Dependent Randomized Rounding via Exchange Properties of Combinatorial Structures

Rico Zenklusen

MIT

Joint work with Chandra Chekuri and Jan Vondrák

Outline

Outline

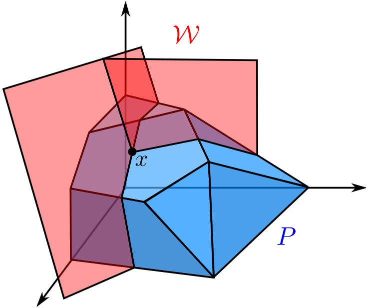
Randomized rounding

A technique to profit from relaxations of hard problems

A typical setting

$$\begin{aligned} \max / \min \quad & f(x) \\ & x \in P \\ & x \in \mathcal{W} \\ & x \in \{0,1\}^n \end{aligned}$$

- $P \subset [0,1]^n$: integer polytope representing "hard" constraints.
- $\mathcal{W}$: "weak" constraints.



The strategy

Randomly round a fractional solution x of the relaxation to $X \in \{0,1\}^n$ so that:

- X satisfies hard constraints: $X \in P$,
- X is good in expectation: $\mathbf{E}[X] \approx x$,
- linear (and possibly other) functions $g(X)$ concentrates around $\mathbf{E}[g(X)]$.
  ➙ Chernoff-type bounds $\Rightarrow g(X) \approx g(x)$, and X is almost in $\mathcal{W}$ whp.

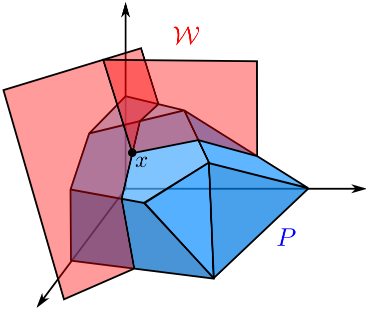
Randomized rounding

A technique to profit from relaxations of hard problems

A typical setting

$$\begin{aligned} \max/\min \quad & f(x) \\ & x \in P \\ & x \in \mathcal{W} \\ & x \in \{0,1\}^n \end{aligned}$$

- $P \subset [0,1]^n$: integer polytope representing "hard" constraints.
- $\mathcal{W}$: "weak" constraints.



The strategy

Randomly round a fractional solution x of the relaxation to $X \in \{0,1\}^n$ so that:

- X satisfies hard constraints: $X \in P$,
- X is good in expectation: $\mathbf{E}[X] \approx x$,
- linear (and possibly other) functions $g(X)$ concentrates around $\mathbf{E}[g(X)]$.
  ➙ Chernoff-type bounds $\Rightarrow g(X) \approx g(x)$, and X is almost in $\mathcal{W}$ whp.

Dependent rounding and negative correlations

Independent randomized rounding (Raghavan and Thompson [1987])

- $\Pr[X_i = 1] = x_i$, (almost) independently for $i \in [n] := 1, \ldots, n$.

✓ Linear functions $g(X)$ satisfy Chernoff-type concentration bounds.

☛ Polytope P has to be very simple for this to work.

Dependent randomized rounding

- Typically, a rounding procedure tailored to P is needed to ensure feasibility.
  $\Rightarrow$ Dependencies between different components of X are created.
- Still, Chernoff-type concentration bounds are desired.
  $\rightarrow$ They often follow from negative correlation.

Dependent rounding and negative correlations

Independent randomized rounding (Raghavan and Thompson [1987])

- $\Pr[X_i = 1] = x_i$, (almost) independently for $i \in [n] := 1, \ldots, n$.

✓ Linear functions $g(X)$ satisfy Chernoff-type concentration bounds.

☛ Polytope P has to be very simple for this to work.

Dependent randomized rounding

- Typically, a rounding procedure tailored to P is needed to ensure feasibility.
  $\Rightarrow$ Dependencies between different components of X are created.
- Still, Chernoff-type concentration bounds are desired.
  $\rightarrow$ They often follow from negative correlation.

Concentration through negative correlation

Obtaining Chernoff bounds without independence

Theorem (Panconesi and Srinivasan [1997])

Let $X \in \{0,1\}^n$ *be a random vector with* $\mathbf{E}[X] = x$*. If for any* $S \subseteq [n]$

- $\Pr[\bigwedge_{i \in S}(X_i = 1)] \leq \prod_{i \in S} x_i$,
- $\Pr[\bigwedge_{i \in S}(X_i = 0)] \leq \prod_{i \in S}(1 - x_i)$,

} *negative correlation*

then for $a \in [0,1]^n$,

- $\Pr\left[a^T X \geq \mu(1+\delta)\right] \leq \left(\frac{e^\delta}{(1+\delta)^{1+\delta}}\right)^\mu$ *for* $\delta \geq 0, \mu \geq \mathbf{E}[a^T X]$
- $\Pr\left[a^T X \leq \mu(1-\delta)\right] \leq e^{-\mu\delta^2/2}$ *for* $\delta \in [0,1], \mu = \mathbf{E}[a^T X]$

Recipe for creating dependent randomized rounding procedures

Round given point $x \in P$ to random integral vector $X \in P$ such that:

- $\mathbf{E}[X] = x$,
- Coordinates of X are negatively correlated.

Concentration through negative correlation

Obtaining Chernoff bounds without independence

Theorem (Panconesi and Srinivasan [1997])

Let $X \in \{0,1\}^n$ be a random vector with $\mathbf{E}[X] = x$. If for any $S \subseteq [n]$

- $\Pr[\bigwedge_{i \in S}(X_i = 1)] \le \prod_{i \in S} x_i$,
- $\Pr[\bigwedge_{i \in S}(X_i = 0)] \le \prod_{i \in S}(1 - x_i)$,

} *negative correlation*

then for $a \in [0,1]^n$,

- $\Pr\left[a^T X \ge \mu(1+\delta)\right] \le \left(\frac{e^\delta}{(1+\delta)^{1+\delta}}\right)^\mu$ *for $\delta \ge 0, \mu \ge \mathbf{E}[a^T X]$*
- $\Pr\left[a^T X \le \mu(1-\delta)\right] \le e^{-\mu\delta^2/2}$ *for $\delta \in [0,1], \mu = \mathbf{E}[a^T X]$*

Recipe for creating dependent randomized rounding procedures

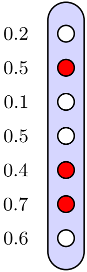
Round given point $x \in P$ to random integral vector $X \in P$ such that:

- $\mathbf{E}[X] = x$,
- Coordinates of X are negatively correlated.

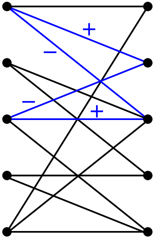
Examples of this approach

Rounding procedures with $\mathbf{E}[X] = x$, and negative correlation

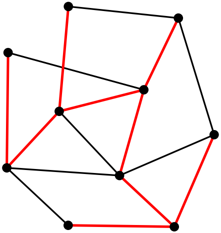
- $P = \{x \in [0,1]^n \mid \sum_{i=1}^n x_i = k\}$ (Srinivasan [2001])
- Assignment polytope; negative correlation only for edges adjacent to any fixed vertex (Gandhi et al. [2006]).
- Spanning tree polytope (Asadpour et al. [2010])
  $\rightarrow$ get thin spanning tree $\Rightarrow$ $O(\log n / \log \log n)$-approximation to ATSP



Recursive approach using pairing trees.



Modify fractional cycles.



Maximum entropy sampling.

Motivating questions and main results

- Which polytopes admit negatively correlated rounding procedures?
- Unifying framework?
- Concentration for non-linear/submodular functions?

We suggest a new rounding technique (randomized swap rounding)

1. For matroid polytopes:
   - $\mathbf{E}[X] = x$, and negative correlation holds,
   - lower-tail concentration bound for monotone submodular functions (using martingale argument).
2. For the intersection of two matroids:
   - $\mathbf{E}[X] = x$, and negative correlation for "equivalent elements" (generalization of stated result on assignment polytope).

- Polytopes admitting negatively correlated rounding procedures are exactly axis-parallel projections of base polytopes of matroids.

Motivating questions and main results

- Which polytopes admit negatively correlated rounding procedures?
- Unifying framework?
- Concentration for non-linear/submodular functions?

We suggest a new rounding technique (randomized swap rounding)

1. For matroid polytopes:
   - $\mathbf{E}[X] = x$, and negative correlation holds,
   - lower-tail concentration bound for monotone submodular functions (using martingale argument).
2. For the intersection of two matroids:
   - $\mathbf{E}[X] = x$, and negative correlation for "equivalent elements" (generalization of stated result on assignment polytope).

- Polytopes admitting negatively correlated rounding procedures are exactly axis-parallel projections of base polytopes of matroids.

Outline

1. Introduction
   - Motivation
2. Randomized swap rounding: a new rounding framework
   - The general framework
   - Swap rounding in matroid polytopes
   - Swap rounding in the intersection of two matroids
3. Some consequences/applications
4. Conclusions

General rounding framework

Some terminology (to highlight underlying combinatorial problem):
S: finite ground set, $\mathcal{I} \subseteq 2^S$: solution set $\rightarrow$ $P = conv(\{\mathbf{1}_I \mid I \in \mathcal{I}\})$

1. Compute convex decomposition of $x = \sum_{i=1}^{m} \beta_i \mathbf{1}_{I_i}$, with $I_1, \ldots, I_m \in \mathcal{I}$.
2. We iteratively merge the sets $I_1, \ldots, I_m$ to a single set $R \in \mathcal{I}$.

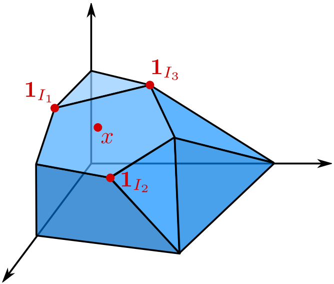
$$
\begin{aligned}
x_1 &= \underbrace{\beta_1 \mathbf{1}_{I_1} + \beta_2 \mathbf{1}_{I_2}} + \beta_3 \mathbf{1}_{I_3} + \ldots \beta_m \mathbf{1}_{I_m} \\
x_2 &= \underbrace{(\beta_1 + \beta_2)\mathbf{1}_{I_{1:2}} + \beta_3 \mathbf{1}_{I_3}} + \ldots \beta_m \mathbf{1}_{I_m} \\
x_3 &= (\beta_1 + \beta_2 + \beta_3)\mathbf{1}_{I_{1:3}} + \ldots \beta_m \mathbf{1}_{I_m} \\
&\vdots \\
x_m &= (\beta_1 + \cdots + \beta_m)\mathbf{1}_{I_{1:m}} = \mathbf{1}_{I_{1:m}}
\end{aligned}
$$

$I_{1:2} = Merge(\beta_1, I_1, \beta_2, I_2)$
$I_{1:3} = Merge(\beta_1 + \beta_2, I_{1:2}, \beta_3, I_3)$

General rounding framework

Some terminology (to highlight underlying combinatorial problem):
S: finite ground set, $\mathcal{I} \subseteq 2^S$: solution set $\rightarrow$ $P = conv(\{\mathbf{1}_I \mid I \in \mathcal{I}\})$

1. Compute convex decomposition of $x = \sum_{i=1}^{m} \beta_i \mathbf{1}_{I_i}$, with $I_1, \ldots, I_m \in \mathcal{I}$.
2. We iteratively merge the sets $I_1, \ldots, I_m$ to a single set $R \in \mathcal{I}$.



$$
\begin{aligned}
x_1 &= \underbrace{\beta_1 \mathbf{1}_{I_1} + \beta_2 \mathbf{1}_{I_2}} + \beta_3 \mathbf{1}_{I_3} + \ldots \beta_m \mathbf{1}_{I_m} \\
x_2 &= \underbrace{(\beta_1 + \beta_2)\mathbf{1}_{I_{1:2}} + \beta_3 \mathbf{1}_{I_3}} + \ldots \beta_m \mathbf{1}_{I_m} \\
x_3 &= (\beta_1 + \beta_2 + \beta_3)\mathbf{1}_{I_{1:3}} + \ldots \beta_m \mathbf{1}_{I_m} \\
&\vdots \\
x_m &= (\beta_1 + \cdots + \beta_m)\mathbf{1}_{I_{1:m}} = \mathbf{1}_{I_{1:m}}
\end{aligned}
$$

$I_{1:2} = Merge(\beta_1, I_1, \beta_2, I_2)$
$I_{1:3} = Merge(\beta_1 + \beta_2, I_{1:2}, \beta_3, I_3)$

Matroids

Definition: matroid $M = (S, \mathcal{I})$

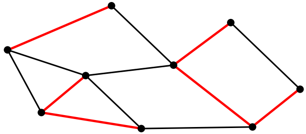
S: finite ground set, $\emptyset \subsetneq \mathcal{I} \subseteq 2^S$: *independent sets* satisfying

- $\forall I \in \mathcal{I}, J \subseteq I \Rightarrow J \in \mathcal{I}$,
- $\forall I, J \in \mathcal{I}, |I| > |J| \Rightarrow \exists i \in I \setminus J$ with $J \cup \{i\} \in \mathcal{I}$.

The set of bases $\mathcal{B}$ are all maximal independent sets.

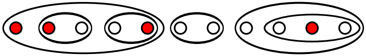
Example: graphic matroid $M = (E, \mathcal{I})$

- $G = (V, E)$: undirected graph
- $\mathcal{I} = \{F \subseteq E \mid F \text{ is a forest}\}$



Example: laminar matroid $M = (S, \mathcal{I})$

- $\mathcal{I} = \{I \subseteq S \mid |I \cap L_i| \leq k_i \ \forall i \in [m]\}$, where $L_1, \ldots, L_m \subseteq S$ is laminar.



Strong exchange property

$\forall B_1, B_2 \in \mathcal{B}, i \in B_1 \Rightarrow \exists j \in B_2$ with $B_1 - i + j \in \mathcal{B}$ and $B_2 - j + i \in \mathcal{B}$.

Matroids

Definition: matroid $M = (S, \mathcal{I})$

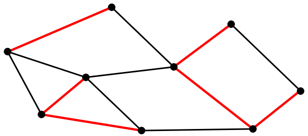
S: finite ground set, $\emptyset \subsetneq \mathcal{I} \subseteq 2^S$: *independent sets* satisfying

- $\forall I \in \mathcal{I},\ J \subseteq I \Rightarrow J \in \mathcal{I}$,
- $\forall I, J \in \mathcal{I}, |I| > |J| \Rightarrow \exists i \in I \setminus J$ with $J \cup \{i\} \in \mathcal{I}$.

The set of bases $\mathcal{B}$ are all maximal independent sets.

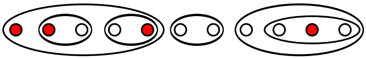
Example: graphic matroid $M = (E, \mathcal{I})$

- $G = (V, E)$: undirected graph
- $\mathcal{I} = \{F \subseteq E \mid F \text{ is a forest}\}$



Example: laminar matroid $M = (S, \mathcal{I})$

- $\mathcal{I} = \{I \subseteq S \mid |I \cap L_i| \leq k_i\ \forall i \in [m]\}$, where $L_1, \ldots, L_m \subseteq S$ is laminar.



Strong exchange property

$\forall B_1, B_2 \in \mathcal{B},\ i \in B_1 \Rightarrow \exists j \in B_2$ with $B_1 - i + j \in \mathcal{B}$ and $B_2 - j + i \in \mathcal{B}$.

Merging for matroid polytopes

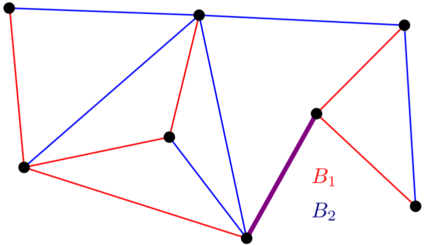
Algorithm Merge($\beta_1, B_1, \beta_2, B_2$)

While $(B_1 \neq B_2)$ do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.

Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

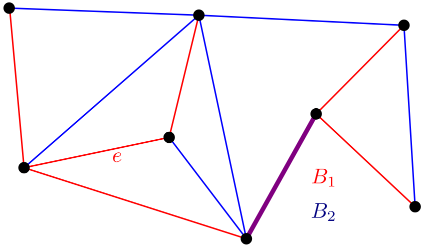
While $(B_1 \neq B_2)$ do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge($\beta_1, B_1, \beta_2, B_2$)

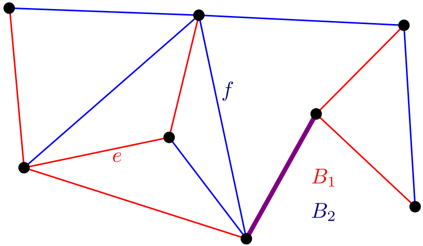
While ($B_1 \neq B_2$) do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

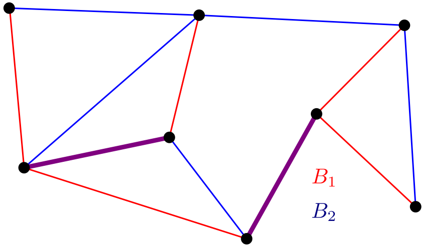
While $(B_1 \neq B_2)$ do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

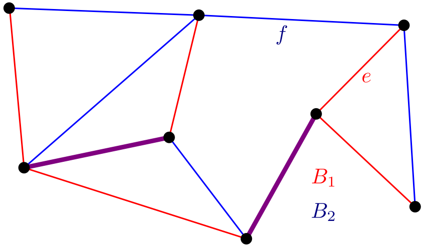
While $(B_1 \neq B_2)$ do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge($\beta_1, B_1, \beta_2, B_2$)

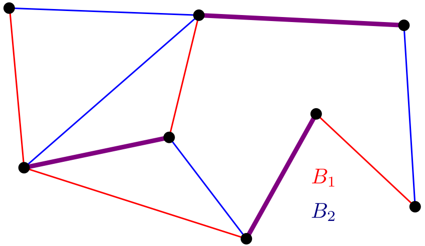
While ($B_1 \neq B_2$) do
Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
$B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge($\beta_1, B_1, \beta_2, B_2$)

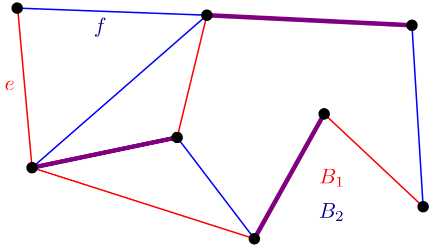
While ($B_1 \neq B_2$) do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

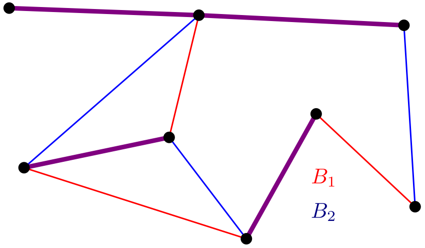
While $(B_1 \neq B_2)$ do
Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
$B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

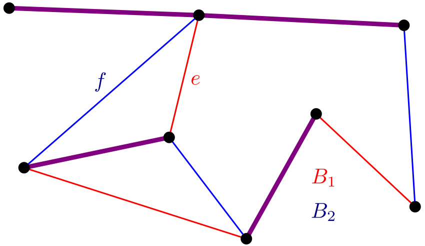
While $(B_1 \neq B_2)$ do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

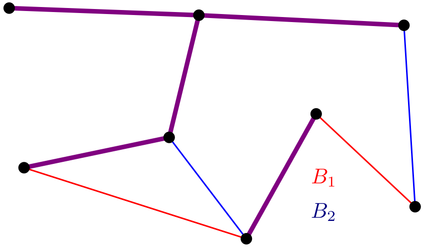
While $(B_1 \neq B_2)$ do
Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
$B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

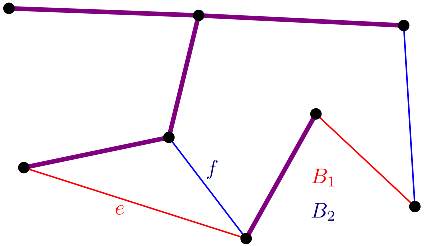
While $(B_1 \neq B_2)$ do
Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
$B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge($\beta_1, B_1, \beta_2, B_2$)

While ($B_1 \neq B_2$) do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

While $(B_1 \neq B_2)$ do

Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that

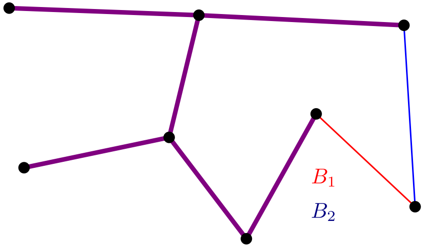
$B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;

With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;

Else $\{B_1 \leftarrow B_1 - e + f\}$;

EndWhile

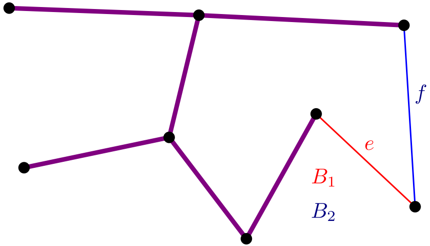
Output B_1.



Merging for matroid polytopes

Algorithm Merge$(\beta_1, B_1, \beta_2, B_2)$

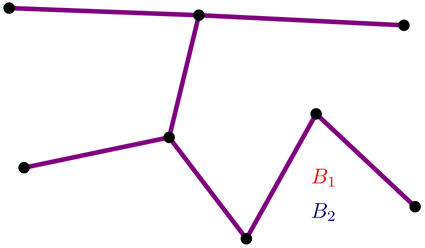
While $(B_1 \neq B_2)$ do
  Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
    $B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
  With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
    Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Merging for matroid polytopes

Algorithm Merge($\beta_1, B_1, \beta_2, B_2$)

While ($B_1 \neq B_2$) do
Pick $e \in B_1 \setminus B_2$ and find $f \in B_2 \setminus B_1$ such that
$B_1 - e + f \in \mathcal{B}$ and $B_2 - f + e \in \mathcal{B}$;
With probability $\beta_1/(\beta_1 + \beta_2)$, $\{B_2 \leftarrow B_2 - f + e\}$;
Else $\{B_1 \leftarrow B_1 - e + f\}$;
EndWhile
Output B_1.



Concentration for linear functions

Lemma

Let $X_t = (X_{1,t}, \ldots, X_{n,t})$ be a non-negative vector-valued random process with initial distribution given by $X_0 = x \in \mathbb{R}^n$ with probability 1 and such that:

- $\mathbf{E}[X_{t+1} \mid X_t] = X_t$,
- between X_t and X_{t+1} at most two components change,
- if two components change, one increases and the other one decreases.

Then for any t, the components of X_t are negatively correlated.

- Above Lemma applies to swap rounding algorithm for matroids.
  $\Rightarrow$ Chernoff bounds hold for linear functions with coefficients in $[0, 1]$.

- We also get lower-tail concentration bounds for monotone submodular functions.
- This does not follow from negative correlation→ martingale approach.

Concentration for linear functions

Lemma

Let $X_t = (X_{1,t}, \ldots, X_{n,t})$ be a non-negative vector-valued random process with initial distribution given by $X_0 = x \in \mathbb{R}^n$ with probability 1 and such that:

- $\mathbf{E}[X_{t+1} \mid X_t] = X_t$,
- between X_t and X_{t+1} at most two components change,
- if two components change, one increases and the other one decreases.

Then for any t, the components of X_t are negatively correlated.

- Above Lemma applies to swap rounding algorithm for matroids.
  $\Rightarrow$ Chernoff bounds hold for linear functions with coefficients in $[0, 1]$.

- We also get lower-tail concentration bounds for monotone submodular functions.
- This does not follow from negative correlation $\rightarrow$ martingale approach.

Concentration for linear functions

Lemma

Let $X_t = (X_{1,t}, \ldots, X_{n,t})$ be a non-negative vector-valued random process with initial distribution given by $X_0 = x \in \mathbb{R}^n$ with probability 1 and such that:

- $\mathbf{E}[X_{t+1} \mid X_t] = X_t$,
- between X_t and X_{t+1} at most two components change,
- if two components change, one increases and the other one decreases.

Then for any t, the components of X_t are negatively correlated.

- Above Lemma applies to swap rounding algorithm for matroids.

  $\Rightarrow$ Chernoff bounds hold for linear functions with coefficients in $[0, 1]$.

- We also get lower-tail concentration bounds for monotone submodular functions.
- This does not follow from negative correlation $\rightarrow$ martingale approach.

Submodular functions

Definition: submodular function

A function $f : 2^S \to \mathbb{R}$ is submodular if it has the property of diminishing returns:

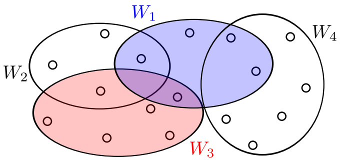
$$f(A+i) - f(A) \geq f(B+i) - f(B) \quad \forall A \subseteq B \subseteq S, i \in S \setminus B.$$

Furthermore, f is monotone if $f(A) \leq f(B)$ $\forall A \subseteq B \subseteq S$.

Example I: coverage function

Let U be a finite ground set and $W_i \subseteq U$ for $i \in S$.

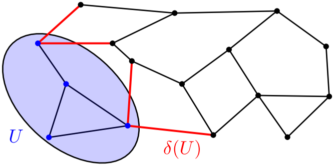
$$f(A) = \left| \bigcup_{i \in A} W_i \right| \quad \forall A \subseteq S$$



Example II: cut function

Given is a graph $G = (V, E)$.

$$f(U) = |\delta(U)| = |E(U, V \setminus U)| \quad \forall U \subseteq V$$



- To work with submodular functions in relaxations, a continuous counterpart known as multilinear extension proved to be very useful.

Submodular functions

Definition: submodular function

A function $f : 2^S \to \mathbb{R}$ is submodular if it has the property of diminishing returns:

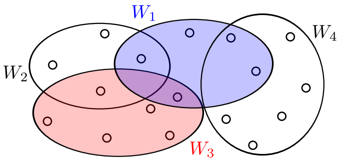
$$f(A+i) - f(A) \geq f(B+i) - f(B) \quad \forall A \subseteq B \subseteq S, i \in S \setminus B.$$

Furthermore, f is monotone if $f(A) \leq f(B)$ $\forall A \subseteq B \subseteq S$.

Example I: coverage function

Let U be a finite ground set and $W_i \subseteq U$ for $i \in S$.

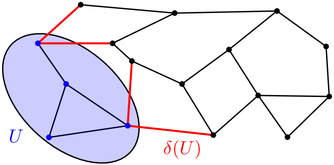
$$f(A) = \left| \bigcup_{i \in A} W_i \right| \quad \forall A \subseteq S$$



Example II: cut function

Given is a graph $G = (V, E)$.

$$f(U) = |\delta(U)| = |E(U, V \setminus U)| \quad \forall U \subseteq V$$



- To work with submodular functions in relaxations, a continuous counterpart known as multilinear extension proved to be very useful.

Multilinear extension of submodular function

Definition: multilinear extension

The multilinear extension F of a submodular function f is defined by:

$$F(x) = \sum_{A \subseteq S} f(A) \prod_{i \in A} x_i \prod_{i \in S \setminus A} (1 - x_i) \quad \forall x \in [0,1]^S.$$

- Hence, $F(x) = \mathbf{E}[f(R)]$ where R is a random set containing each element $i \in S$ independently with probability x_i.

Theorem (Vondrák [2008])

There is a $(1 - 1/e)$-approximation for maximizing F over any 0/1 polytope over which one can optimize efficiently linear functions.

Multilinear extension of submodular function

Definition: multilinear extension

The multilinear extension F of a submodular function f is defined by:

$$F(x) = \sum_{A \subseteq S} f(A) \prod_{i \in A} x_i \prod_{i \in S \setminus A} (1 - x_i) \quad \forall x \in [0,1]^S.$$

- Hence, $F(x) = \mathbf{E}[f(R)]$ where R is a random set containing each element $i \in S$ independently with probability x_i.

Theorem (Vondrák [2008])

There is a $(1 - 1/e)$-approximation for maximizing F over any $0/1$ polytope over which one can optimize efficiently linear functions.

Concentration for submodular functions

Consider the following setting:

- $f : \{0,1\}^n \to \mathbb{R}_+$: monotone submodular function with marginal values ≤ 1,
- $F : [0,1]^n \to \mathbb{R}_+$: multilinear extension of f,
- $x \in P$: point in matroid polytope P to round,
- $X \in P \cap \{0,1\}^n$: random point obtained by randomized swap rounding.

Theorem

$$\Pr[f(X) \leq (1-\delta)F(x)] \leq e^{-F(x)\delta^2/8} \quad \forall \delta > 0.$$

$\Rightarrow$ If x approximately maximizes F then X approximately maximizes f.

Remarks

- A deterministic algorithm was already known for obtaining $X \in \{0,1\}^n$ such that $f(X) \geq F(x)$ (Calinescu et al. [2007]).
- Advantage of randomized approach: handle additional weak linear/submodular constraints.

Concentration for submodular functions

Consider the following setting:

- $f : \{0,1\}^n \to \mathbb{R}_+$: monotone submodular function with marginal values ≤ 1,
- $F : [0,1]^n \to \mathbb{R}_+$: multilinear extension of f,
- $x \in P$: point in matroid polytope P to round,
- $X \in P \cap \{0,1\}^n$: random point obtained by randomized swap rounding.

Theorem

$$\Pr[f(X) \leq (1-\delta)F(x)] \leq e^{-F(x)\delta^2/8} \quad \forall \delta > 0.$$

$\Rightarrow$ If x approximately maximizes F then X approximately maximizes f.

Remarks

- A deterministic algorithm was already known for obtaining $X \in \{0,1\}^n$ such that $f(X) \geq F(x)$ (Calinescu et al. [2007]).
- Advantage of randomized approach: handle additional weak linear/submodular constraints.

Concentration for submodular functions

Consider the following setting:

- $f : \{0,1\}^n \to \mathbb{R}_+$: monotone submodular function with marginal values ≤ 1,
- $F : [0,1]^n \to \mathbb{R}_+$: multilinear extension of f,
- $x \in P$: point in matroid polytope P to round,
- $X \in P \cap \{0,1\}^n$: random point obtained by randomized swap rounding.

Theorem

$$\Pr[f(X) \leq (1-\delta)F(x)] \leq e^{-F(x)\delta^2/8} \quad \forall \delta > 0.$$

$\Rightarrow$ If x approximately maximizes F then X approximately maximizes f.

Remarks

- A deterministic algorithm was already known for obtaining $X \in \{0,1\}^n$ such that $f(X) \geq F(x)$ (Calinescu et al. [2007]).
- Advantage of randomized approach: handle additional weak linear/submodular constraints.

Outline

Some consequences/applications

A congestion minization problem

Given:
- Matroid $M = (S, \mathcal{I})$,
- Matrix $A \in \mathbb{R}^{m \times S}$.

Task:
- $\min\{\lambda \mid \exists$ base B in M with $A \cdot \mathbf{1}_B \leq \lambda \mathbf{1}\}$.

Theorem

There is an $O(\log m / \log\log m)$-approximation to the above problem.

Network routing: comparison to previous results

Consider congestion minimization in a network routing context: there are m source-destination pairs (s_i, t_i), for each of which a set of s_i-t_i paths is given.

- If one path per commodity has to be chosen: $O(\log m / \log\log m)$-approximation by Raghavan and Thompson [1987].
- k_i paths have to be chosen for commodity i: $O(\log m / \log logm)$-approximation by Srinivasan [2001].
- Using swap rounding, $O(\log m / \log\log m)$ approximation is obtained for any matroid constraint on the paths to choose.

Some consequences/applications

A congestion minization problem

Given:
- Matroid $M = (S, \mathcal{I})$,
- Matrix $A \in \mathbb{R}^{m \times S}$.

Task:
- $\min\{\lambda \mid \exists$ base B in M with $A \cdot \mathbf{1}_B \leq \lambda \mathbf{1}\}$.

Theorem

There is an $O(\log m / \log \log m)$-approximation to the above problem.

Network routing: comparison to previous results

Consider congestion minimization in a network routing context: there are m source-destination pairs (s_i, t_i), for each of which a set of s_i-t_i paths is given.

- If one path per commodity has to be chosen: $O(\log m / \log \log m)$-approximation by Raghavan and Thompson [1987].
- k_i paths have to be chosen for commodity i: $O(\log m / \log logm)$-approximation by Srinivasan [2001].
- Using swap rounding, $O(\log m / \log \log m)$ approximation is obtained for any matroid constraint on the paths to choose.

Some consequences/applications

A congestion minization problem

Given:
- Matroid $M = (S, \mathcal{I})$,
- Matrix $A \in \mathbb{R}^{m \times S}$.

Task:
- $\min\{\lambda \mid \exists$ base B in M with $A \cdot \mathbf{1}_B \leq \lambda \mathbf{1}\}$.

Theorem

There is an $O(\log m / \log \log m)$-approximation to the above problem.

Network routing: comparison to previous results

Consider congestion minimization in a network routing context: there are m source-destination pairs (s_i, t_i), for each of which a set of s_i-t_i paths is given.

- If one path per commodity has to be chosen: $O(\log m / \log \log m)$-approximation by Raghavan and Thompson [1987].
- k_i paths have to be chosen for commodity i: $O(\log m / \log \mathit{logm})$-approximation by Srinivasan [2001].
- Using swap rounding, $O(\log m / \log \log m)$ approximation is obtained for any matroid constraint on the paths to choose.

Some consequences/applications (II)

Max-min submodular allocation

Given:
- Constant number k of agents interested in a set N of items.
- Agent $i \in [k]$ has monotone submodular utility funct. $w_i : 2^N \to \mathbb{R}_+$.

Task:
- Find allocation of items to players, i.e., disjoint sets $S_1, \dots, S_k \subseteq N$ maximizing $\min_{i \in [k]} w_i(S_i)$.

Theorem

There is a $(1 - 1/e - \epsilon)$-approximation to the above problem for any $\epsilon > 0$.

Sketch of algorithm

(i) Guess a constant number of items for each agent.

(ii) Get $(1 - 1/e)$-approx. to following relaxation using (variant of) continuous greedy: $\max\{\min_{i \in [k]} F_i(x_{i1}, \dots, x_{in}) \mid \sum_{i \in [k]} x_{ij} \leq 1 \ \forall j \in N, x_{ij} \geq 0\}$, where F_i is multilinear extension of w_i, and $n = |N|$.

(iii) Round obtained fractional solution.

Theorem (consequence of Mirrokni et al. [2008])

A $(1 - (1 - 1/|N|)^{|N|} - \epsilon)$-approximation, requires exponentially many queries.

Outline

Conclusions

- Randomized swap rounding provides a unifying and simple framework for several known applications.
- Generality of matroids and matroid intersections allows us to easily handle richer sets of constraints.
- Lower-tail concentration bound for submodular functions, allows for approximate maximization of submodular functions under a variety of hard/weak constraints.

- Extension of the general swap rounding framework to other problems?
- Extension of martingale concentration argument to other settings?
- Derandomization?

References I

A. Asadpour, M. X. Goemans, A. Madry, S. Oveis Gharan, and A. Saberi. An $O(\log n/\log\log n)$-approximation algrithm for the asymmetric traveling salesman problem. In *Proceedings of the 20th Annual ACM -SIAM Symposium on Discrete Algorithms (SODA)*, 2010.

G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák. Maximizing a submodular set function subject to a matroid constraint. In *IPCO '07: Proceedings of the 12th international conference on Integer Programming and Combinatorial Optimization*, pages 182–196, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 978-3-540-72791-0.

R. Gandhi, S. Khuller, S. Parthasarathy, and A. Srinivasan. Dependent rounding and its applications to approximation algorithms. *J. ACM*, 53(3):324–360, 2006. ISSN 0004-5411. doi: http://doi.acm.org/10.1145/1147954.1147956.

V. Mirrokni, M. Schapira, and J. Vondrák. Tight information-theoretic lower bounds for welfare maximization in combinatorial auctions. In *EC '08: Proceedings of the 9th ACM conference on Electronic commerce*, pages 70–77, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-169-9. doi: http://doi.acm.org/10.1145/1386790.1386805.

References II

A. Panconesi and A. Srinivasan. Randomized distributed edge coloring via an extension of the chernoff–hoeffding bounds. *SIAM Journal on Computing*, 26(2):350–368, 1997. ISSN 0097-5397. doi: http://dx.doi.org/10.1137/S0097539793250767.

P. Raghavan and C. D. Thompson. Randomized rounding: A technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374, December 1987.

A. Srinivasan. Distributions on level-sets with applications to approximation algorithms. In *Proceedings of the 42nd IEEE symposium on Foundations of Computer Science (FOCS)*, page 588, Washington, DC, USA, 2001. IEEE Computer Society. ISBN 0-7695-1390-5.

J. Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 67–74, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-047-0. doi: http://doi.acm.org/10.1145/1374376.1374389.