Approximability of Pricing Problems

Piotr Krysta

joint work with Patrick Briest

Department of Computer Science
University of Liverpool
U.K.

28-03-2007

Multi-Product Pricing Problems

- Websites comparing available products help customers make optimal buying decisions.
- Customers reveal their preferences and budgets.

Multi-Product Pricing Problems

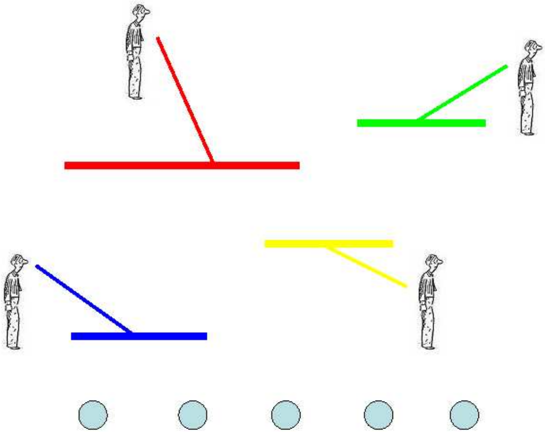
- Websites comparing available products help customers make optimal buying decisions.
- Customers reveal their preferences and budgets.

Goal: use available data to compute profit maximmizing prices for a company's product range.

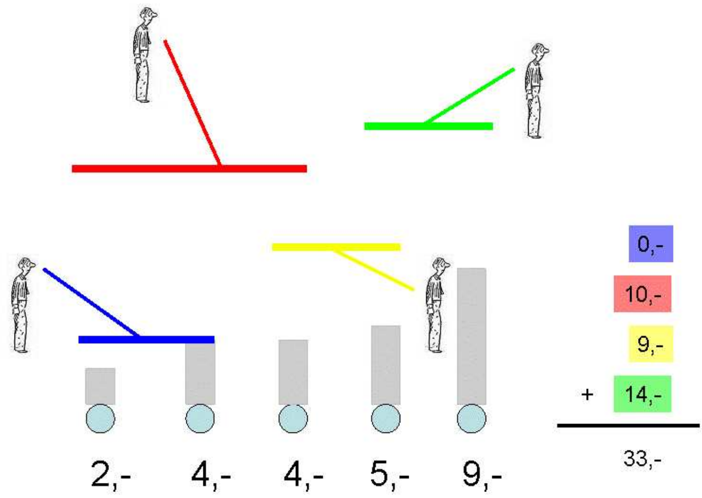
Different approaches taken to model markets. Here:

- Single-Minded Unlimited-Supply Pricing:
  - *single-minded* customers, each interested in a single set of products,
  - *unlimited supply*, i.e., no production constraints.
  - Customer buys if the sum of prices is below her budget.
- Unit-Demand Pricing:
  - *unit-demand* customers, each buy a single product in a set of products,
  - *unlimited or limited supply*,
  - Customer buys only products with prices below their budgets.

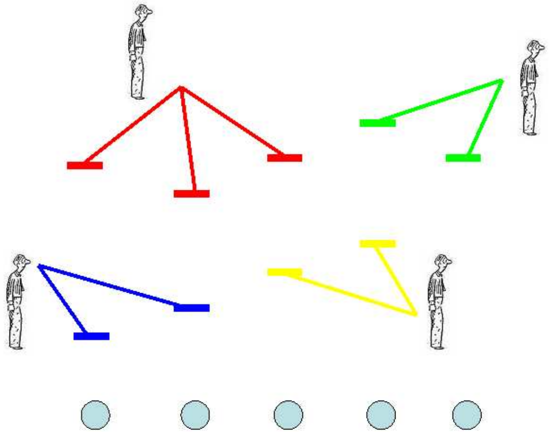
Example $\longrightarrow$ Single-Minded Pricing:



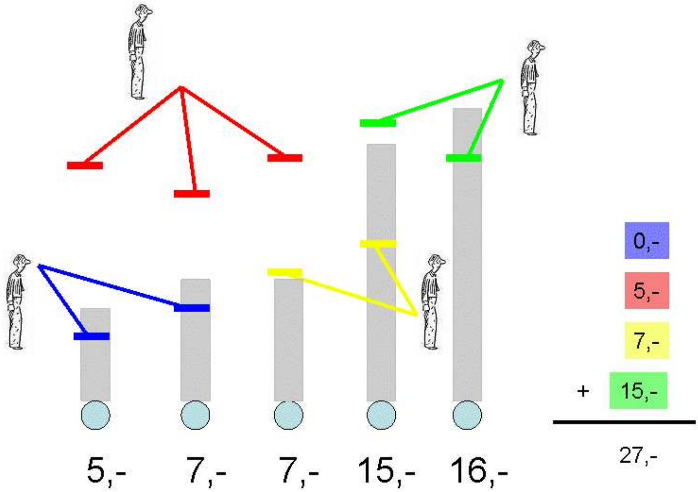
Example $\longrightarrow$ Single-Minded Pricing:



Example $\longrightarrow$ Unit-Demand Pricing:



Example $\longrightarrow$ Unit-Demand Pricing:



Single-Minded Unlimited-Supply Pricing [*Briest, Krysta, SODA '06*]

Single-Minded Unlimited-Supply Pricing *(SUSP)*

Given products $\mathcal{U}$ and sets $\mathcal{S}$ with values $v(S)$ find prices p, such that

$$\sum_{S:\sum_{e\in S} p(e)\leq v(S)} \sum_{e\in S} p(e) \longrightarrow \max.$$

$\rightsquigarrow$ models pricing of direct connections in computer or transportation networks.

Pricing in Graphs*(G-SUSP)*

Given graph $G = (V, E)$ and paths $\mathcal{P}$, assign profit-maximizing prices p to edges.

First investigated by *Guruswami et al. (2005)*. Recent inapproximability result due to *Demaine et al. (2006)*.

In general:

- $O(\log|\mathcal{U}| + \log|\mathcal{S}|)$-approximation
- inapproximable within $O(\log^{\delta}|\mathcal{U}|)$ for some $0 < \delta < 1$

With G being a line *(Highway Problem)*:

- poly-time algo for integral valuations of constant size
- pseudopolynomial time algo for paths of constant length

Q: Is there a poly-time algorithm for the Highway Problem?

First investigated by *Guruswami et al. (2005)*. Recent inapproximability result due to *Demaine et al. (2006)*.

In general:

- $O(\log|\mathcal{U}| + \log|\mathcal{S}|)$-approximation
- inapproximable within $O(\log^{\delta}|\mathcal{U}|)$ for some $0 < \delta < 1$

With G being a line *(Highway Problem)*:

- poly-time algo for integral valuations of constant size
- pseudopolynomial time algo for paths of constant length

Q: Is there a poly-time algorithm for the Highway Problem? No!

Hardness Results

The Highway Problem

Theorem

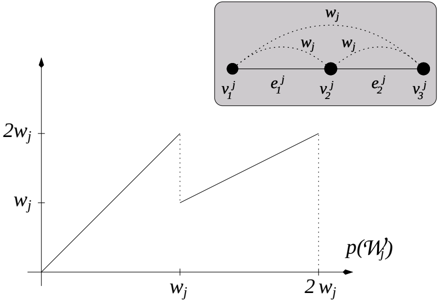
The Highway Problem is NP-hard.

Sketch of Proof: PARTITION problem:

Given positive weights $w_1, \ldots, w_n$, does there exist $S \subset \{1, \ldots, n\}$, such that

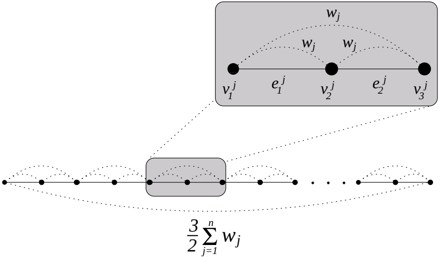
$$\sum_{j \in S} w_j = \sum_{j \notin S} w_j \ ?$$

Design gadgets that capture the discrete nature of this problem.



Weight Gadgets

Maximum profit out of $\mathcal{W}_j$ is $2w_j$.
It is obtained iff $p(\mathcal{W}_j) = p(e_1^j) + p(e_2^j)$ is set to w_j or $2w_j$.



Maximum profit $\frac{7}{2}\sum_{j=1}^{n} w_j$ is obtained iff there exists $S \subset \{1,\ldots,n\}$ with $\sum_{j\in S} w_j = \sum_{j\notin S} w_j$. $\square$

G-SUSP

Inapproximability of Sparse Problem Instances

APX-hardness of G-SUSP due to *Guruswami et al. (2005)*.

Applications in realistic network settings often lead to sparse problem instances. Hardness of approximation still holds if:

- G has constant degree d
- paths have constant lengths $\leq \ell$
- at most a constant number B of paths per edge
- only constant height valuations

Theorem

G-SUSP on sparse instances is APX-hard.

Approximation Algorithms

Best ratio in the general case: $\log |\mathcal{U}| + \log |\mathcal{S}|$
Guruswami, Hartline, Karlin, Kempe, Kenyon, McSherry (2005)

Not approximable within $\log^{\delta} |\mathcal{U}|$ for some $0 < \delta < 1$.
Demaine, Feige, Hajiaghayi, Salavatipour (2006)

Can we do better on sparse problem instances, i.e., can we obtain approximation ratios depending on

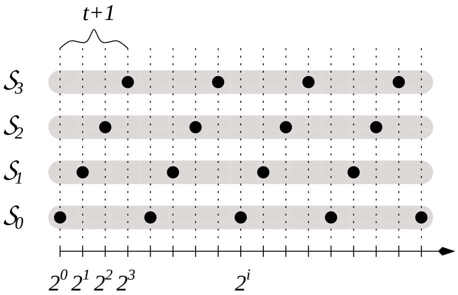
- ℓ, the maximum cardinality of any set $S \in \mathcal{S}$
- B, the maximum number of sets containing some product $e \in \mathcal{U}$

rather than $|\mathcal{U}|$ and $|\mathcal{S}|$?

An $O(\log \ell + \log B)$-Approximation

Let $\delta(S) = v(S)/|S|$ be *price per product* of set S.

1. Round all $\delta(S)$ to powers of 2. Let $\mathcal{S} = \mathcal{S}_0 \cup \ldots \cup \mathcal{S}_t$ where $t = \lceil \log \ell^2 B \rceil - 1$. In $\mathcal{S}_i$: $\delta(S) > \delta(T) \Rightarrow \delta(S)/\delta(T) \geq \ell^2 B$.



2. In each $\mathcal{S}_i$ select non-intersecting sets with maximum δ-value and compute optimal prices.

Analysis:

- $Opt(\mathcal{S}) \leq \sum_{i=1}^{t} Opt(\mathcal{S}_i)$
- Let $S \in \mathcal{S}_i$, $\mathcal{I}(S)$ intersecting sets with smaller δ-values:

$$v(S) \geq \sum_{T \in \mathcal{I}(S)} v(T)$$

- Let $\mathcal{S}_i^*$ be non-intersecting sets with max. δ as in the algo. Then

$$Opt(\mathcal{S}_i) \leq 2 \cdot Opt(\mathcal{S}_i^*),$$

and, since we compute $\max_i Opt(\mathcal{S}_i^*)$:

Theorem

The above algorithm has approximation ratio $O(\log \ell + \log B)$.

Upper bounding technique

We relate $Opt(\mathcal{S}_i^*)$ to $Opt(\mathcal{S}_i)$ by using as an upper bound

$$Opt(\mathcal{S}_i) \leq \sum_{S \in \mathcal{S}_i} v(S),$$

i.e., the sum of all valuations.

Using this upper bounding technique, no approximation ratio $o(\log B)$ can be achieved.

In many applications: $B >> \ell$.

Can we obtain ratios independent of B?

An $O(\ell^2)$-Approximation

Define *(smoothed)* s-SUSP by changing the objective to

$$\sum_{S \in \Lambda(p)} \sum_{e \in S} p(e),$$

where $\Lambda(p) = \{S \in \mathcal{S} \mid p(e) \leq \delta(S) \, \forall \, e \in S\}$.

We derive an $O(\ell)$-approximation for s-SUSP.

1. For every $e \in \mathcal{U}$ compute the optimal price $p^*(e)$ assuming all other prices were 0.
2. Resolve existing *conflicts*.

Set S is *conflicting*, if

$$\exists \, e, f \in S \; : \; p^*(e) \leq \delta(S) < p^*(f).$$

Upper bounding technique

1. For every $e \in \mathcal{U}$ compute the optimal price $p^*(e)$ assuming all other prices were 0.
2. Our upper bound: $Opt \leq \sum_{e \in \mathcal{U}} p^*(e)$

Summary (SUSP):

- Hardness results
  - NP-hardness of the Highway Problem
  - APX-hardness of G-SUSP for sparse instances
- Approximation Algorithms
  - $O(\log \ell + \log B)$-approximation ($\rightsquigarrow$ partitioning)
  - $O(\ell^2)$-approximation ($\rightsquigarrow$ conflict graph)

Unit-Demand Pricing [*Briest, Krysta, SODA '07*]

Unit-Demand Pricing *(UDP)*

Given products $\mathcal{U}$ and consumer samples $\mathcal{C}$ consisting of budgets $b(c, e) \in \mathbb{R}_0^+ \; \forall c \in \mathcal{C}, e \in \mathcal{U}$.

Unit-Demand Pricing *(UDP)*

Given products $\mathcal{U}$ and consumer samples $\mathcal{C}$ consisting of budgets $b(c,e) \in \mathbb{R}_0^+ \; \forall c \in \mathcal{C}, e \in \mathcal{U}$.

For prices $p : \mathcal{U} \to \mathbb{R}_0^+$:
$\mathcal{A}(p) = \{c \in \mathcal{C} \,|\, \exists e \in \mathcal{U} \; : \; p(e) \leq b(c,e)\}$ = consumers affording to buy any product.

Unit-Demand Pricing *(UDP)*

Given products $\mathcal{U}$ and consumer samples $\mathcal{C}$ consisting of budgets $b(c,e) \in \mathbb{R}_0^+ \; \forall c \in \mathcal{C}, e \in \mathcal{U}$.

For prices $p : \mathcal{U} \to \mathbb{R}_0^+$:
$\mathcal{A}(p) = \{c \in \mathcal{C} \,|\, \exists e \in \mathcal{U} \,:\, p(e) \leq b(c,e)\}$ = consumers affording to buy any product.

In no price ladder scenario (NPL) we find prices p that maximize:

- $\sum_{c \in \mathcal{A}(p)} \min\{p(e) \,|\, p(e) \leq b(c,e)\}$ (Udp-Min-Npl)

Unit-Demand Pricing *(UDP)*

Given products $\mathcal{U}$ and consumer samples $\mathcal{C}$ consisting of budgets $b(c,e) \in \mathbb{R}_0^+ \; \forall c \in \mathcal{C}, e \in \mathcal{U}$.

For prices $p : \mathcal{U} \rightarrow \mathbb{R}_0^+$:
$\mathcal{A}(p) = \{c \in \mathcal{C} \,|\, \exists e \in \mathcal{U} \,:\, p(e) \leq b(c,e)\}$ = consumers affording to buy any product.

In no price ladder scenario (NPL) we find prices p that maximize:

- $\sum_{c \in \mathcal{A}(p)} \min\{p(e) \,|\, p(e) \leq b(c,e)\}$ (Udp-Min-Npl)
- $\sum_{c \in \mathcal{A}(p)} \max\{p(e) \,|\, p(e) \leq b(c,e)\}$ (Udp-Max-Npl)

Unit-Demand Pricing *(UDP)*

Given products $\mathcal{U}$ and consumer samples $\mathcal{C}$ consisting of budgets $b(c,e) \in \mathbb{R}_0^+ \; \forall c \in \mathcal{C}, e \in \mathcal{U}$.

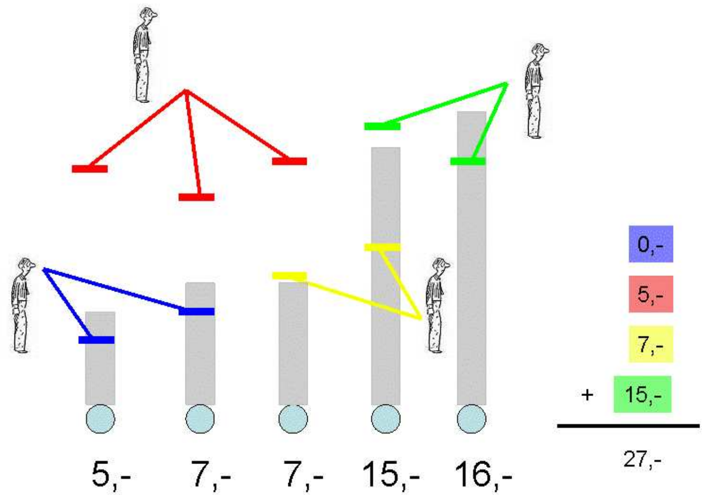
For prices $p : \mathcal{U} \to \mathbb{R}_0^+$:
$\mathcal{A}(p) = \{c \in \mathcal{C} \mid \exists e \in \mathcal{U} \, : \, p(e) \leq b(c,e)\}$ = consumers affording to buy any product.

In no price ladder scenario (NPL) we find prices p that maximize:

- $\sum_{c \in \mathcal{A}(p)} \min\{p(e) \mid p(e) \leq b(c,e)\}$ (UDP-MIN-NPL)
- $\sum_{c \in \mathcal{A}(p)} \max\{p(e) \mid p(e) \leq b(c,e)\}$ (UDP-MAX-NPL)

Given a price ladder constraint (PL), $p(e_1) \leq \cdots \leq p(e_{|\mathcal{U}|})$, UDP-{MIN,MAX}-PL asks for prices p satisfying PL.

Example $\longrightarrow$ UDP-MIN-PL:



UDP introduced as non-parametric multi-product pricing in [1].

[1] *Glynn, Rusmevichientong and Van Roy (2003)*

[2] *Aggarwal, Feder, Motwani and Zhu (2004)*

UDP-MIN-{PL,NPL}:

- UDP-MIN-PL poly-time for uniform budgets consumers [1].
- UDP-MIN-NPL APX-hard, has $\mathcal{O}(\log|\mathcal{C}|)$-approx [2].

UDP-MAX-{PL,NPL}:

- UDP-MAX-PL has a PTAS [2].
- UDP-MAX-PL, limited supply: 4-approx [2].
- UDP-MAX-NPL 16/15-hard, has 1.59-approx [2].

Q: UDP-MIN-NPL: Is there a const approx ?

UDP introduced as non-parametric multi-product pricing in [1].

[1] *Glynn, Rusmevichientong and Van Roy (2003)*

[2] *Aggarwal, Feder, Motwani and Zhu (2004)*

UDP-MIN-{PL,NPL}:

- UDP-MIN-PL poly-time for uniform budgets consumers [1].
- UDP-MIN-NPL APX-hard, has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2].

UDP-MAX-{PL,NPL}:

- UDP-MAX-PL has a PTAS [2].
- UDP-MAX-PL, limited supply: 4-approx [2].
- UDP-MAX-NPL 16/15-hard, has 1.59-approx [2].

Q: UDP-MIN-NPL: Is there a const approx ? No!

UDP introduced as non-parametric multi-product pricing in [1].

[1] *Glynn, Rusmevichientong and Van Roy (2003)*

[2] *Aggarwal, Feder, Motwani and Zhu (2004)*

UDP-MIN-{PL,NPL}:

- UDP-MIN-PL poly-time for uniform budgets consumers [1].
- UDP-MIN-NPL APX-hard, has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2].

UDP-MAX-{PL,NPL}:

- UDP-MAX-PL has a PTAS [2].
- UDP-MAX-PL, limited supply: 4-approx [2].
- UDP-MAX-NPL 16/15-hard, has 1.59-approx [2].

Q: UDP-MIN-NPL: Is there a const approx ? No!
UDP-MAX-PL: Is PTAS best possible approx ?

UDP introduced as non-parametric multi-product pricing in [1].

[1] *Glynn, Rusmevichientong and Van Roy (2003)*

[2] *Aggarwal, Feder, Motwani and Zhu (2004)*

Udp-Min-{Pl,Npl}:

- Udp-Min-Pl poly-time for uniform budgets consumers [1].
- Udp-Min-Npl APX-hard, has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2].

Udp-Max-{Pl,Npl}:

- Udp-Max-Pl has a PTAS [2].
- Udp-Max-Pl, limited supply: 4-approx [2].
- Udp-Max-Npl 16/15-hard, has 1.59-approx [2].

Q: Udp-Min-Npl: Is there a const approx ? No!
Udp-Max-Pl: Is PTAS best possible approx ? Yes!

UDP introduced as non-parametric multi-product pricing in [1].

[1] *Glynn, Rusmevichientong and Van Roy (2003)*

[2] *Aggarwal, Feder, Motwani and Zhu (2004)*

UDP-MIN-{PL,NPL}:

- UDP-MIN-PL poly-time for uniform budgets consumers [1].
- UDP-MIN-NPL APX-hard, has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2].

UDP-MAX-{PL,NPL}:

- UDP-MAX-PL has a PTAS [2].
- UDP-MAX-PL, limited supply: 4-approx [2].
- UDP-MAX-NPL 16/15-hard, has 1.59-approx [2].

Q: UDP-MIN-NPL: Is there a const approx ? No!
UDP-MAX-PL: Is PTAS best possible approx ? Yes!
UDP-MAX-NPL, limit'd supply: const-approx, APX-hard ?

UDP introduced as non-parametric multi-product pricing in [1].

[1] *Glynn, Rusmevichientong and Van Roy (2003)*

[2] *Aggarwal, Feder, Motwani and Zhu (2004)*

UDP-MIN-{PL,NPL}:

- UDP-MIN-PL poly-time for uniform budgets consumers [1].
- UDP-MIN-NPL APX-hard, has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2].

UDP-MAX-{PL,NPL}:

- UDP-MAX-PL has a PTAS [2].
- UDP-MAX-PL, limited supply: 4-approx [2].
- UDP-MAX-NPL 16/15-hard, has 1.59-approx [2].

Q: UDP-MIN-NPL: Is there a const approx ? No!
UDP-MAX-PL: Is PTAS best possible approx ? Yes!
UDP-MAX-NPL, limit'd supply: const-approx, APX-hard ? Yes!

Hardness Results

UDP-MIN-NPL: Is there a const approx ? No!

Recall: UDP-MIN-NPL has $\mathcal{O}(\log|\mathcal{C}|)$-approx [2]. We prove:

Theorem

UDP-MIN-{PL,NPL} is not approximable within $\mathcal{O}(\log^{\varepsilon}|\mathcal{C}|)$ for some $\varepsilon > 0$, unless NP $\subseteq$ DTIME$(n^{\mathcal{O}(\log\log n)})$.

Recall: $\textsc{Udp-Min-Npl}$ has $\mathcal{O}(\log|\mathcal{C}|)$-approx [2]. We prove:

Theorem

$\textsc{Udp-Min-}\{\textsc{Pl},\textsc{Npl}\}$ is not approximable within $\mathcal{O}(\log^{\varepsilon}|\mathcal{C}|)$ for some $\varepsilon > 0$, unless NP $\subseteq$ DTIME$(n^{\mathcal{O}(\log\log n)})$.

Sketch of Proof:

Let $\alpha(G)$ = size of the maximum independent set in graph G.

Recall: UDP-MIN-NPL has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2]. We prove:

Theorem

UDP-MIN-{PL,NPL} is not approximable within $\mathcal{O}(\log^{\varepsilon} |\mathcal{C}|)$ for some $\varepsilon > 0$, unless NP $\subseteq$ DTIME$(n^{\mathcal{O}(\log \log n)})$.

Sketch of Proof:

Let $\alpha(G)$ = size of the maximum independent set in graph G.

Proposition [Alon, Feige, Wigderson, Zuckerman'95]

$n \in \mathbb{N}$, $\mathcal{G} = \{G : G = (V, E)$ with max degree $\mathcal{O}(\log n), |V| = n\}$.
There is $\varepsilon > 0$, s.t. $\mathcal{O}(\log^{\varepsilon} n)$-approx to $\alpha(G)$ is NP-hard for $G \in \mathcal{G}$.

Recall: $\textsc{Udp-Min-Npl}$ has $\mathcal{O}(\log |\mathcal{C}|)$-approx [2]. We prove:

Theorem

$\textsc{Udp-Min-}\{\textsc{Pl},\textsc{Npl}\}$ is not approximable within $\mathcal{O}(\log^{\varepsilon} |\mathcal{C}|)$ for some $\varepsilon > 0$, unless $\mathrm{NP} \subseteq \mathrm{DTIME}(n^{\mathcal{O}(\log\log n)})$.

Sketch of Proof:

Let $\alpha(G)$ = size of the maximum independent set in graph G.

Proposition [Alon, Feige, Wigderson, Zuckerman'95]

$n \in \mathbb{N}$, $\mathcal{G} = \{G : G = (V, E)$ with max degree $\mathcal{O}(\log n), |V| = n\}$.
There is $\varepsilon > 0$, s.t. $\mathcal{O}(\log^{\varepsilon} n)$-approx to $\alpha(G)$ is NP-hard for $G \in \mathcal{G}$.

Given $G \in \mathcal{G}$, we reduce finding $\alpha(G)$ to $\textsc{Udp-Min-Pl}$.

Recall: $\textsc{Udp-Min-Npl}$ has $\mathcal{O}(\log|\mathcal{C}|)$-approx [2]. We prove:

Theorem

$\textsc{Udp-Min-}\{\textsc{Pl},\textsc{Npl}\}$ is not approximable within $\mathcal{O}(\log^{\varepsilon}|\mathcal{C}|)$ for some $\varepsilon > 0$, unless NP $\subseteq$ DTIME$(n^{\mathcal{O}(\log\log n)})$.

Sketch of Proof:

Let $\alpha(G)$ = size of the maximum independent set in graph G.

Proposition [Alon, Feige, Wigderson, Zuckerman'95]

$n \in \mathbb{N}$, $\mathcal{G} = \{G : G = (V,E)$ with max degree $\mathcal{O}(\log n), |V| = n\}$.
There is $\varepsilon > 0$, s.t. $\mathcal{O}(\log^{\varepsilon} n)$-approx to $\alpha(G)$ is NP-hard for $G \in \mathcal{G}$.

Given $G \in \mathcal{G}$, we reduce finding $\alpha(G)$ to $\textsc{Udp-Min-Pl}$.

Assume *a.c.*: $\textsc{Udp-Min-Pl}$ has $\mathcal{O}(\log^{\varepsilon-\delta}|\mathcal{C}|)$-approx for some $\delta > 0$.

Recall: Udp-Min-Npl has $\mathcal{O}(\log|\mathcal{C}|)$-approx [2]. We prove:

Theorem

Udp-Min-{Pl,Npl} is not approximable within $\mathcal{O}(\log^{\varepsilon}|\mathcal{C}|)$ for some $\varepsilon > 0$, unless NP $\subseteq$ DTIME$(n^{\mathcal{O}(\log\log n)})$.

Sketch of Proof:

Let $\alpha(G)$ = size of the maximum independent set in graph G.

Proposition [Alon, Feige, Wigderson, Zuckerman'95]

$n \in \mathbb{N}$, $\mathcal{G} = \{G : G = (V, E)$ with max degree $\mathcal{O}(\log n), |V| = n\}$. There is $\varepsilon > 0$, s.t. $\mathcal{O}(\log^{\varepsilon} n)$-approx to $\alpha(G)$ is NP-hard for $G \in \mathcal{G}$.

Given $G \in \mathcal{G}$, we reduce finding $\alpha(G)$ to Udp-Min-Pl.

Assume *a.c.*: Udp-Min-Pl has $\mathcal{O}(\log^{\varepsilon-\delta}|\mathcal{C}|)$-approx for some $\delta > 0$.

We will show that this gives $\mathcal{O}(\log^{\varepsilon} n)$-approx for $\alpha(G)$ in time $n^{\mathcal{O}(\log\log n)}$.

Sketch of Proof:

Independent Set Problem (IS)

Given undirected graph $G = (V, E)$, $|V| = n$, $|E| = m$, find maximum cardinality subset $V' \subseteq V$ with $\{v, w\} \notin E$ for any $v, w \in V'$.

Sketch of Proof:

Independent Set Problem (IS)

Given undirected graph $G = (V, E)$, $|V| = n$, $|E| = m$, find maximum cardinality subset $V' \subseteq V$ with $\{v, w\} \notin E$ for any $v, w \in V'$.

Many hardness results, among others hard to approximate within

- $\mathcal{O}(n^{\varepsilon})$
- $\mathcal{O}(\Delta^{\varepsilon})$ in graphs of maximum degree Δ

for some $\varepsilon > 0$, unless P=NP.

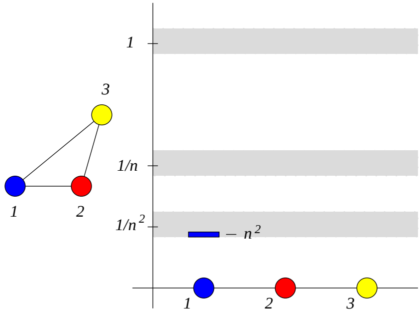
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



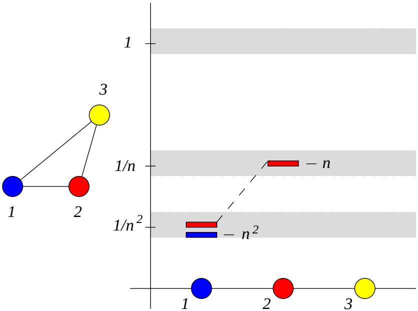
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \dots, v_n\}$

$\mathcal{P} = \{e_1, \dots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



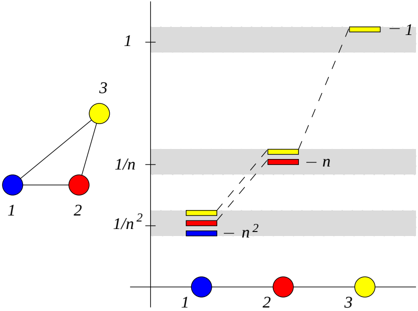
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



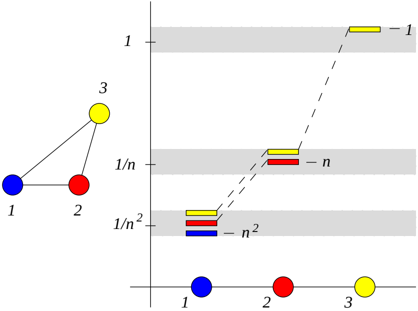
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



$rev(\mathcal{C}_j) = 1 \Rightarrow rev(\mathcal{C}_i) \leq 1/n$ for all i with $v_i \in \mathcal{V}_j$ or $v_j \in \mathcal{V}_i$

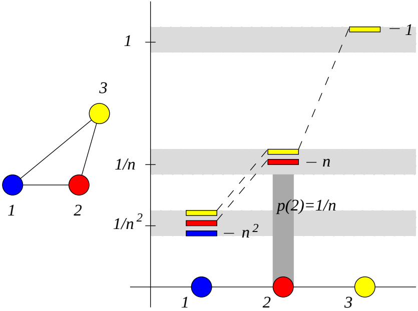
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



$rev(\mathcal{C}_j) = 1 \Rightarrow rev(\mathcal{C}_i) \leq 1/n$ for all i with $v_i \in \mathcal{V}_j$ or $v_j \in \mathcal{V}_i$

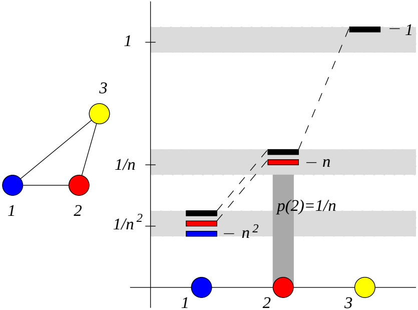
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



$rev(\mathcal{C}_j) = 1 \Rightarrow rev(\mathcal{C}_i) \leq 1/n$ for all i with $v_i \in \mathcal{V}_j$ or $v_j \in \mathcal{V}_i$

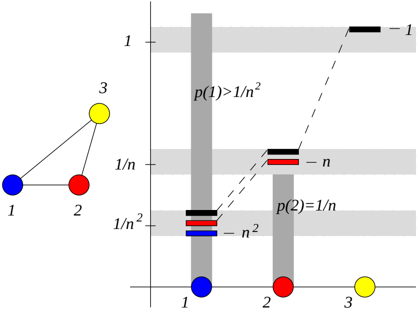
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



$rev(\mathcal{C}_j) = 1 \Rightarrow rev(\mathcal{C}_i) \leq 1/n$ for all i with $v_i \in \mathcal{V}_j$ or $v_j \in \mathcal{V}_i$

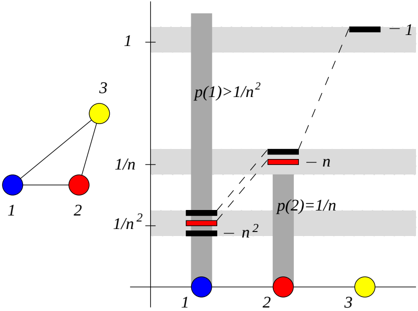
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



$rev(\mathcal{C}_j) = 1 \Rightarrow rev(\mathcal{C}_i) \leq 1/n$ for all i with $v_i \in \mathcal{V}_j$ or $v_j \in \mathcal{V}_i$

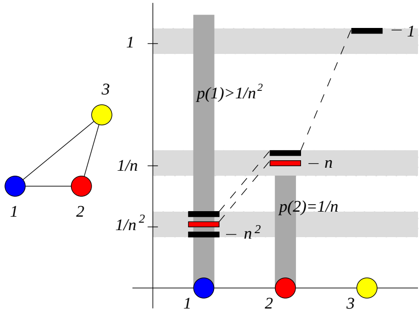
Sketch of Proof:

Independence via geometrically increasing threshold prices:

$G = (V, E)$, $V = \{v_1, \ldots, v_n\}$

$\mathcal{P} = \{e_1, \ldots, e_n\}$, $\mu_j = 1/n^{n-j}$
$\mathcal{V}_j = \{v_j\} \cup \{v_i | \{v_i, v_j\} \in E, i < j\}$

$v_j \rightsquigarrow n^{n-j}$ consumers $\mathcal{C}_j$ with budgets μ_i for all $i \in \mathcal{V}_j$, 0 else



$rev(\mathcal{C}_j) = 1 \Rightarrow rev(\mathcal{C}_i) \leq 1/n$ for all i with $v_i \in \mathcal{V}_j$ or $v_j \in \mathcal{V}_i$

Gadgets allow to encode independence. But graphs of size n result in instances of size $\Omega(n^n)$.

Sketch of Proof:

Way out: Trade some hardness for sparser problem instances.

Theorem

IS in graphs on n vertices with maximum degree $\Delta(n) = \mathcal{O}(\log n)$ is not approximable within $\mathcal{O}(\log^{\varepsilon} n)$ for some $\varepsilon > 0$, unless P=NP.

Sketch of Proof:

Way out: Trade some hardness for sparser problem instances.

Theorem

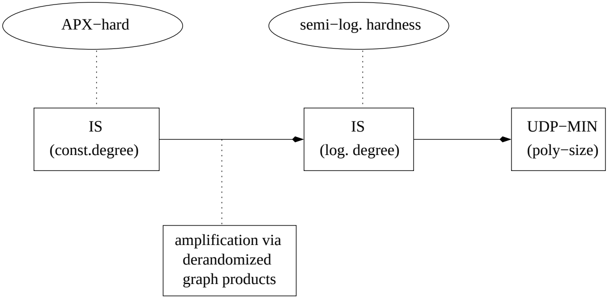
IS in graphs on n vertices with maximum degree $\Delta(n) = \mathcal{O}(\log n)$ is not approximable within $\mathcal{O}(\log^{\varepsilon} n)$ for some $\varepsilon > 0$, unless P=NP.

These graphs are just what we need, because...

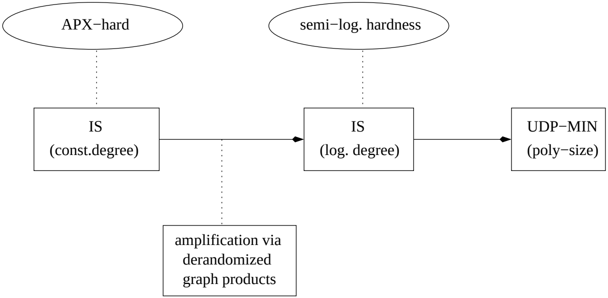
Graphs of maximum degree Δ are $(\Delta + 1)$-colorable.

...and vertices of one color can be realized on one price level.

Sketch of Proof:



Sketch of Proof:



Theorem

The UDP-MIN-{PL,NPL} problem is not approximable within $\mathcal{O}(\log^{\varepsilon}|\mathcal{C}|)$ for some $\varepsilon > 0$, unless NP $\subseteq$ DTIME$(n^{\mathcal{O}(\log\log n)})$.

Approximation Algorithms

UDP-MAX-NPL, limit'd supply: const-approx, APX-hard ? Yes!

Theorem

$\textsc{Udp-Max-Npl}$ with supply 1 in P, with limited supply ≤ 2 is APX-hard.

Theorem

There is a 2-approximation algorithm for $\textsc{Udp-Max-Npl}$ with limited supply.

Sketch of Proof:
The following is a 2-approximation algorithm:

LocalSearch: Initialize p arbitrarily and compute opt allocation under p. While there exists product e and price $p' \neq p(e)$ s.t. new opt allocation is better, set $p(e) = p'$.

Summary (UDP):

- UDP-MIN-{PL,NPL} is intractable (no const approx), even with PL
- UDP-MAX-{PL,NPL} is tractable (const approx), even with NPL and limited supply