Connection Between Discrete Convex Analysis and Auction Theory

Akiyoshi Shioura
Tokyo Institute of Technology

2023-March-16
International Workshop on
Discrete Convex Analysis and Economics

Auctions with Multiple Indivisible Items

- bidders report their valuation for item sets

- auctioneer determines item price & allocation

  in which all bidders maximize their payoff

  ← Walrasian equilibrium

Question

- When equilibrium exist? Sufficient condition?
- How to compute equilibrium?

Discrete Convex Analysis (Murota 1998)

- Theory of discrete convex functions on $\mathbb{Z}^n$
- Two discrete convexity concepts
  - L♮-convex / M♮-convex
- Various properties
  - Conjugacy between L♮-convexity / M♮-convexity
  - Algorithms for L♮-convex/M♮-convex optimization

Murota (2003) Discrete Convex Analysis, SIAM
Murota (2016) J. Mechanism & Institution Design
"Discrete convex analysis: A tool for economics and game theory"

Connection of Auction & DCA

Discrete Convexity/Concavity in Multi-Item Auctions

- Gross(Strong)-substitutes valuation = M♮-concave fn
- Indirect Utility = L♮-convex fn

Equilibrium Computation and Discrete Convex Optimization

- equilibrium allocation
  = constrained M♮-concave maximization
- equilibrium price = L♮-convex minimization

Discrete Convexity/Concavity in Multi-Item Auctions

- Gross(Strong)-substitutes valuation = M♮-concave fn
- Indirect Utility = L♮-convex fn

Auction Setting: Items & Bidders

items $N = \{1, 2, \ldots, n\}$, $u(j)$ units available for item $j \in N$

$x \in [0, u]_{\mathbb{Z}}$ --- (multi-)set of items

x= (2, 0, 1, 0, 3)

bidders $M = \{1,2, \ldots, m\}$

valuation fn $f_i: [0, u]_{\mathbb{Z}} \to \mathbb{Z}$ of bidder $i \in M$

$f_i(x)$ = "value" of item set $x \in [0, u]_{\mathbb{Z}}$

Walrasian Equilibrium

given price $p = (p_1, p_2, \dots, p_n)$,

bidder wants to maximize payoff $f_i(x) - p^{\mathrm{T}} x$

demand set $D_i(p) = \arg\max\{f_i(x) - p^T x \mid x \in [0, u]_{\mathbb{Z}}\}$

allocation $(x_1, x_2, \dots, x_m)$: $x_i \in \mathbb{Z}_+^n$, $x_1 + x_2 + \cdots + x_m = u$

Def: Walrasian equilibrium:
pair of allocation $(x_1^*, x_2^*, \dots, x_m^*)$ & price p^*
s.t. $x_i^* \in D_i(p^*)$

Models of Multi-Item Auctions

- # of items demanded by each bidder:
  only one (single-demand) / more than one (multi-demand)
- # of units available for each item:
  only one (single-unit) / more than one (multi-unit)

multi-demand/ multi-unit
$f: [0, u]_{\mathbb{Z}} \to \mathbb{Z}$

multi-demand/ single-unit
$f: \{0,1\}^n \to \mathbb{Z}$

single-demand/ single-unit
$f: N \to \mathbb{Z}$

assignment model

Walrasian Equilibrium

given price $p = (p_1, p_2, \dots, p_n)$,

bidder wants to maximize payoff $f_i(x) - p^{\mathrm{T}} x$

demand set $D_i(p) = \arg\max\{f_i(x) - p^T x \mid x \in [0, u]_{\mathbb{Z}}\}$

allocation $(x_1, x_2, \dots, x_m)$: $x_i \in \mathbb{Z}_+^n$, $x_1 + x_2 + \cdots + x_m = u$

Def: Walrasian equilibrium:
pair of allocation $(x_1^*, x_2^*, \dots, x_m^*)$ & price p^*
s.t. $x_i^* \in D_i(p^*)$

※Not always: ∃ Walrasian equilibrium

Question
- When equilibrium exist? Sufficient condition?
- How to compute equilibrium?

Condition for Equilibrium Existence in Multi-Demand/Single-Unit Model

Thm f: valuations on $\{0,1\}^n$ [Kelso-Crawford1982, et al.]
gross-substitutes ➔ ∃ Walrasian equilibrium

$$D(p) = \arg\max_x \{f(x) - p^{\mathrm{T}} x\}$$

Def: gross-substitutes (GS) condition for valuations on $\{0,1\}^n$:
$\forall p \in \mathbb{R}^n$, $\forall j \in N$, $q = p + \lambda e_j$, $\forall x \in D(p)$, $\exists y \in D(q)$:

$$y(k) \geq x(k) \quad (\forall k \in N \setminus \{j\})$$

- higher price for some item, more demand for other items

Thm for valuations on $\{0,1\}^n$ [Fujishige-Yang2003]
gross-substitutes ⟷ M♮-concavity

Definition of M♮-concave Function

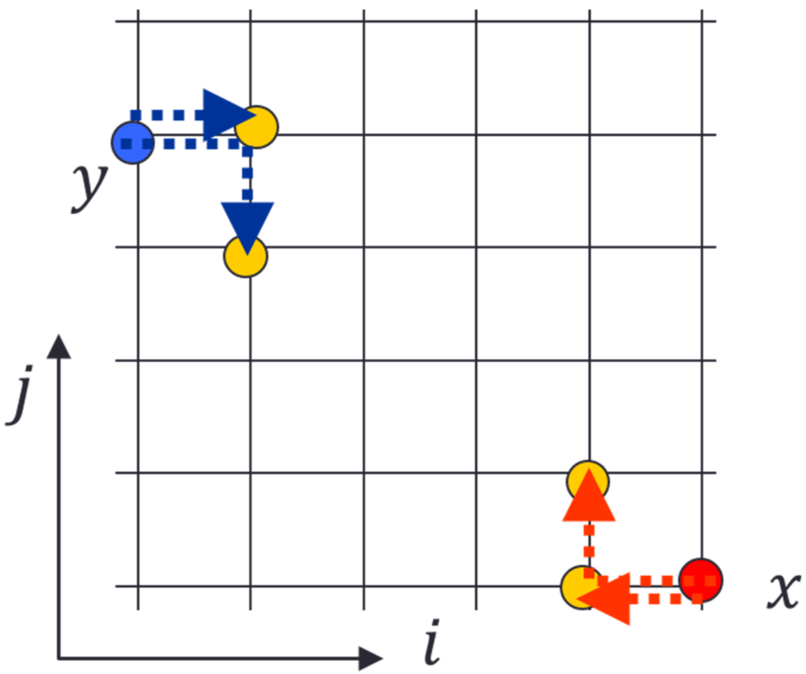
[Murota-Shioura 99]

Def: $f: \mathbb{Z}^n \to \mathbb{R} \cup \{+\infty\}$ is M♮-concave ⟺

$\forall x, y \in \mathbb{Z}^n, \forall i: x(i) > y(i)$:

(i) $f(x) + f(y) \le f(x - \chi_i) + f(y + \chi_i)$, or

(ii) $\exists j: x(j) < y(j)$ s.t. $f(x) + f(y) \le f\big(x - \chi_i + \chi_j\big) + f(y + \chi_i - \chi_j)$



variant of M-concave fn [Murota 96]

valuated matroid [Dress-Wenzel90]

≒ M♮-concave fn on $\{0,1\}^n$

Examples of M♮-concave Fns on $\{0,1\}^n$

- additive (linear) valuation: with values $v = (v(1), \ldots, v(n))$,

$$f(x) = v^{\mathrm{T}} x \qquad (x \in \{0,1\}^n)$$

- symmetric concave valuation: with concave fn $\varphi: \mathbb{Z} \to \mathbb{R}$,

$$f(x) = \varphi(\textstyle\sum_j x(j)) \qquad (x \in \{0,1\}^n)$$

- unit-demand valuation: with values $v = (v(1), \ldots, v(n))$,

$$f(x) = \max_{j: x(j)=1} v(j) \qquad (x \in \{0,1\}^n)$$

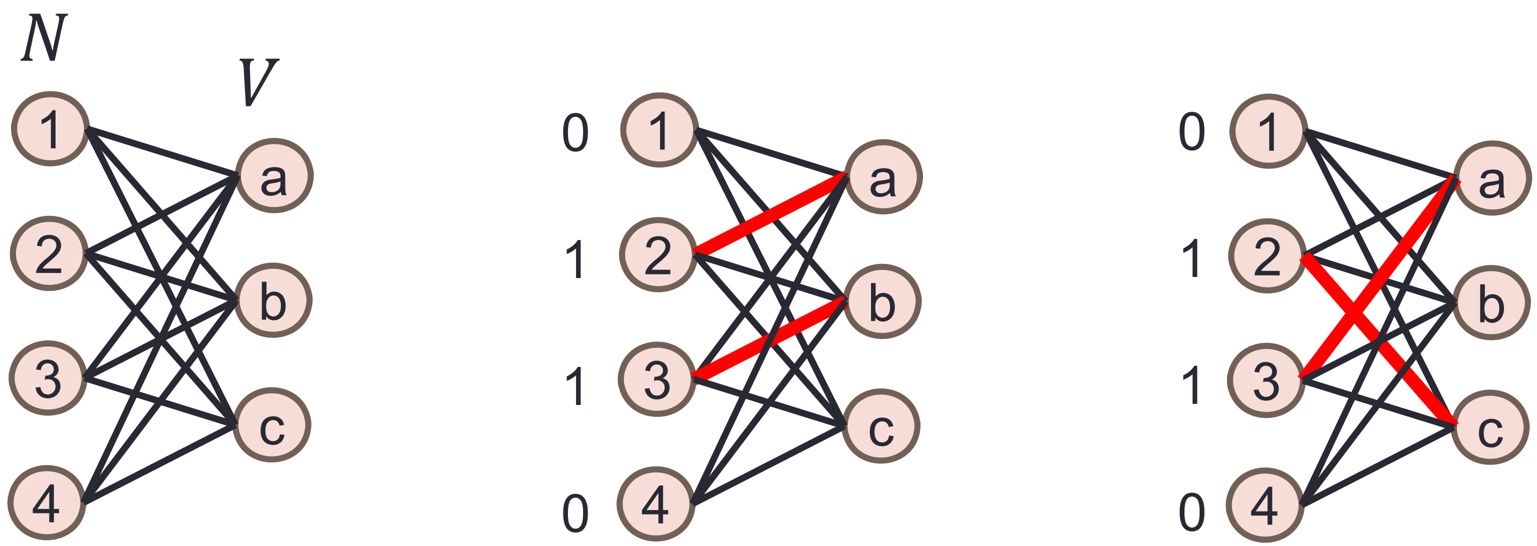
Assignment Valuation

valuation $f\colon \{0,1\}^n \to \mathbb{Z}$ defined by

max-weight one-to-one matching

w_{jh} $(j \in N, h \in V)$: edge weight

$$f(x) = \max\left\{ \sum_{(j,h)\in M} w_{jh} \,\middle|\, M\text{: matching covering } \{j \in N \mid x(j) = 1\} \right\}$$



M♮-concave Fns from Matroids

- matroid rank: $f(X) = r(X) \quad (X \subseteq N)$

- weighted matroid rank:

  $f(X) = \max\{\sum_{j \in Y} w(j) \mid Y \subseteq X$, matroid indep. set$\}$ $(X \subseteq N)$

  [Shioura2012]

- matroid rank sum: $f(X) = \alpha_1 r_1(X) + \cdots + \alpha_k r_k(X) \ (X \subseteq N)$

  r_i & r_{i+1} are strong quotient, $\alpha_i \geq 0$ [Shioura2012]

- valuated matroid = M♮-concave fn on $\{0,1\}^n$

※ function on $\{0,1\}^n$ ←→ fn on subsets of $N = \{1, \ldots, n\}$

Equilibrium Existence in Multi-Demand/Multi-Unit Model

Def: gross-substitutes (GS) condition for valuation on $\{0,1\}^n$:
$\forall p \in \mathbb{R}^n, \quad q = p + \lambda e_j, \quad \forall x \in D(p), \quad \exists y \in D(q)$:

$$y(k) \geq x(k) \quad (\forall k \in N \setminus \{j\})$$

GS naturally extends to valuations on $[0,u]_{\mathbb{Z}}$,

but not sufficient for equilibrium existence

Example:

$f_1, f_2 : [0,u]_{\mathbb{Z}} \to \mathbb{Z}, u = (2,1)$
satisfy (GS),
but no equilibrium

$f_1(x)$	$x(1)=0$	1	2
$x(2)=0$	0	3	6
$x(2)=1$	4	5	6

$f_2(x)$	$x(1)=0$	1	2
$x(2)=0$	0	4	4
$x(2)=1$	1	5	5

$$\max\{f_1(x) + f_2(u-x)\}$$
$$= f_1(1,1) + f_2(1,0) = 5 + 4$$

[cf. Milgrom-Strulovici (2019)]

Conditions for Equilibrium Existence in Multi-Demand/Multi-Unit Model

Thm for valuations on $[0, u]_{\mathbb{Z}}$ [Danilov-Koshevoy-Murota 2001]

M♮-concavity ➔ ∃ Walrasian equilibrium

stronger variants of GS equivalent to M♮-concavity

e.g.: Murota-Tamura03, Danilov-Koshevoy-Lang03, et al.

Def: strong-substitutes (SS) condition: [Milgrom-Strulovici2009]

(GS) holds when all units of items are regarded as distinct items

Def: GS + Law of Aggregate Demand (LAD) condition: [Murota-Shioura-Yang2013]

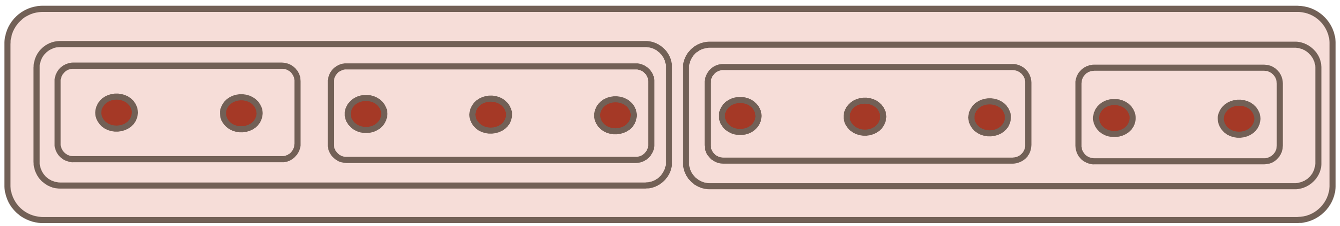
$\forall p \in \mathbb{R}^n, \quad q = p + \lambda e_j, \qquad \forall x \in D_i(p), \ \exists y \in D_i(q)$:

$$y(k) \geq x(k) \ (\forall k \in N \setminus \{j\}), \quad \sum_{k \in N} y(k) \leq \sum_{k \in N} x(k)$$

Thm (SS) ⟷ (GS+LAD) ⟷ M♮-concavity

Examples of M♮-concave Fns on $[0,u]_Z$

- additive (linear): with values $v = (v(1), \dots, v(n))$,
$$f(x) = v^{\mathrm{T}} x \qquad (x \in [0,u]_{\mathbb{Z}})$$
- symmetric concave: with concave fn $\varphi: \mathbb{Z} \to \mathbb{Z}$,
$$f(x) = \varphi(\textstyle\sum_j x(j)) \qquad (x \in [0,u]_{\mathbb{Z}})$$
- separable concave: with concave fns $\varphi_j: \mathbb{Z} \to \mathbb{Z}\ (j \in N)$,
$$f(x) = \textstyle\sum_j \varphi_j(x(j)) \qquad (x \in [0,u]_{\mathbb{Z}})$$
- laminar concave: with laminar family $\{A_1, A_2, \dots, A_k\}$
  and concave fns $\varphi_i: \mathbb{Z} \to \mathbb{Z}\ (i = 1, \dots, k)$,
$$f(x) = \textstyle\sum_i \varphi_i\left(\sum_{j \in A_i} x(j)\right)$$
[Danilov-Koshevoy-Murota98,01]



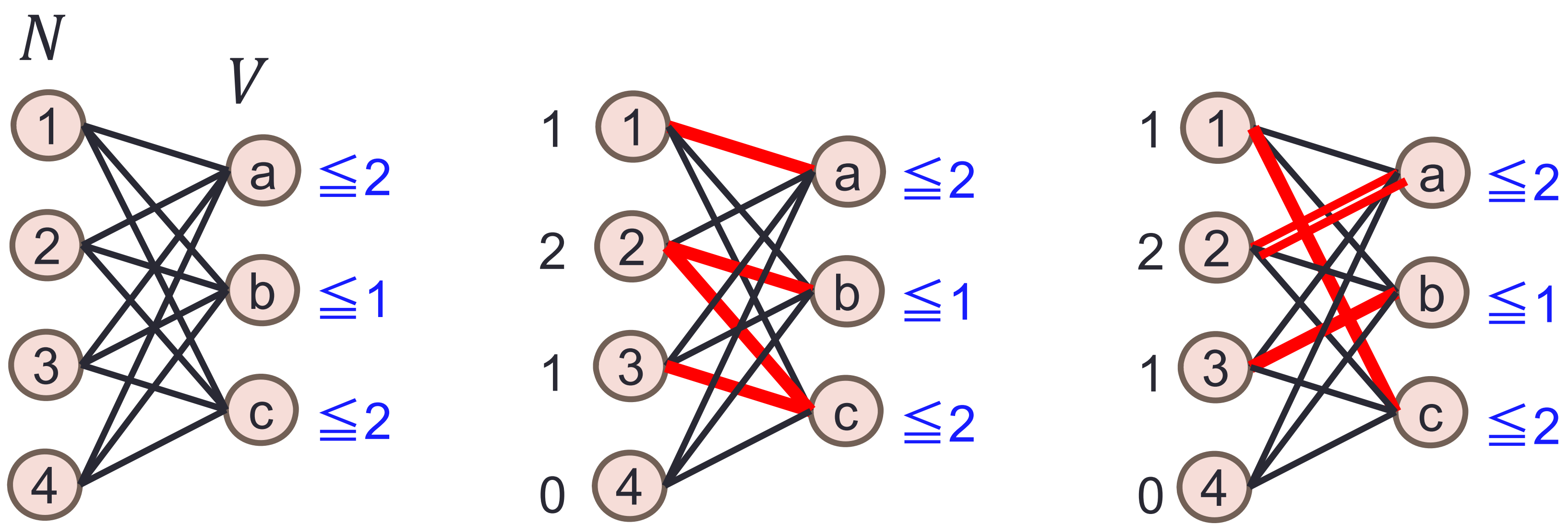
Multi-Unit Assignment Valuation

defined by max-weight many-to-many matching

w_{jh} $(j \in N, h \in V)$: edge weight

$d(h) \in \mathbb{Z}_+$ $(h \in V)$: degree upper-bound

$$f(x) = \max\{\textstyle\sum_{(j,h)} w_{jh} y_{jh} \mid \sum_h y_{jh} = x(j)\ (j \in N),$$
$$\textstyle\sum_j y_{jh} \leq d(h)\ (h \in V),\ y_{jh} \in \mathbb{Z}_+\ (j \in N, h \in V)\}$$



Indirect Utility Function

valuation fn $f : [0, u]_{\mathbb{Z}} \to \mathbb{Z}$

demand set $D(p) = \arg\max_x \{f(x) - p^T x\}$

indirect utility fn $V(p) = \max_x \{f(x) - p^T x\}$

- used in design & analysis of auctions
- given explicitly as bidding language

(e.g., Product-Mix auction [Klemperer2010])

Prop

- indirect utility V is convex
- valuation f is SS ➔ indirect utility V is submodular
- V can be regarded as function on $\mathbb{Z}^n$ [cf. Ausubel 2006]

Examples of Indirect Utility Fns

- additive (linear) $f(x) = \sum_j v(j) x(j)$

  $\Rightarrow V(p) = \sum_{j=1}^{n} \max\{0, v(j) - p(j)\}$

- unit-demand $f(x) = \max_{j : x(j) = 1} v(j)$

  $\Rightarrow V(p) = \max_{1 \le j \le n} \{v(j) - p(j)\}$ ("max-payoff" function)

- multi-unit assignment

$$f(x) = \max\{\textstyle\sum_{(j,h)} w_{jh} y_{jh} \mid \sum_h y_{jh} = x(j)\ (j \in N),$$
$$\textstyle\sum_j y_{jh} \le d(h)\ (h \in V),\ y_{jh} \in \mathbb{Z}_+\ (j \in N, h \in V)\}$$

$\Rightarrow V(p) = \sum_{h \in V} d(h) \max_{1 \le j \le n} \{w_{jh} - p(j)\}$

("max-payoff-sum" function)

Product-mix auction
[Klemperer2010]

Indirect Utility Fn of SS Valuation

Thm: indirect utility of SS valuations

$\subseteq$ difference of "max-payoff-sum" functions

[Klemperer2010, Baldwin-Klemperer2021]

$$\max_x\{f(x) - p^{\mathrm{T}}x\}$$

$$= \sum_h d(h) \max_{1 \le j \le n}\{w_{jh} - p(j)\} - \sum_k d'(k) \max_{1 \le j \le n}\{w'_{jk} - p(j)\}$$

※ computation/checking validity of representation is "difficult"

- given f, computation of w_{jh}, w'_{jk}, d, d' has pseudo-poly. lower-bound

  [Goldberg-Lock-Marmolejo-Cossío2022]
- given w_{jh}, w'_{jk}, d, d', checking SS of f is coNP-complete

  [Baldwin-Goldberg-Klemperer-Lock2021]

Indirect Utility and L♮-convexity

- convexity concepts in Discrete Convex Analysis
  - M♮-convexity and L♮-convexity
- conjugacy between L♮-convexity / M♮-convexity

M-L Conjugacy Thm: (Murota 1998) by Legendre transform,
M♮-convex fn $f(x)$ $\Leftrightarrow$ L♮-convex fn $g(p)$

Legendre transform $g(p) = \max_{x}\{p^{\mathrm{T}}x - f(x)\}$

Thm M♮-concave $\Leftrightarrow$ Strong-Substitutes

$\Downarrow$

Thm indirect utility V on $\mathbb{Z}^n$ is L♮-convex
$\Leftrightarrow$ valuation f is SS (cf. Murota-Shioura-Yang 2013)

Definition of L♮-convex Fn

continuous fn $g: \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is convex

⇔ mid-point convex : $\forall p, q \in \mathbb{R}^n$,

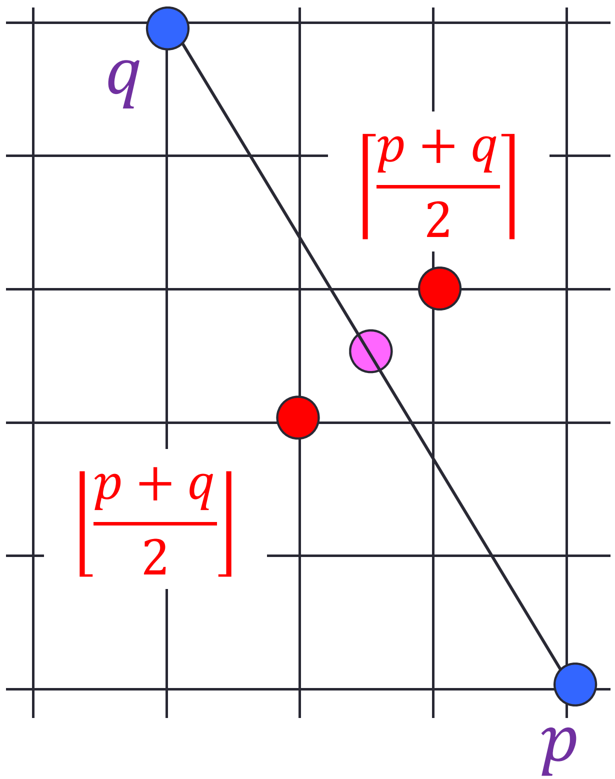
$$g(p) + g(q) \geq 2g\left(\frac{p+q}{2}\right)$$

[Murota1998]
[Fujishige-Murota2000]

Def : $g: \mathbb{Z}^n \to \mathbb{R} \cup \{+\infty\}$ is L♮-convex

⇔ discrete mid-point convex : $\forall p, q \in \mathbb{Z}^n$,

$$g(p) + g(q) \geq g\left(\left\lceil\frac{p+q}{2}\right\rceil\right) + g\left(\left\lfloor\frac{p+q}{2}\right\rfloor\right)$$



※ original definition
in terms of submodularity

Original Definition of L♮-convexity by Submodularity

[Murota1998]
[Fujishige-Murota2000]

Def: $g: \mathbb{Z}^n \to \mathbb{R} \cup \{+\infty\}$ is L♮-convex

$\Leftrightarrow$ $\tilde{g}: \mathbb{Z} \times \mathbb{Z}^n \to \mathbb{R} \cup \{+\infty\}$ is submodular

$$\tilde{g}(p_0, p) = g(p - p_0 \mathbf{1}) \qquad ((p_0, p) \in \mathbb{Z} \times \mathbb{Z}^n)$$

$$\mathbf{1} = (1,1, \ldots, 1)$$

Prop: L♮-convex $\Rightarrow$ submodular on $\mathbb{Z}^n$

L♮-convex fn on $\{0,1\}^n$ $\Leftrightarrow$ submodular set fn

Examples of L♮-convex Fn

- quadratic fn $g(p) = p^T A p$ is L♮-convex
  $\Leftrightarrow$ $a_{ij} \le 0 \ (i \ne j), \quad \sum_j a_{ij} \ge 0$

$$\begin{bmatrix} 4 & & & -1 \\ & 3 & -2 & \\ & -2 & 3 & -1 \\ -1 & & -1 & 5 \end{bmatrix}$$

- range: $g(p) = \max\{0, p_1, p_2, \dots, p_n\} - \min\{0, p_1, p_2, \dots, p_n\}$

- min-cost tension problem
  $g(p) = \sum_{i=1}^{n} \varphi_i(p_i) + \sum_{i,j} \psi_{ij}(p_i - p_j)$ (φ_i, ψ_{ij}: univariate conv)
  - dual of min-cost flow
  - “energy fn” in image processing

Representation of L♮-convex Fn

Thm: indirect utility of SS valuations

$\subseteq$ difference of "max-payoff-sum" functions

[Klemperer2010, Baldwin-Klemperer2021]

Thm indirect utility V on $\mathbb{Z}^n$ is L♮-convex

$\Leftrightarrow$ valuation f is SS

$\Downarrow$

Cor: $\forall$ L♮-conv fn g, $\exists\, g_1, g_2$: "simple" L♮-conv fn

s.t. $g = g_1 - g_2$

Summary of 1st Half

Discrete Convexity/Concavity in Multi-Item Auctions

- Gross(Strong)-substitutes valuation = M♮-concave fn
- Indirect Utility = L♮-convex fn

- Shioura, Tamura (2015) J. Operations Research Society Japan
  "Gross substitutes condition and discrete concavity for multi-unit valuations"
- Murota (2016) J. Mechanism & Institution Design
  "Discrete convex analysis: A tool for economics and game theory"

Murota (2003) Discrete Convex Analysis, SIAM

Equilibrium Computation and Discrete Convex Optimization

computation of

- equilibrium allocation
  = constrained M♮-concave maximization
- equilibrium price = L♮-convex minimization

Setting of Auction: Bidder's Information

Case 1: values of valuation fn f_i are available

Case 2: bidder's valuation fn f_i is given implicitly

2-1: values of indirect utility $V_i(p)$ are available

2-2: demand sets $D_i(p)$ are available

demand set $D(p) = \arg\max_x \{f(x) - p^T x\}$

indirect utility fn $V(p) = \max_x \{f(x) - p^T x\}$

Setting of Auction: Bidder's Information

Case 1: values of valuation fn f_i are available

Case 2: bidder's valuation fn f_i is given implicitly

2-1: values of indirect utility $V_i(p)$ are available

2-2: demand sets $D_i(p)$ are available

demand set $D(p) = \arg\max_x \{f(x) - p^T x\}$

indirect utility fn $V(p) = \max_x \{f(x) - p^T x\}$

Equilibrium Computation from Valuations

Case 1: values of valuation fn f_i are available

Thm: Suppose equilibrium exists.
equilibrium allocation ←→ optimal allocation of

$$\begin{aligned} &\text{max.} \quad f_1(x_1) + \cdots + f_m(x_m) \\ &\text{sub. to } x_1 + \cdots + x_m = u, \;\; x_i \in \mathbb{Z}_+^n \end{aligned}$$

- optimal allocation → equilibrium allocation
- use duality & properties of SS (M♮-concavity)
  → equilibrium price

Equilibrium Computation from Valuations

Case 1: values of valuation fn f_i are available

multi-demand/ multi-unit $f: [0, u]_{\mathbb{Z}} \to \mathbb{R}$

multi-demand/ single-unit $f: \{0,1\}^n \to \mathbb{R}$

single-demand/ single-unit $f: N \to \mathbb{R}$

single-demand model

- optimal allocation = max-weight matching

multi-demand model with SS/GS valuations

- optimal allocation
  = maximization of M♮-concave fn
  over M♮-convex set

efficient algorithms available

Setting of Auction: Bidder's Information

Case 1: values of valuation fn f_i are available

Case 2: bidder's valuation fn f_i is given implicitly

2-1: values of indirect utility $V_i(p)$ are available

2-2: demand sets $D_i(p)$ are available

demand set $D(p) = \arg\max_x \{f(x) - p^T x\}$

indirect utility fn $V(p) = \max_x \{f(x) - p^T x\}$

Equilibrium Computation from Indirect Utility

Case 2-1: values of indirect utility $V_i(p)$ are available

$$V(p) = \max_x \{f(x) - p^T x\}$$

Lyapunov fn: $L(p) = p^{\mathrm{T}} u + \sum_i V_i(p)$ [Ausubel 2006]

Thm: Assume SS for f_i.

(i) V_i and L are submodular

(ii) p : minimizer of L ⟷ p: equilibrium price

(iii) ∃ integral minimizer of L

equilibrium price computation

= Minimization of Lyapunov function

Lyapunov Fn and L♮-convexity

Case 2-1: values of indirect utility $V_i(p)$ are available

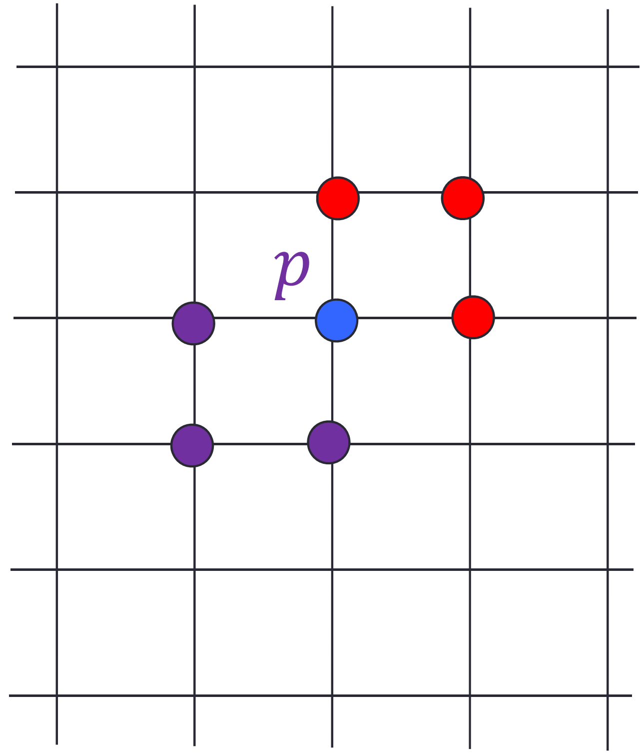
Thm: indirect utility V_i is L♮-convex
⇔ valuation f_i is SS (cf. Murota-Shioura-Yang 2013)

Thm: Assume SS for f_i.
Lyapunov fn = L♮-convex fn (cf. Murota-Shioura-Yang 2013)

equilibrium price computation
= Minimization of L♮-convex function

Minimization of L♮-convex Fn

$g: \mathbb{Z}^n \to \mathbb{R} \cup \{+\infty\}$ --- L♮-convex



Problem: Minimize $g(p)$ for $p \in \mathbb{Z}^n$

Thm (optimality condition): (Murota 1998)

p: global min

⟺ p: local min

$$g(p) \leq g(p \pm e_X) \quad (\forall X \subseteq N)$$

$e_X \in \{0,1\}^n$: characteristic vector

$$e_X(v) = \begin{cases} 1 & (v \in X) \\ 0 & (v \in N \setminus X) \end{cases}$$

Algorithms for L♮-convex Minimization

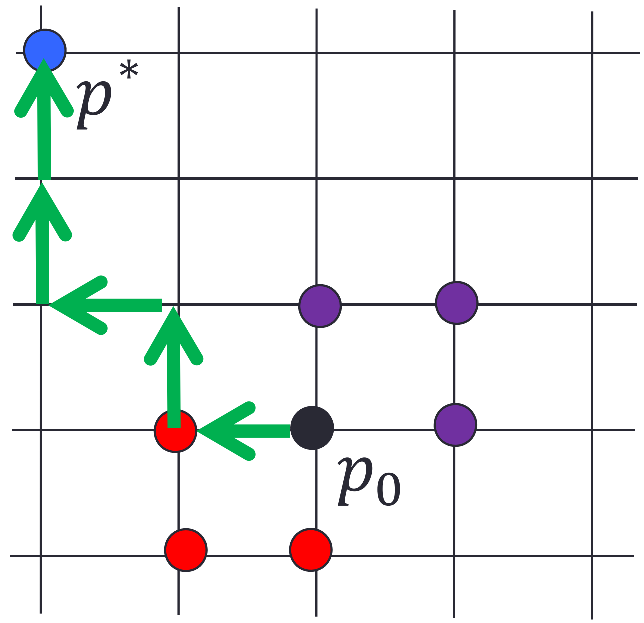
Steepest Descent (Up&Down)

Step 0: $p := p_0$ (initial pt)

Step 1: Take $\delta \in \{+1, -1\}$ & $X \subseteq N$
to minimize $g(p + \delta e_X)$

Step 2: $g(p + \delta e_X) \geq g(p) \rightarrow$ finish
(current p is opt sol)

Step 3: $p := p + \delta e_X$, Go to Step 1



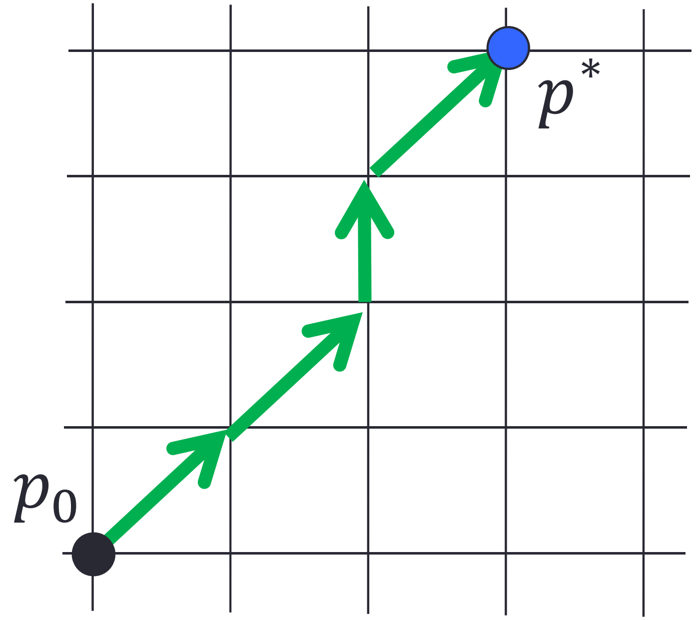
Steepest Descent (Up)

Step 0: $p := p_0$ (sufficiently small)

Step 1: Take $X \subseteq N$
to minimize $g(p + e_X)$

Step 2: $g(p + e_X) \geq g(p) \rightarrow$ finish
(current p is opt sol)

Step 3: $p := p + e_X$, Go to Step 1



Speed-Up of Algorithms

steepest-descent algorithms are simple but slow

- each iteration moves the vector by one step only
- # of iterations = “distance” between initial sol. & opt. sol.
- computation of steepest-descent direction is time-consuming
  $\leftarrow$ submodular fn minimization

computational techniques for improvement

- use of long step
- scaling technique
- DC (Difference of Convex) approach

- some experimental results [Baldwin-Bichler-Fichtl-Klemperer2022]

Techniques for Improvement: Long Step

use longest possible step length

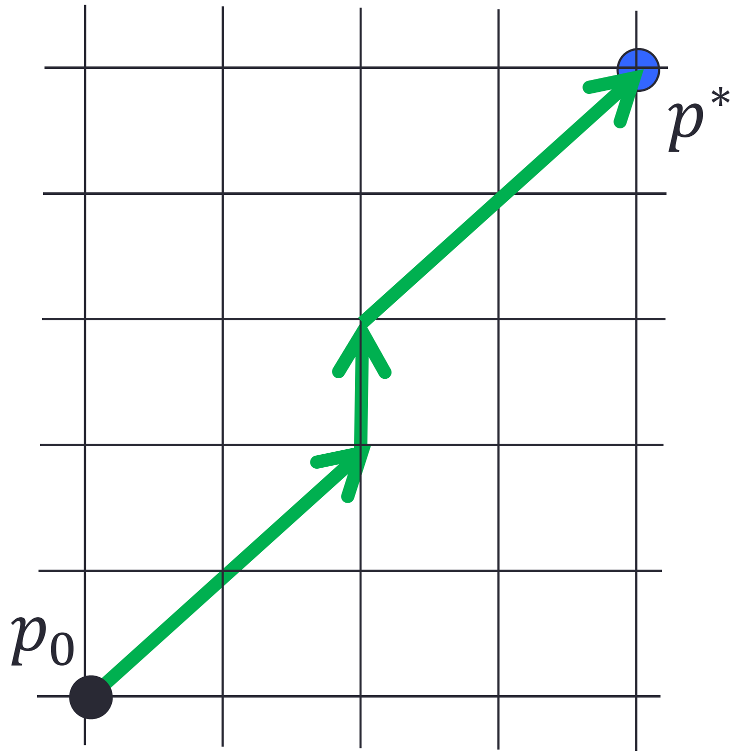
[Shioura2017, Baldwin-Goldberg-Klemperer-Lock2021]

Steepest Descent Long Step (Up)

Step 3: Take maximum λ with

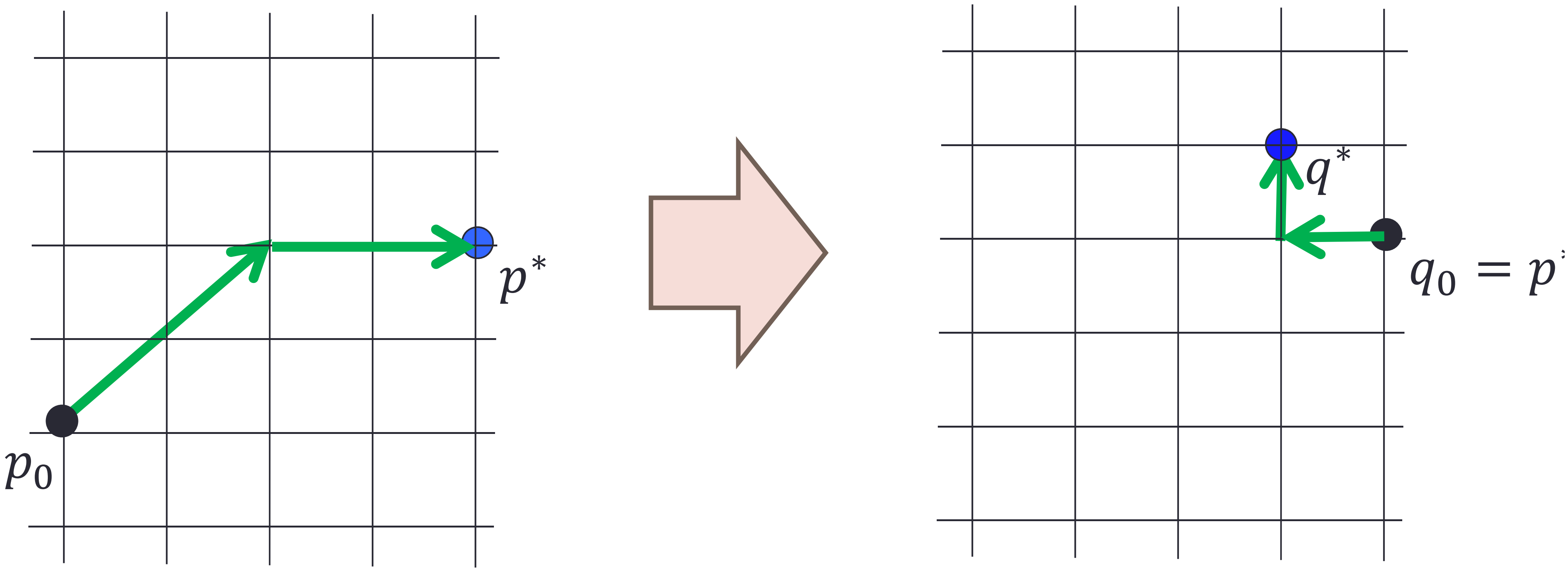
$$\frac{g(p+\lambda e_X)-g(p)}{\lambda} = g(p+e_X)-g(p)$$

Step 4: $p := p+\lambda e_X$, Go to Step 1



Techniques for Improvement: Scaling

- long step length → short step length [Murota2003]
- neighborhood of approx. minimizer contains exact minimizer



step length $\lambda = 2$ step length $\lambda = 1$

(Discrete) DC Approach

[Maehara-Murota2015, Baldwin-Bichler-Fichtl-Klemperer2022]

- "minimize $g(p + \delta e_X)$" is time-consuming for general L♮-conv. fn
- simpler L♮-conv. fn (e.g., max-payoff-sum) is easier to handle

⇩

- use representation $g = g_1 - g_2$ with "simple" L♮-conv g_1, g_2
- apply (discrete) DC algorithm for minimizing $g_1 - g_2$

[Pham Dinh-Souad1986][Maehara-Murota2015]

$$g(p) = \sum_h d(h) \max_{1\le j\le n}\{w_{jh} - p(j)\} - \sum_k d'(k) \max_{1\le j\le n}\{w'_{jk} - p(j)\}$$

Setting of Auction: Bidder's Information

Case 1: values of valuation fn f_i are available

Case 2: bidder's valuation fn f_i is given implicitly

2-1: values of indirect utility $V_i(p)$ are available

2-2: demand sets $D_i(p)$ are available

- discussed most extensively in the literature
- equilibrium price is computed by Iterative Auction

Iterative Auction

- iterative auction: protocol (algorithm) for finding equilibrium price
  - repeatedly update price using bidders' demand sets

Step 0. set initial price $p = (p_1, \ldots, p_n)$

Step 1. bidders report demand sets $D_i(p)$

Step 2. If ∃ allocation $(x_1^*, x_2^*, \ldots, x_m^*)$ s.t. $x_i^* \in D_i(p)$

(or alternative condition is satisfied)

➔ stop (p is equilibrium price)

Step 3. update p appropriately. Go to Step 1.

- ascending auction: p increases monotonically

Ascending Auction Using Lyapunov Fn

[Ausubel 2006]

Lyapunov fn: $L(p) = p^{\mathrm{T}} u + \sum_i V_i(p)$ $\quad V_i(p) = \max_x \{f_i(x) - p^T x\}$

Idea: $L(p + e_Y) - L(p)$ can be obtained from demand sets

$$L(p + e_Y) - L(p) = \sum_{j \in Y} u(j) - \sum_i \min \left\{ \sum_{j \in Y} x(j) \,\middle|\, x \in D_i(p) \right\}$$

Ascending Auction

- Step 0: $p :=$ sufficiently small integral vector (e.g., $\mathbf{0}$)
- Step 1: find $X \subseteq N$ minimizing $L(p + e_X)$
- Step 2: $L(p + e_X) \geq L(p)$
  ➔ stop (p is equilibrium price)
- Step 3: $p := p + e_X$, Go to Step 1

$$e_X = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Analysis of Ascending Auction by DCA

Thm: Assume SS for f_i.
Lyapunov fn = L♮-convex fn (cf. Murota-Shioura-Yang 2013)

equilibrium price computation = Minimization of L♮-convex fn

Obs: Ascending Auction
= Steepest Descent (Up) for Lyapunov fn

exact bound for Steepest Descent (Up) (MSY2013)

Cor: $\min\{\|p^* - p_0\|_\infty \mid p^*: \text{integer equil.},\ p^* \geq p_0\}$

Iterative Auctions from DCA

- Ascending Auction (Ausubel 2006) = Steepest Descent (Up)
- Descending Auction (Ausubel 2006) = Steepest Descent (Down)
  - merit: prices move monotonically
  - demerit: sufficiently small/large initial prices
- Greedy Auction (Murota-Shioura-Yang2013)
  = Steepest Descent (Up&Down)
  - merit: initial prices can be chosen arbitrarily
    # of iterations is small
  - demerit: prices move up & down
- Two-Phase Auction (Murota-Shioura-Yang2013)
  - Ascending ➔ Descending
  - merit: initial prices can be chosen arbitrarily
    prices move almost monotonically
  - demerit: larger # of iterations

Two-Phase Auction

Ascending
+ Descending

Step 0: $p :=$ any integral vector
Step A1: find $X \subseteq N$ minimizing $L(p + e_X)$
Step A2: $L(p + e_X) \geq L(p) \rightarrow$ go to Step D1
Step A3: $p := p + e_X$, Go to Step A1
Step D1: find $Y \subseteq N$ minimizing $L(p - e_Y)$
Step D2: $L(p - e_Y) \geq L(p) \rightarrow$ stop (p is equilibrium)
Step D3: $p := p - e_Y$, Go to Step D1

- merit
  - any initial vector can be used
  - almost monotone w.r.t. prices
- demerit
  - more iterations than Greedy

Thm: # iters $\leqq$ 3 × (# iters of Greedy)

Summary

Discrete Convexity/Concavity in Multi-Item Auctions

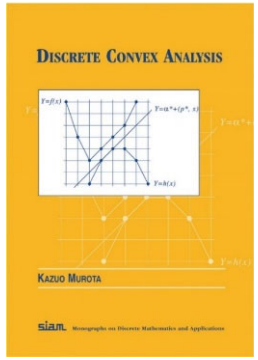
- Gross(Strong)-substitutes valuation = M♮-concave fn
- Indirect Utility = L♮-convex fn

Equilibrium Computation and Discrete Convex Optimization

- equilibrium allocation
  = constrained M♮-concave maximization
- equilibrium price = L♮-convex minimization

Surveys & Books

- Shioura, Tamura (2015)   J. Operations Research Society Japan
  "Gross substitutes condition and discrete concavity for multi-unit valuations"
- Shioura (2017)   J. Operations Research Society Japan
  "Algorithms for L-convex function minimization"
- Murota (2016)   J. Mechanism & Institution Design
  "Discrete convex analysis: A tool for economics and game theory"



- Murota (2003) Discrete Convex Analysis, SIAM
- two Japanese books
  - 室田, 塩浦 (2013) 離散凸解析と最適化アルゴリズム
  - 田村 (2009) 離散凸解析とゲーム理論