Symmetry and hardness of submodular maximization problems

Jan Vondrák[1]

[1] Department of Mathematics
Princeton University

Submodularity

Definition

A function $f : 2^X \to \mathbb{R}$ is submodular if for any S, T,

$$f(S \cup T) + f(S \cap T) \leq f(S) + f(T).$$

Submodularity

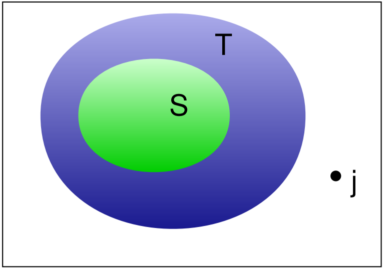
Definition

A function $f : 2^X \to \mathbb{R}$ is submodular if for any S, T,

$$f(S \cup T) + f(S \cap T) \leq f(S) + f(T).$$

Alternative definition: Define the *marginal value of element j*,

$$f_S(j) = f(S \cup \{j\}) - f(S).$$



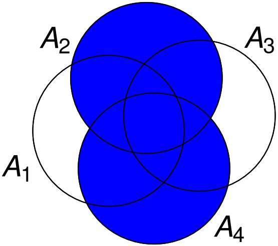
f is submodular, if $\forall S \subset T, j \notin T$:

$$f_S(j) \geq f_T(j).$$

Examples of Submodular Functions

Coverage function:
Given $A_1, \ldots, A_n \subset U$,

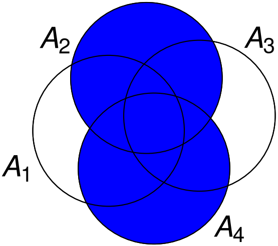
$$f(S) = \left| \bigcup_{j \in S} A_j \right|.$$

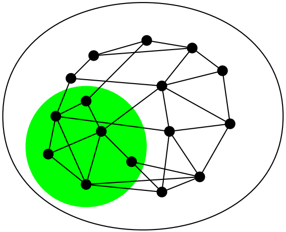


Examples of Submodular Functions

Coverage function:
Given $A_1, \ldots, A_n \subset U$,

$$f(S) = \left| \bigcup_{j \in S} A_j \right|.$$





Cut function:

$$\delta(T) = |e(T, \overline{T})|$$

Matroids

Definition

A matroid on N is a system of *independent sets* $\mathcal{M} \subset 2^N$, satisfying

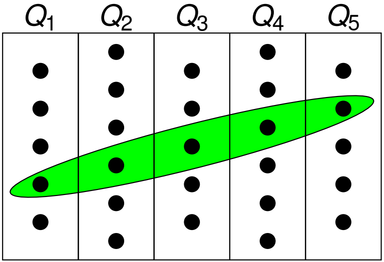
1. $\forall B \in \mathcal{M}, A \subset B \;\Rightarrow\; A \in \mathcal{M}$.
2. $\forall A, B \in \mathcal{M}, |A| < |B| \;\Rightarrow\; \exists x \in B \setminus A;\; A \cup \{x\} \in \mathcal{M}$.

Matroids

Definition

A matroid on N is a system of *independent sets* $\mathcal{M} \subset 2^N$, satisfying

1. $\forall B \in \mathcal{M}, A \subset B \;\Rightarrow\; A \in \mathcal{M}$.
2. $\forall A, B \in \mathcal{M}, |A| < |B| \;\Rightarrow\; \exists x \in B \setminus A;\; A \cup \{x\} \in \mathcal{M}$.



Example: *partition matroid*

S is independent, if
$|S \cap Q_i| \leq 1$ for each Q_i.

Submodular optimization

Submodular functions:

1. A general framework capturing useful combinatorial structure
2. A natural property to be assumed in certain settings (utilities/valuations with diminishing returns)

Submodular optimization

Submodular functions:

1. A general framework capturing useful combinatorial structure
2. A natural property to be assumed in certain settings (utilities/valuations with diminishing returns)

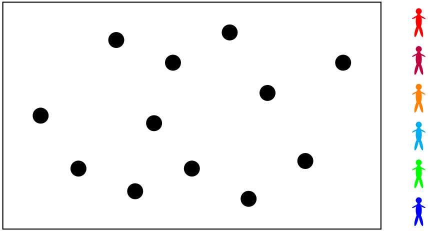
Oracle model: For a given set S, we can query $f(S)$.

Submodular minimization:

- Any submodular function can be *minimized* in polynomial time [Schrijver, Fleischer-Fujishige-Iwata 2001]

Maximum Submodular Welfare

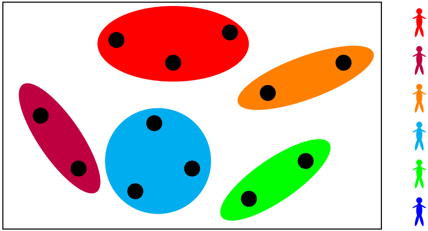
Given: $|X| = m$ items, n players with *utility functions* $w_i : 2^X \to \mathbb{R}_+$.
Goal: Find an allocation of disjoint sets $S_1, \ldots, S_n$ to the n players.



$|X| = m$

Maximum Submodular Welfare

Given: $|X| = m$ items, n players with *utility functions* $w_i : 2^X \to \mathbb{R}_+$.
Goal: Find an allocation of disjoint sets $S_1, \ldots, S_n$ to the n players.



$|X| = m$

Allocation $(S_1, S_2, \ldots, S_n)$

has value $\sum_{i=1}^{n} w_i(S_i)$.

We want to maximize the *social welfare* $\sum_{i=1}^{n} w_i(S_i)$.

Submodular maximization

Submodular maximization problems:
NP-hard in special cases such as Max Cut, Max k-cover, Maximum Submodular Welfare, Maximum Bisection.

Submodular maximization

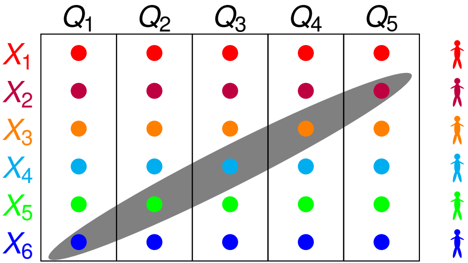
Submodular maximization problems:
NP-hard in special cases such as Max Cut, Max k-cover, Maximum Submodular Welfare, Maximum Bisection.

Early approximation results:

- Greedy $(1-1/e)$-approximation for $\max\{f(S) : |S| \leq k\}$ where $f(S)$ is monotone submodular [Nemhauser,Wolsey,Fisher'78]
- $(1-1/e+\epsilon)$ impossible in the oracle model [NWF'78] and NP-hard in the special case of Max k-cover [Feige'98]
- Greedy $1/2$-approx. for $\max\{f(S) : S \in \mathcal{M}\}$, matroid $\mathcal{M}$ [NWF'78]
- Maximum Submodular Welfare is a special case of a matroid constraint, hence there is a greedy $1/2$-approximation [Lehmann,Lehmann,Nissan'01]

Submodular Welfare → matroid constraint

Submodular Welfare: *Given n players with submodular utility functions $w_i : 2^X \to \mathbb{R}_+$. Allocate the items so as to maximize $\sum_{i=1}^{n} w_i(S_i)$.*



Reduction:

Create n clones of each item,
$f(S) = \sum w_i(S \cap X_i)$,
$\mathcal{M} = \{S : \forall i; |S \cap Q_i| \le 1\}$
(a partition matroid).

The Submodular Welfare Problem is equivalent to $\max\{f(S) : S \in \mathcal{M}\}$.

Recent results

- $(1-1/e)$-approximation for Maximum Submodular Welfare [V.'08], optimal due to [Khot et al.'05]
- $(1-1/e)$-approximation for monotone submodular maximization over a matroid [Calinescu,Chekuri,Pál,V.'08]
- $(1-1/e-\epsilon)$-approximation for monotone submodular maximization with knapsack constraints [Kulik,Tamir,Shachnai'09]
- $2/5$-approximation for unconstrained (non-monotone) submodular maximization; $(1/2+\epsilon)$ impossible in the oracle model [Feige,Mirrokni,V.'07]
- $(1/5-\epsilon)$-approximation for non-monotone submodular maximization with knapsack constraints [Lee et al.'09]

Recent results

- $(1 - 1/e)$-approximation for Maximum Submodular Welfare [V.'08], optimal due to [Khot et al.'05]
- $(1 - 1/e)$-approximation for monotone submodular maximization over a matroid [Calinescu,Chekuri,Pál,V.'08]
- $(1 - 1/e - \epsilon)$-approximation for monotone submodular maximization with knapsack constraints [Kulik,Tamir,Shachnai'09]
- $2/5$-approximation for unconstrained (non-monotone) submodular maximization; $(1/2 + \epsilon)$ impossible in the oracle model [Feige,Mirrokni,V.'07]
- $(1/5 - \epsilon)$-approximation for non-monotone submodular maximization with knapsack constraints [Lee et al.'09]

... all using a **multilinear relaxation** of the problem

Multilinear relaxation

Discrete Problem

$\max\{f(S) : S \in \mathcal{F}\}$

Continuous Problem

$\max\{F(x) : x \in P(\mathcal{F})\}$

Multilinear relaxation

Discrete Problem

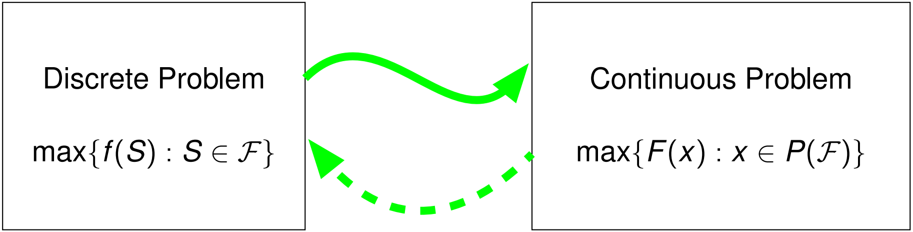
$\max\{f(S) : S \in \mathcal{F}\}$

Continuous Problem

$\max\{F(x) : x \in P(\mathcal{F})\}$

- $P(\mathcal{F}) = \text{conv}\{\mathbf{1}_S : S \in \mathcal{F}\} \subset [0,1]^X$.

Multilinear relaxation



- $P(\mathcal{F}) = \mathrm{conv}\{\mathbf{1}_S : S \in \mathcal{F}\} \subset [0,1]^X$.
- $F(x) = \mathbb{E}[f(\hat{x})]$, where $\hat{x}$ is obtained by rounding each x_j independently to $\{0,1\}$:

$$F(x) = \sum_{S \subseteq X} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i).$$

Properties of the multilinear relaxation

$F(x) = \mathbb{E}[f(\hat{x})]$ has the following properties:

- $f(S)$ is monotone $\Leftrightarrow \forall i; \frac{\partial F}{\partial x_i} \geq 0$.
- $f(S)$ is submodular $\Leftrightarrow \forall i, j; \frac{\partial^2 F}{\partial x_i \partial x_j} \leq 0$.
- $F(x)$ can be evaluated probabilistically to an arbitrary precision.
- *Pipage rounding* [Calinescu, Chekuri, Pál, V. '07]:
  For a submodular $f(S)$ and a matroid $\mathcal{M}$, any $x \in P(\mathcal{M})$ can be rounded to a vertex $S \in \mathcal{M}$ such that $f(S) \geq F(x)$.

Properties of the multilinear relaxation

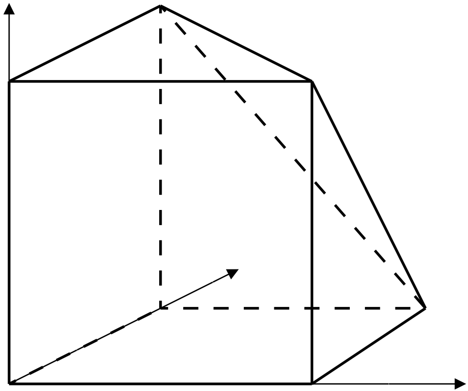
$F(x) = \mathbb{E}[f(\hat{x})]$ has the following properties:

- $f(S)$ is monotone $\Leftrightarrow \forall i; \frac{\partial F}{\partial x_i} \geq 0$.
- $f(S)$ is submodular $\Leftrightarrow \forall i, j; \frac{\partial^2 F}{\partial x_i \partial x_j} \leq 0$.
- $F(x)$ can be evaluated probabilistically to an arbitrary precision.
- *Pipage rounding* [Calinescu, Chekuri, Pál, V. '07]:
  For a submodular $f(S)$ and a matroid $\mathcal{M}$, any $x \in P(\mathcal{M})$ can be rounded to a vertex $S \in \mathcal{M}$ such that $f(S) \geq F(x)$.

However: $\max\{F(x) : x \in P(\mathcal{F})\}$ is not a concave optimization problem, and it cannot be solved optimally in general.

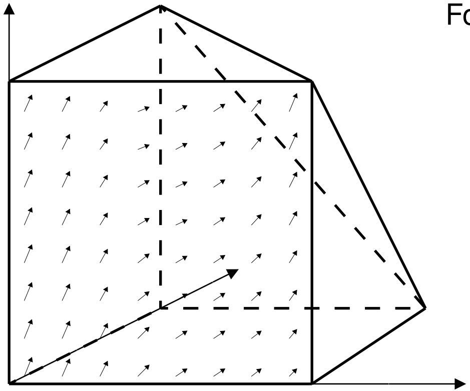
Solving the multilinear relaxation approximately

Problem: $\max\{F(y) : y \in P\}$.
Special case: $F(y)$ monotone, P matroid polytope.



Solving the multilinear relaxation approximately

Problem: $\max\{F(y) : y \in P\}$.
Special case: $F(y)$ monotone, P matroid polytope.

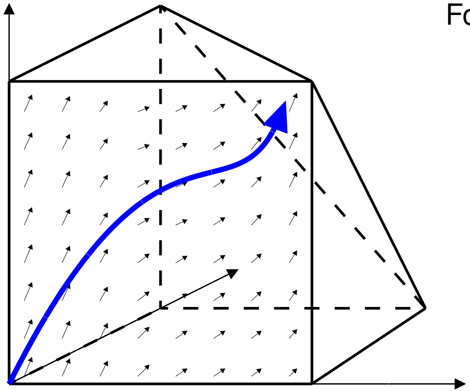


For each $y \in P$, define $v(y)$ by

$$v(y) = \operatorname{argmax}_{v \in P}(v \cdot \nabla F).$$

Solving the multilinear relaxation approximately

Problem: $\max\{F(y) : y \in P\}$.
Special case: $F(y)$ monotone, P matroid polytope.



For each $y \in P$, define $v(y)$ by

$$v(y) = \text{argmax}_{v \in P}(v \cdot \nabla F).$$

Define a curve $y(t)$:

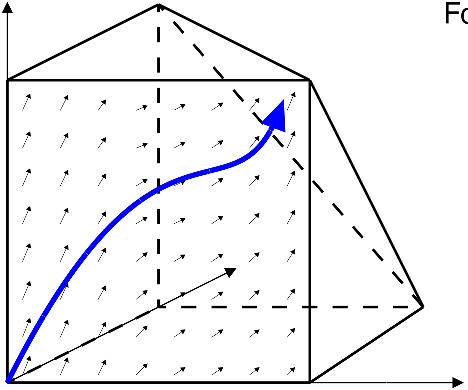
$$y(0) = 0$$

$$\frac{dy}{dt} = v(y)$$

Run this process
for $t \in [0, 1]$ and return $y(1)$.

Solving the multilinear relaxation approximately

Problem: $\max\{F(y) : y \in P\}$.
Special case: $F(y)$ monotone, P matroid polytope.



For each $y \in P$, define $v(y)$ by

$$v(y) = \mathrm{argmax}_{v \in P}(v \cdot \nabla F).$$

Define a curve $y(t)$:

$$y(0) = 0$$

$$\frac{dy}{dt} = v(y)$$

Run this process for $t \in [0, 1]$ and return $y(1)$.

Lemma: $y(1) \in P$ and $F(y(1)) \geq (1 - 1/e)OPT$.

New results [V. '09]

Submodular max over matroid bases: $\max\{f(S) : S \in \mathcal{B}\}$
where $f(S)$ is (non-monotone) submodular and $\mathcal{B}$ the bases of $\mathcal{M}$.
Known: $1/6$-approximation if $\mathcal{M}$ has 2 disjoint bases [LMNS '09].

New results [V. '09]

Submodular max over matroid bases: $\max\{f(S) : S \in \mathcal{B}\}$
where $f(S)$ is (non-monotone) submodular and $\mathcal{B}$ the bases of $\mathcal{M}$.
Known: $1/6$-approximation if $\mathcal{M}$ has 2 disjoint bases [LMNS '09].

Our results:

- $\frac{1}{2}(1 - 1/\nu)$-approximation for $\max\{f(S) : S \in \mathcal{B}\}$, assuming the fractional base packing number is $\nu \in [1, 2]$.
- $(1 - 1/\nu + \epsilon)$-approximation for the same problem would require exponentially many value queries.

New results [V. '09]

Submodular max over matroid bases: $\max\{f(S) : S \in \mathcal{B}\}$
where $f(S)$ is (non-monotone) submodular and $\mathcal{B}$ the bases of $\mathcal{M}$.
Known: $1/6$-approximation if $\mathcal{M}$ has 2 disjoint bases [LMNS '09].

Our results:

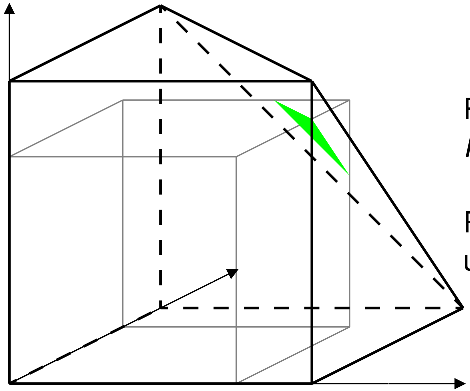
- $\frac{1}{2}(1 - 1/\nu)$-approximation for $\max\{f(S) : S \in \mathcal{B}\}$, assuming the fractional base packing number is $\nu \in [1, 2]$.
- $(1 - 1/\nu + \epsilon)$-approximation for the same problem would require exponentially many value queries.

Hardness ... a special case of a more general result:

Symmetry gap $\longrightarrow$ hardness of approximation.

Algorithm for $\max\{f(S) : S \in \mathcal{B}\}$

Local search in $P(\mathcal{B})$:
move in the direction of $\mathbf{e}_i - \mathbf{e}_j$, if possible and $\frac{\partial F}{\partial x_i} - \frac{\partial F}{\partial x_j} \geq 0$.



Restrict the local search to $P(\mathcal{B}) \cap [0, t]^n$ where $t = 1/\nu$.

Round the fractional solution using pipage rounding.

Claim: For any local maximum x, $F(x) \geq \frac{1}{2}(1 - t)OPT$.

The symmetry gap

Symmetric instance: $\max\{f(S) : S \in \mathcal{F}\}$ on a ground set X is symmetric under a group of permutations $\mathcal{G} \subset \mathbf{S}(X)$, if for any $\sigma \in \mathcal{G}$,

- $f(S) = f(\sigma(S))$
- $S \in \mathcal{F} \Leftrightarrow \sigma(S) \in \mathcal{F}$

The symmetry gap

Symmetric instance: $\max\{f(S) : S \in \mathcal{F}\}$ on a ground set X is symmetric under a group of permutations $\mathcal{G} \subset \mathbf{S}(X)$, if for any $\sigma \in \mathcal{G}$,

- $f(S) = f(\sigma(S))$
- $S \in \mathcal{F} \Leftrightarrow \sigma(S) \in \mathcal{F}$

Symmetrization: For any $x \in [0,1]^X$, we define $\bar{x} = \mathbb{E}_{\sigma \in \mathcal{G}}[\sigma(x)]$.

The symmetry gap

Symmetric instance: $\max\{f(S) : S \in \mathcal{F}\}$ on a ground set X is symmetric under a group of permutations $\mathcal{G} \subset \mathbf{S}(X)$, if for any $\sigma \in \mathcal{G}$,

- $f(S) = f(\sigma(S))$
- $S \in \mathcal{F} \Leftrightarrow \sigma(S) \in \mathcal{F}$

Symmetrization: For any $x \in [0,1]^X$, we define $\bar{x} = \mathbb{E}_{\sigma \in \mathcal{G}}[\sigma(x)]$.

Symmetry gap:

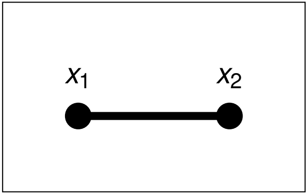
$$\gamma = \frac{\overline{OPT}}{OPT}$$

where

$$OPT = \max\{F(x) : x \in P(\mathcal{F})\}$$

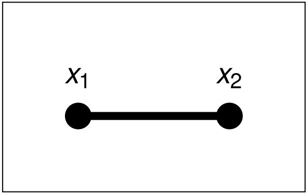
$$\overline{OPT} = \max\{F(\bar{x}) : x \in P(\mathcal{F})\}.$$

Example: Max Cut on K_2



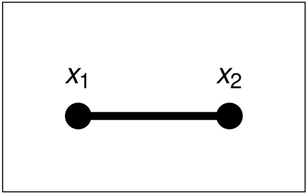
- $X = \{1, 2\}$, $\mathcal{F} = 2^X$, $P(\mathcal{F}) = [0, 1]^2$.
- $f(S) = 1$ if $|S| = 1$, otherwise 0.
- $F(x) = x_1(1 - x_2) + (1 - x_1)x_2$

Example: Max Cut on K_2



- $X = \{1, 2\}$, $\mathcal{F} = 2^X$, $P(\mathcal{F}) = [0, 1]^2$.
- $f(S) = 1$ if $|S| = 1$, otherwise 0.
- $F(x) = x_1(1 - x_2) + (1 - x_1)x_2$
- Symmetric under $\mathcal{G} = \mathbf{S}_2$, all permutations of 2 elements.
- For $x = (x_1, x_2)$, $\bar{x} = (\frac{x_1+x_2}{2}, \frac{x_1+x_2}{2})$.

Example: Max Cut on K_2



- $X = \{1, 2\}$, $\mathcal{F} = 2^X$, $P(\mathcal{F}) = [0, 1]^2$.
- $f(S) = 1$ if $|S| = 1$, otherwise 0.
- $F(x) = x_1(1 - x_2) + (1 - x_1)x_2$
- Symmetric under $\mathcal{G} = \mathbf{S}_2$, all permutations of 2 elements.
- For $x = (x_1, x_2)$, $\bar{x} = (\frac{x_1+x_2}{2}, \frac{x_1+x_2}{2})$.
- $OPT = \max\{F(x) : x \in P(\mathcal{F})\} = F(1, 0) = 1$.
- $\overline{OPT} = \max\{F(\bar{x}) : x \in P(\mathcal{F})\} = F(1/2, 1/2) = 1/2$.

Symmetry gap $\Rightarrow$ hardness

Assume: $\max\{f(S) : S \in \mathcal{F}\}$ is an instance with symmetry gap γ.

Refinement of $\mathcal{F}$: make n copies of each element of X, and define a family $\mathcal{F}'$ of subsets of $X \times [n]$ so that

- $S \in \mathcal{F}'$ iff $x^S \in P(\mathcal{F})$ where
  x_i^S is the fraction of copies of element i that S contains.

Symmetry gap $\Rightarrow$ hardness

Assume: $\max\{f(S) : S \in \mathcal{F}\}$ is an instance with symmetry gap γ.

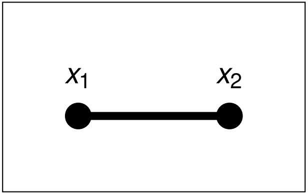
Refinement of $\mathcal{F}$: make n copies of each element of X, and define a family $\mathcal{F}'$ of subsets of $X \times [n]$ so that

- $S \in \mathcal{F}'$ iff $x^S \in P(\mathcal{F})$ where
  x_i^S is the fraction of copies of element i that S contains.

Theorem

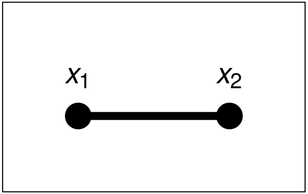
For any instance of submodular maximization with symmetry gap γ, and any $\epsilon > 0$, any $(\gamma + \epsilon)$-approximation for the class of instances $\max\{f'(S) : S \in \mathcal{F}'\}$ where f' is submodular and $\mathcal{F}'$ is a refinement of $\mathcal{F}$, would require exponentially many value queries.

Application 1: general submodular maximization



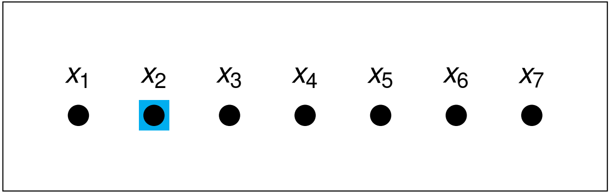
- $\max\{f(S) : S \subseteq \{1,2\}\}$: symmetric under $\mathbf{S}_2$.
- Symmetry gap is $\gamma = 1/2$.
- Refined instances are instances of unconstrained (non-monotone) submodular maximization.

Application 1: general submodular maximization



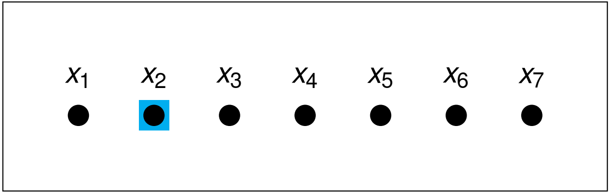
- $\max\{f(S) : S \subseteq \{1,2\}\}$: symmetric under $\mathbf{S}_2$.
- Symmetry gap is $\gamma = 1/2$.
- Refined instances are instances of unconstrained (non-monotone) submodular maximization.
- Theorem implies that a better than $1/2$-approximation is impossible (previously known [FMV '07]).

Application 2: monotone submodular maximization



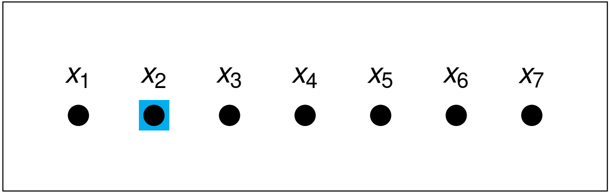
- $X = [k]$, $\mathcal{F} = \{S \subseteq [k] : |S| \leq 1\}$.
- $f(S) = \min\{|S|, 1\}$, symmetric under $\mathbf{S}_k$.

Application 2: monotone submodular maximization



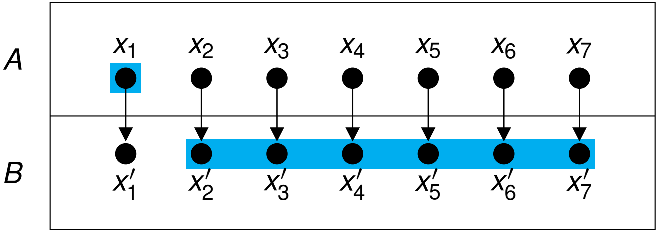
- $X = [k]$, $\mathcal{F} = \{S \subseteq [k] : |S| \leq 1\}$.
- $f(S) = \min\{|S|, 1\}$, symmetric under $\mathbf{S}_k$.
- $OPT = F(1, 0, \ldots, 0) = 1$.
- $\overline{OPT} = F(\frac{1}{k}, \frac{1}{k}, \ldots, \frac{1}{k}) = 1 - (1 - \frac{1}{k})^k$.

Application 2: monotone submodular maximization



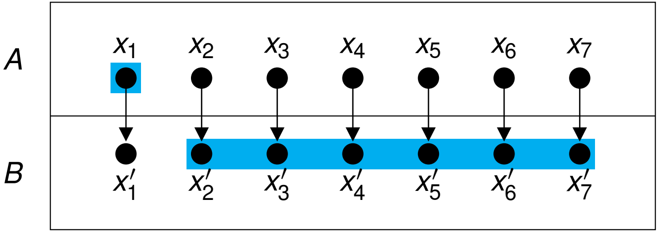
- $X = [k]$, $\mathcal{F} = \{S \subseteq [k] : |S| \leq 1\}$.
- $f(S) = \min\{|S|, 1\}$, symmetric under $\mathbf{S}_k$.
- $OPT = F(1, 0, \ldots, 0) = 1$.
- $\overline{OPT} = F(\frac{1}{k}, \frac{1}{k}, \ldots, \frac{1}{k}) = 1 - (1 - \frac{1}{k})^k$.
- Refined instances: monotone submodular maximization under a cardinality constraint.
- Theorem implies that a better than $1 - (1 - \frac{1}{k})^k$-approximation is impossible (previously known [NW '78]).

Application 3: non-monotone submodular over bases



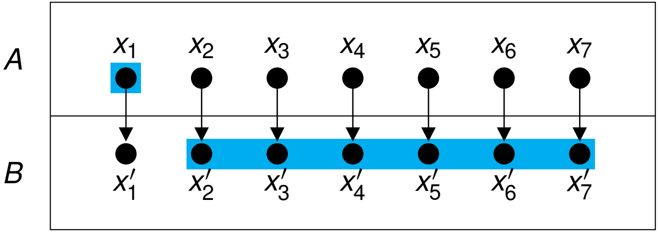
- $X = A \cup B$, $|A| = |B| = k$,
  $\mathcal{F} = \{S \subseteq X : |S \cap A| = 1, |S \cap B| = k - 1\}$.
- $f(S) =$ number of arcs leaving S; symmetric under $\mathbf{S}_k$.

Application 3: non-monotone submodular over bases



- $X = A \cup B$, $|A| = |B| = k$,
  $\mathcal{F} = \{S \subseteq X : |S \cap A| = 1, |S \cap B| = k-1\}$.
- $f(S)$ = number of arcs leaving S; symmetric under $\mathbf{S}_k$.
- $OPT = F(1, 0, \ldots, 0; 0, 1, \ldots, 1) = 1$.
- $\overline{OPT} = F(\frac{1}{k}, \ldots, \frac{1}{k}; 1 - \frac{1}{k}, \ldots, 1 - \frac{1}{k}) = \frac{1}{k}$.

Application 3: non-monotone submodular over bases



- $X = A \cup B$, $|A| = |B| = k$, $\mathcal{F} = \{S \subseteq X : |S \cap A| = 1, |S \cap B| = k - 1\}$.
- $f(S)$ = number of arcs leaving S; symmetric under $\mathbf{S}_k$.
- $OPT = F(1, 0, \ldots, 0; 0, 1, \ldots, 1) = 1$.
- $\overline{OPT} = F(\frac{1}{k}, \ldots, \frac{1}{k}; 1 - \frac{1}{k}, \ldots, 1 - \frac{1}{k}) = \frac{1}{k}$.
- Refined instances: non-monotone submodular maximization over matroid bases, with base packing number $\nu = k/(k-1)$.
- Theorem implies that a better than $\frac{1}{k}$-approximation is impossible.

Symmetry -> Hardness: the proof

Main ideas: (following [FMV '07, MSV '08])

1. Refined instances can be viewed as continuous optimization problems.
2. A continuous instance can be tweaked so that it looks like a symmetric instance to any algorithm.

Symmetry -> Hardness: the proof

Main ideas: (following [FMV '07, MSV '08])

1. Refined instances can be viewed as continuous optimization problems.
2. A continuous instance can be tweaked so that it looks like a symmetric instance to any algorithm.

Why is it possible? Let $G(x) = F(\bar{x})$, $\bar{x} = \mathbb{E}_{\sigma \in \mathcal{G}}[\sigma(x)]$.

- $OPT = \max\{F(x) : x \in P(\mathcal{F})\}$
- $\overline{OPT} = \max\{G(x) : x \in P(\mathcal{F})\}$.

Symmetry -> Hardness: the proof

Main ideas: (following [FMV '07, MSV '08])

1. Refined instances can be viewed as continuous optimization problems.
2. A continuous instance can be tweaked so that it looks like a symmetric instance to any algorithm.

Why is it possible? Let $G(x) = F(\bar{x})$, $\bar{x} = \mathbb{E}_{\sigma \in \mathcal{G}}[\sigma(x)]$.

- $OPT = \max\{F(x) : x \in P(\mathcal{F})\}$
- $\overline{OPT} = \max\{G(x) : x \in P(\mathcal{F})\}$.

Lemma: For any "symmetric point", $x = \bar{x}$,

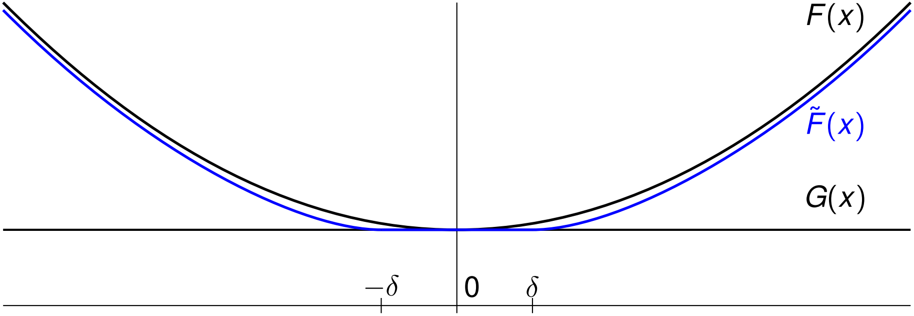
$$\nabla F(x) = \nabla G(x).$$

I.e., the two instances look very similar in the "symmetric region".

Main technical lemma

Given $F(x)$ and its symmetrized version $G(x) = F(\bar{x})$, for any $\epsilon > 0$ there is $\delta > 0$ and a function $\tilde{F}(x)$ such that

- Whenever $||x - \bar{x}|| < \delta$, $\tilde{F}(x) = G(x)$ (*symmetric region*).
- For all x, $|\tilde{F}(x) - F(x)| < \epsilon$.
- For all i, j, $\frac{\partial^2 \tilde{F}}{\partial x_i \partial x_j} \leq 0$.

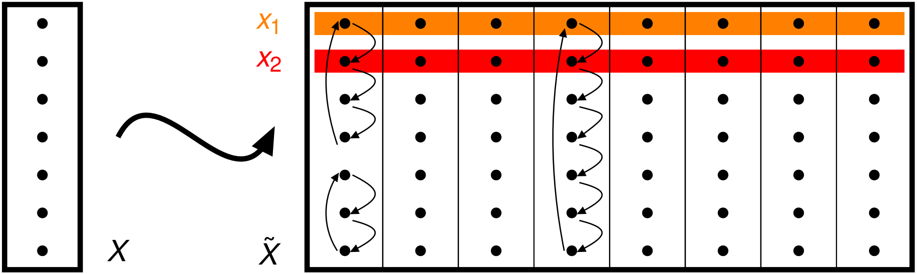


Final construction

Yao's principle: to prove a lower bound for randomized algorithms, consider a random input that fools any deterministic algorithm.

Final construction

Yao's principle: to prove a lower bound for randomized algorithms, consider a random input that fools any deterministic algorithm.



- π = random shuffle of each copy of X by an indep. $\sigma \in \mathcal{G}$.
- Given $S \subseteq \tilde{X}$, let x_i = fraction of copies of i in $\pi(S)$.
- We define $\tilde{f}(S) = \tilde{F}(x)$.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.
2. With high probability, any query is in the *symmetric region*.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.
2. With high probability, any query is in the *symmetric region*.
3. Assuming this happens, the answers are independent of π and the algorithm follows a fixed path of computation $\mathcal{P}$.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.
2. With high probability, any query is in the *symmetric region*.
3. Assuming this happens, the answers are independent of π and the algorithm follows a fixed path of computation $\mathcal{P}$.
4. If $\mathcal{P}$ contains subexponentially many queries, they are all in the *symmetric region* w.h.p.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.
2. With high probability, any query is in the *symmetric region*.
3. Assuming this happens, the answers are independent of π and the algorithm follows a fixed path of computation $\mathcal{P}$.
4. If $\mathcal{P}$ contains subexponentially many queries, they are all in the *symmetric region* w.h.p.
5. Hence, the algorithm follows $\mathcal{P}$ w.h.p. and gives an answer independent of π.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.
2. With high probability, any query is in the *symmetric region*.
3. Assuming this happens, the answers are independent of π and the algorithm follows a fixed path of computation $\mathcal{P}$.
4. If $\mathcal{P}$ contains subexponentially many queries, they are all in the *symmetric region* w.h.p.
5. Hence, the algorithm follows $\mathcal{P}$ w.h.p. and gives an answer independent of π.
6. This answer is itself symmetric and hence no better than $\overline{OPT}$.

Finishing the proof

1. $\max\{\tilde{f}(S) : S \in \mathcal{F}'\}$ is a refinement of the original instance.
2. With high probability, any query is in the *symmetric region*.
3. Assuming this happens, the answers are independent of π and the algorithm follows a fixed path of computation $\mathcal{P}$.
4. If $\mathcal{P}$ contains subexponentially many queries, they are all in the *symmetric region* w.h.p.
5. Hence, the algorithm follows $\mathcal{P}$ w.h.p. and gives an answer independent of π.
6. This answer is itself symmetric and hence no better than $\overline{OPT}$.
7. The actual optimum is at least $OPT - \epsilon$.

Open questions

- What other hardness results can be derived from symmetry gap examples?

Open questions

- What other hardness results can be derived from symmetry gap examples?
- Is it possible the optimal approximation factor is always matched by a symmetry gap? (say, for matroid-type constraints)

Open questions

- What other hardness results can be derived from symmetry gap examples?
- Is it possible the optimal approximation factor is always matched by a symmetry gap? (say, for matroid-type constraints)
- Can the query-complexity results be replicated as NP-hardness results?

Open questions

- What other hardness results can be derived from symmetry gap examples?
- Is it possible the optimal approximation factor is always matched by a symmetry gap? (say, for matroid-type constraints)
- Can the query-complexity results be replicated as NP-hardness results?
- Specific case - unconstrained submodular maximization:
  1. Can we achieve $1/2$-approximation?
  2. Is there a concrete special case where $(1/2 + \epsilon)$-approximation is NP-hard (only $3/4 + \epsilon$ known [FMV '07])