & \text{price of good } j \\
q_{ij} & \geq & 0 & u_{ij}/u_i \leq p_j \text{ or } u_{ij}/p_j \leq u_i \text{ always} \\
\sum_{i \in A} x_{ij} & \leq & 1 & \\
-x_{ij} & \leq & 0 & \\
x_{ij} > 0 & \text{implies} & q_{ij} = 0 & \text{if } x_{ij} > 0 \text{ then } u_{ij}/u_i = p_j \text{ and hence } u_{ij}/p_j = u_i \\
\sum_{i \in A} x_{ij} < 1 & \text{implies} & p_j = 0 & \text{if good } j \text{ is not fully allocated, its price is zero.}
\end{array}$$

Further consequence: $\sum_j x_{ij} p_j = \sum_j x_{ij} u_{ij}/u_i = 1$, i.e., each agent spends exactly one unit of money, and $p_j > 0$ iff $u_{ij} > 0$ for some i. Alternatively:

- buyer i has budget m_i, $m_i = 1$.
- are there prices p_j, $1 \leq j \leq m$, and allocations x_{ij} such that
  - all goods are completely sold: $\sum_i x_{ij} = 1$
  - all money is spent: $\sum_j x_{ij} p_j = m_i$
  - only bang per buck items are bought:

$$x_{ij} > 0 \quad \Rightarrow \quad \frac{u_{ij}}{p_j} = \alpha_i, \text{ where } \alpha_i = \max_{\ell} \frac{u_{i\ell}}{p_\ell}$$

Approximating Nash Social Welfare for Indivisible Goods

Algorithm *ALG* computes a ρ-approximation if for every problem instance I

$$\mathrm{NSW}(ALG(I)) \geq \frac{\mathrm{NSW}(S^*)}{\rho} .$$

Extensions of the 2-approximation algorithm:

- Additive-separable concave valuations [Anari, Mai, Oveis Gharan, Vazirani, 2016]
- Multiple copies of each item [Bei, Garg, Hoefer, Mehlhorn, SAGT'17]
- budget-additive valuations $u_i = \min\left(c_i, \sum_j u_{ij}x_{ij}\right)$, where c_i is a utility cap of agent i. [Garg, Hoefer, Mehlhorn, SODA 2018]

Approximating Nash Social Welfare for Indivisible Goods

Additive Valuations:

$$u_i = \sum_j u_{ij} x_{ij}$$

- APX-hard, no 1.00008-approximation unless $\mathrm{P} = \mathrm{NP}$ [Lee, IPL'17]
- 2.889-approximation via markets [Cole, Gkatzelis, STOC'15]
- e-approximation via stable polynomials [Anari, Gharan, Singh, Saberi, ITCS'17]
- 2-approximation via markets
  [Cole, Devanur, Gkatzelis, Jain, Mai, Vazirani, Yazdanbod, EC'17]
- 1.45-approximation via limited envy [Barman, Krishnamurthy, Vaish, 2017]
  [Caragiannis, Kurokawa, Moulin, Procaccia, Shah, Wang, EC'16]
- In every case: define a suitable relaxation, solve it, and round the solution.

Extensions of the 2-approximation algorithm:

- Additive-separable concave valuations [Anari, Mai, Oveis Gharan, Vazirani, 2016]
- Multiple copies of each item [Bei, Garg, Hoefer, Mehlhorn, SAGT'17]
- budget-additive valuations $u_i = \min\left(c_i, \sum_j u_{ij} x_{ij}\right)$, where c_i is a utility cap of agent i. [Garg, Hoefer, Mehlhorn, SODA 2018]

Approximating Nash Social Welfare for Indivisible Goods

Additive Valuations:

$$u_i = \sum_j u_{ij} x_{ij}$$

- APX-hard, no 1.00008-approximation unless $P = NP$ [Lee, IPL'17]
- 2.889-approximation via markets [Cole, Gkatzelis, STOC'15]
- e-approximation via stable polynomials [Anari, Gharan, Singh, Saberi, ITCS'17]
- 2-approximation via markets [Cole, Devanur, Gkatzelis, Jain, Mai, Vazirani, Yazdanbod, EC'17]
- 1.45-approximation via limited envy [Barman, Krishnamurthy, Vaish, 2017] [Caragiannis, Kurokawa, Moulin, Procaccia, Shah, Wang, EC'16]
- In every case: define a suitable relaxation, solve it, and round the solution.

Extensions of the 2-approximation algorithm:

- Additive-separable concave valuations [Anari, Mai, Oveis Gharan, Vazirani, 2016]
- Multiple copies of each item [Bei, Garg, Hoefer, Mehlhorn, SAGT'17]
- budget-additive valuations $u_i = \min\left(c_i, \sum_j u_{ij} x_{ij}\right)$, where c_i is a utility cap of agent i. [Garg, Hoefer, Mehlhorn, SODA 2018]

Relaxation – First Attempt

Optimization Problem as (Non-Linear) Integer Program:

$$
\begin{array}{rlcll}
\text{Max.} & \left(\prod_{i \in A} \sum_{j \in G} u_{ij} x_{ij}\right)^{1/n} & & & \\
\text{s.t.} & \sum_{i \in A} x_{ij} & \leq & 1 & j \in G \\
 & x_{ij} & \in & \{0,1\} & i \in A,\ j \in G
\end{array}
$$

Relaxation – First Attempt

Optimization Problem as (Non-Linear) Integer Program:

$$
\begin{array}{rrcll}
\text{Max.} & \displaystyle\frac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G} u_{ij}x_{ij}\right) & & & \\
\text{s.t.} & \displaystyle\sum_{i\in A} x_{ij} & \leq & 1 & j \in G \\
 & x_{ij} & \in & \{0,1\} & i \in A,\ j \in G
\end{array}
$$

Relaxation – First Attempt

Relaxation to Eisenberg-Gale-Type Convex Program:

$$
\begin{aligned}
\text{Max.} \quad & \frac{1}{n} \sum_{i \in A} \log \left(\sum_{j \in G} u_{ij} x_{ij} \right) \\
\text{s.t.} \quad & \sum_{i \in A} x_{ij} \leq 1 \quad j \in G \\
& x_{ij} \geq 0 \quad i \in A,\ j \in G
\end{aligned}
$$

Relaxation – First Attempt

Relaxation to Eisenberg-Gale-Type Convex Program:

$$
\begin{aligned}
\text{Max.} \quad & \frac{1}{n} \sum_{i \in A} \log \left(\sum_{j \in G} u_{ij} x_{ij} \right) \\
\text{s.t.} \quad & \sum_{i \in A} x_{ij} \leq 1 \quad j \in G \\
& x_{ij} \geq 0 \quad i \in A,\ j \in G
\end{aligned}
$$

We already know this program. We even know how to solve it. But, is it a good relaxation?

Relaxation – First Attempt

Relaxation to Eisenberg-Gale-Type Convex Program:

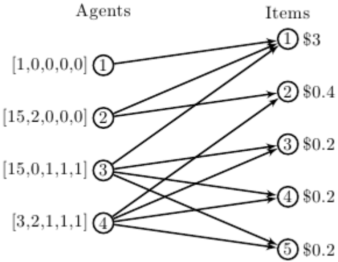
$$\begin{array}{llll} \text{Max.} & \frac{1}{n}\sum_{i\in A}\log\left(\sum_{j\in G} u_{ij}x_{ij}\right) & & \\ \text{s.t.} & \sum_{i\in A} x_{ij} \quad \leq & 1 & j\in G \\ & \qquad x_{ij} \quad \geq & 0 & i\in A,\ j\in G \end{array}$$

We already know this program. We even know how to solve it. But, is it a good relaxation?

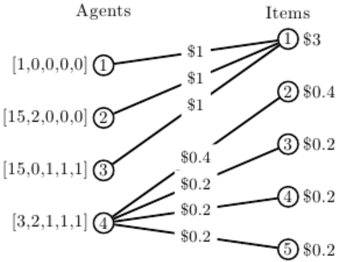
NO, IT IS NOT. Consider: n buyers, $m = n$, $u_{ij} = 1$ for $1 \leq j < n$ and $u_{ij} = 2^n$ for $j = n$ and all i.

Optimal fractional allocation: $u_i = (n - 1 + 2^n)/n$ for all i.
NSW $= \left(((n-1+2^n)/n)^n\right)^{1/n} \geq 2^n/n$.

Optimal integral allocation NSW $= \left(1^{n-1}2^n\right)^{1/n} = 2$.



(a) The MBB graph $G(p)$.



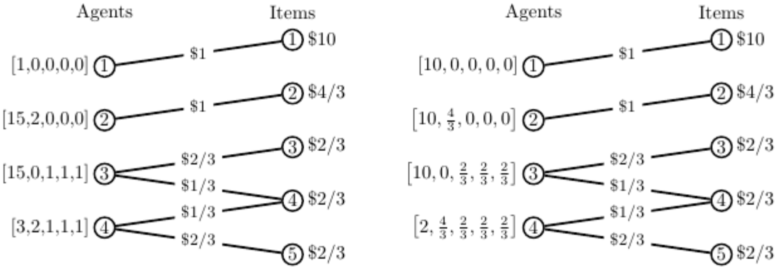
(b) The spending graph $Q(x)$.

For each agent, its utility vector is shown on the left. The entries correspond to the five items on the right, i.e., $u_{11} = 1$ and $u_{21} = 15$. Prices are shown next to each item. The graph on the left is the MBB-graph, i.e., each agent is connected to all goods that give maximum utility per unit of money. Each agent has one unit of money.

The graph on the right shows the money flow in the Fisher market equilibrium. Note that three buyers spend all their money on item 1, and buyer 4 buys goods 2, 3, and 4. This gives very little information on how to round.

Cole - Gkatzelis, Spending-Constrained Equilibrium

Cole-Gkatzelis: Restrict the money inflow to each good to 1, i.e., inflow is 1 if the price is larger than one and inflow is equal to price otherwise.



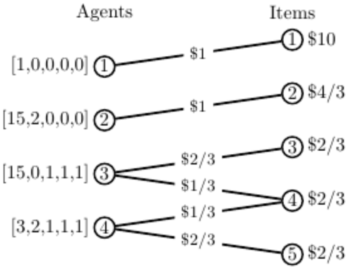
) A spending-restricted equilibrium $(\overline{x}, \overline{p})$. (b) Valuations that are scaled for $\overline{p}$.

The prices and money flow in equilibrium are shown on the left. Note that the price of first good is now much higher so that no buyer other than buyer 1 is interested in the first good.

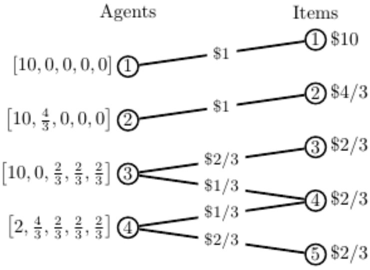
On the right the utility vectors are scaled (recall that NSW is invariant under scaling) such that the utility per unit of money becomes 1, i.e., $u_{ij}/p_j \le 1$ for all edges and $u_{ij}/p_j = 1$ for MBB-edges. This implies that every agent has a utililty of 1 in Market equilibrium.

Cole - Gkatzelis, Spending-Constrained Equilibrium

Cole-Gkatzelis: Restrict the money inflow to each good to 1, i.e., inflow is 1 if the price is larger than one and inflow is equal to price otherwise.



) A spending-restricted equilibrium $(\overline{x}, \overline{p})$.



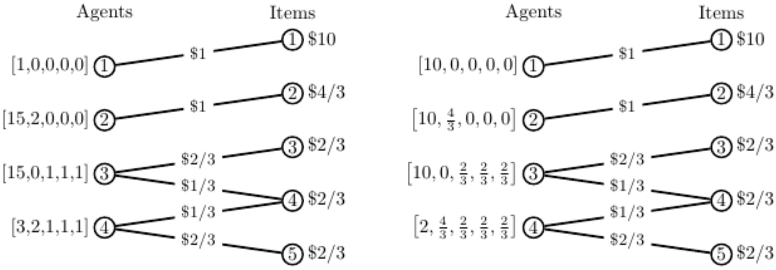
(b) Valuations that are scaled for $\overline{p}$.

The spending-constrained equilibrium suggests to assign good 1 to agent 1, good 2 to agent 2, goods 3 to agent 3, good 5 to agent 5, and good 4 to either agent 3 or 4. The NSW of this assignment is

$$(10 \cdot 4/3 \cdot 4/3 \cdot 2/3)^{1/4} = (320/9)^{1/4} \geq \frac{1}{2.4} \left(\prod_{\{j \mid p_j > 1\}} p_j \right)^{1/4} .$$

Cole - Gkatzelis, Spending-Constrained Equilibrium

Cole-Gkatzelis: Restrict the money inflow to each good to 1, i.e., inflow is 1 if the price is larger than one and inflow is equal to price otherwise.



) A spending-restricted equilibrium $(\overline{x}, \overline{p})$. (b) Valuations that are scaled for $\overline{p}$.

If the allow partial allocation of the goods of price at most one, the NSW becomes

$$(10 \cdot 4/3 \cdot 1 \cdot 1)^{1/4} = (40/3)^{1/4} = \left(\prod_{\{j \mid p_j > 1\}} p_j \right)^{1/4}.$$

Approximation Algorithm for Indivisible Goods (Cole-Gkatzelis)

The spending-constrained equilibrium can be computed in polynomial time.

The equilibrium can be rounded to an integral allocation **x** such that

$$\mathrm{NSW}(\mathbf{x}) \geq \frac{1}{2e^{1/(2e)}} \cdot \prod_{j:p_j>1} p_j \qquad \text{Note } 2e^{1/(2e)} \approx 2.404 \;.$$

The optimal integral allocation $\mathbf{x}^*$ has value at most

$$\mathrm{NSW}(\mathbf{x}^*) \leq \prod_{j:p_j>1} p_j \;.$$

Proposition [Garg, Hoefer, Mehlhorn, SODA'18]

Essentially the same ration can be achieved for budget additive utilities $\min(c_i, \sum_j u_{ij}x_{ij})$. It is NP-hard to approximate NSW with budget-additive valuations to within a factor of $\sqrt{8/7}$ (≈ 1.069).

Approximation Algorithm for Indivisible Goods (Cole-Gkatzelis)

The spending-constrained equilibrium can be computed in polynomial time.

The equilibrium can be rounded to an integral allocation $\mathbf{x}$ such that

$$\mathrm{NSW}(\mathbf{x}) \quad \geq \quad \frac{1}{2e^{1/(2e)}} \cdot \prod_{j:p_j>1} p_j \qquad \text{Note } 2e^{1/(2e)} \approx 2.404 \ .$$

The optimal integral allocation $\mathbf{x}^*$ has value at most

$$\mathrm{NSW}(\mathbf{x}^*) \quad \leq \quad \prod_{j:p_j>1} p_j \ .$$

Proposition [Garg, Hoefer, Mehlhorn, SODA'18]

Essentially the same ration can be achieved for budget additive utilities $\min(c_i, \sum_j u_{ij} x_{ij})$. It is NP-hard to approximate NSW with budget-additive valuations to within a factor of $\sqrt{8/7}$ (≈ 1.069).

Fair Division of Indivisible Goods

Approximation Algorithm for Indivisible Goods (Cole-Gkatzelis)

The spending-constrained equilibrium can be computed in polynomial time.

The equilibrium can be rounded to an integral allocation $\mathbf{x}$ such that

$$\text{NSW}(\mathbf{x}) \quad \geq \quad \frac{1}{2e^{1/(2e)}} \cdot \prod_{j:p_j>1} p_j \qquad \text{Note } 2e^{1/(2e)} \approx 2.404 \ .$$

The optimal integral allocation $\mathbf{x}^*$ has value at most

$$\text{NSW}(\mathbf{x}^*) \quad \leq \quad \prod_{j:p_j>1} p_j \ .$$

Proposition [Garg, Hoefer, Mehlhorn, SODA'18]

Essentially the same ration can be achieved for budget additive utilities $\min(c_i, \sum_j u_{ij}x_{ij})$. It is NP-hard to approximate NSW with budget-additive valuations to within a factor of $\sqrt{8/7}$ (≈ 1.069).

Approximation Algorithm for Indivisible Goods (Cole-Gkatzelis)

The spending-constrained equilibrium can be computed in polynomial time.

The equilibrium can be rounded to an integral allocation $\mathbf{x}$ such that

$$\mathrm{NSW}(\mathbf{x}) \quad \geq \quad \frac{1}{2e^{1/(2e)}} \cdot \prod_{j:p_j>1} p_j \qquad \text{Note } 2e^{1/(2e)} \approx 2.404 \ .$$

The optimal integral allocation $\mathbf{x}^*$ has value at most

$$\mathrm{NSW}(\mathbf{x}^*) \quad \leq \quad \prod_{j:p_j>1} p_j \ .$$

Proposition

[Garg, Hoefer, Mehlhorn, SODA'18]

Essentially the same ration can be achieved for budget additive utilities $\min(c_i, \sum_j u_{ij}x_{ij})$. It is NP-hard to approximate NSW with budget-additive valuations to within a factor of $\sqrt{8/7}$ (≈ 1.069).

Upper Bound: $\mathrm{NSW}(\mathbf{x}^*) \leq \prod_{j:p_j>1} p_j$ **for every integral** $\mathbf{x}^*$**.**

Scale utilities so that $u_{ij}/p_j \leq 1$ always and $u_{ij} = p_j$ for MBB-edges.

$H = \{j \mid p_j > 1\}$ and $L = \{j \mid p_j \leq 1\}$. Allow goods in L to allocated partially. Use $\mathbf{x}'$ to denote optimal solution. If j is assigned to i, i receives utility p_j.

Scale utilities so that $u_{ij}/p_j \le 1$ always and $u_{ij} = p_j$ for MBB-edges.

$H = \{j \mid p_j > 1\}$ and $L = \{j \mid p_j \le 1\}$. Allow goods in L to allocated partially.
Use $\mathbf{x}'$ to denote optimal solution. If j is assigned to i, i receives utility p_j.

Assume first, $H = \emptyset$. Then $\sum_i u_i(\mathbf{x}') = \sum_j p_j = n$ and hence

$$\left(\prod_i u_i(\mathbf{x}')\right)^{1/n} \le \left(1/n \sum_i u_i(\mathbf{x}')\right)^{1/n} \le 1.$$

Upper Bound: $\text{NSW}(\mathbf{x}^*) \le \prod_{j : p_j > 1} p_j$ for every integral $\mathbf{x}^*$.

Scale utilities so that $u_{ij}/p_j \le 1$ always and $u_{ij} = p_j$ for MBB-edges.

$H = \{j \mid p_j > 1\}$ and $L = \{j \mid p_j \le 1\}$. Allow goods in L to allocated partially. Use $\mathbf{x}'$ to denote optimal solution. If j is assigned to i, i receives utility p_j.

At most $|H|$ agents are assigned a good in H. So $n - H$ agents receive only goods in L; call this set A_L. Money flow to goods in H is $|H|$. Hence

$$\sum_{i \in A_L} u_i(\mathbf{x}') \le \sum_{j \in L} p_j \le n - |H| \quad \text{and hence} \quad \exists i_0 \in A_L \text{ s.t. } u_{i_0}(\mathbf{x}') \le 1.$$

Upper Bound: $\text{NSW}(\mathbf{x}^*) \leq \prod_{j:p_j>1} p_j$ for every integral $\mathbf{x}^*$.

Scale utilities so that $u_{ij}/p_j \leq 1$ always and $u_{ij} = p_j$ for MBB-edges.

$H = \{j \mid p_j > 1\}$ and $L = \{j \mid p_j \leq 1\}$. Allow goods in L to allocated partially. Use $\mathbf{x}'$ to denote optimal solution. If j is assigned to i, i receives utility p_j.

At most $|H|$ agents are assigned a good in H. So $n - H$ agents receive only goods in L; call this set A_L. Money flow to goods in H is $|H|$. Hence

$$\sum_{i \in A_L} u_i(\mathbf{x}') \leq \sum_{j \in L} p_j \leq n - |H| \quad \text{and hence} \quad \exists i_0 \in A_L \text{ s.t. } u_{i_0}(\mathbf{x}') \leq 1.$$

Assume $u_i(\mathbf{x}') > 1$ for some i that is partially assigned a good in L. Shift from i to i_0 and improve $\text{NSW}(\mathbf{x}')$, a contradiction. In particular, $u_i(\mathbf{x}') \leq 1$ for all i that are partially assigned some good in L.

Upper Bound: $\mathrm{NSW}(\mathbf{x}^*) \le \prod_{j:p_j>1} p_j$ for every integral $\mathbf{x}^*$.

Scale utilities so that $u_{ij}/p_j \le 1$ always and $u_{ij} = p_j$ for MBB-edges.

$H = \{j \mid p_j > 1\}$ and $L = \{j \mid p_j \le 1\}$. Allow goods in L to allocated partially. Use $\mathbf{x}'$ to denote optimal solution. If j is assigned to i, i receives utility p_j.

At most $|H|$ agents are assigned a good in H. So $n - H$ agents receive only goods in L; call this set A_L. Money flow to goods in H is $|H|$. Hence

$$\sum_{i \in A_L} u_i(\mathbf{x}') \le \sum_{j \in L} p_j \le n - |H| \quad \text{and hence} \quad \exists i_0 \in A_L \text{ s.t. } u_{i_0}(\mathbf{x}') \le 1.$$

Assume $u_i(\mathbf{x}') > 1$ for some i that is partially assigned a good in L. Shift from i to i_0 and improve $\mathrm{NSW}(\mathbf{x}')$, a contradiction. In particular, $u_i(\mathbf{x}') \le 1$ for all i that are partially assigned some good in L.

Assume next that some agent in $A \setminus A_L$ receives two goods from H, say h and j. Reassign one to i_0 and note that that $(\alpha + p_h)p_j \ge \alpha(p_h + p_j)$ for $u_{i_0}(\mathbf{x}') = \alpha \le 1 < p_h, p_j$.

Rounding the Equilibrium

Step 1: Allocation graph forms a forest. For each tree component, assign some agent to be the root.

Step 2: For every good j keep at most one child-agent. If $p_j \leq 1/2$, cut the edges to all children and make them roots. If $p_j > 1/2$, keep the child-agent i that buys the largest amount of j among the child agents.

In other words, an agent i is cut from its parent j, if $p_j \leq 1/2$ or some sibling buys more of j than i.

Step 3: Goods with no child-agent are assigned to their parent-agent.

- No agent suffers under step 3.
- In step 2, agents may loose allocation. If an agent loses allocation, it becomes a root. Hence, only roots lose allocation. A root loses at most half of its value (because a root has spent at most 1/2 on its former parent-good).

Rounding the Equilibrium

Structure of the forest:

- Every good has exactly one child agent.
- Assigning its parent-good to a non-root will give it a value that it at least 1/2 of its value in Market equilibrium (because price of parent-good is at 1/2 and value of every agent in Market equilibrium is one).

Step 4: For each tree component: Define path

$$a_1 - g_1 - a_2 - g_2 - \cdots - a_\ell - g_\ell - a_{\ell+1},$$

where g_i is the child of a_i that contributes most to a_i's utility. Assign g_i to a_i. Outside path assign each good to its child-agent.

Aside the path, good goes to child agent, gives a 2-approximation. On the path, approximation is captured by a telescopic product term.

Analysis of Path: Assume first that all prices are at most one.

Define path

$$a_1 - g_1 - a_2 - g_2 - \cdots - a_\ell - g_\ell - a_{\ell+1},$$

where g_i is the child of a_i that contributes most to a_i's utility. Assign g_i to a_i.

So a_i receives utility p_i in rounded solution. Let q_i be the fraction of g_i that goes to a_i in Market equilibrium.

- For $i \geq 2$, a_i spends at least q_{i-1} on its children in Market equilibrium: At most fraction $1 - q_{i-1}$ of g_{i-1} is allocated to a_i and hence spending on parent is at most this amount.
- a_1 spends at least $1/2$ on its children in Market equilibrium.
- $a_{\ell+1}$ receives at least utility q_ℓ from its children in Market equilibrium and hence in rounded solution.
- Let k_i be the number of children-goods of a_i.
- $q_1 p_1 =$ money spend by a_1 on $g_1 \geq 1/(2k_1)$
- $q_i p_i =$ money spent by a_i to $g_i \geq q_{i-1}/k_i$ for $1 \leq i \leq \ell$.

$$\prod_{1 \leq i \leq \ell+1} \text{value of } a_i \text{ in rounded solution} \geq p_1 \cdots p_\ell \cdot q_\ell \geq \frac{1}{2q_1 k_1} \frac{q_1}{q_2 k_2} \cdot \frac{q_{\ell-1}}{q_\ell k_\ell} q_\ell$$

$$= \frac{1}{2k_1 \cdots k_\ell}$$

Summary

- Market equilibriua for linear Arrow-Debrau and Fisher Markets
- Nash Social Welfare captures fair division.
- Nash Social Welfare for Divisible Goods and Fisher Markets
- Approximation Algs for Nash Social Welfare for Indivisible Goods via Spending-Restricted Markets