Tutorial: The Lasserre Hierarchy in Approximation algorithms

Thomas Rothvoß

Department of Mathematics, MIT

MAPSP 2013

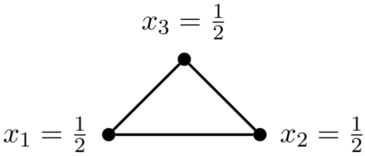
Massachusetts Institute of Technology

Motivation

Problem: Weak LP $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$

Example: Independent Set

- Relaxation: $K = \{\mathbb{R}^V \mid x_u + x_v \leq 1 \; \forall (u,v) \in E\}$



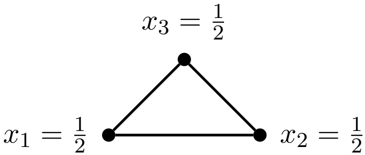
Motivation

Problem: Weak LP $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$

- **Option I:** Find and add violated inequalities
  Cons: Which ones? Solvable in polytime?

Example: Independent Set

- Relaxation: $K = \{\mathbb{R}^V \mid x_u + x_v \leq 1 \; \forall (u,v) \in E\}$



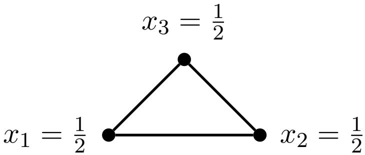
Motivation

Problem: Weak LP $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$

- **Option I:** Find and add violated inequalities
  Cons: Which ones? Solvable in polytime?
- **Option II:** Apply the Lasserre hierarchy!
  Pros: Solvable in polytime + known properties

Example: Independent Set

- Relaxation: $K = \{\mathbb{R}^V \mid x_u + x_v \leq 1 \ \forall (u,v) \in E\}$



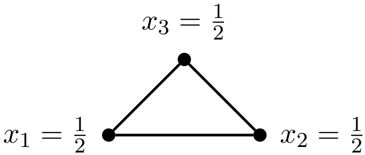
Motivation

Problem: Weak LP $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$

- **Option I:** Find and add violated inequalities
  Cons: Which ones? Solvable in polytime?
- **Option II:** Apply the Lasserre hierarchy!
  Pros: Solvable in polytime + known properties

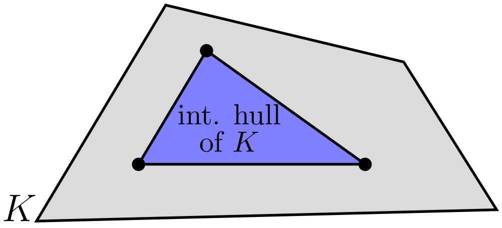
Example: Independent Set

- Relaxation: $K = \{\mathbb{R}^V \mid x_u + x_v \leq 1 \; \forall (u,v) \in E\}$

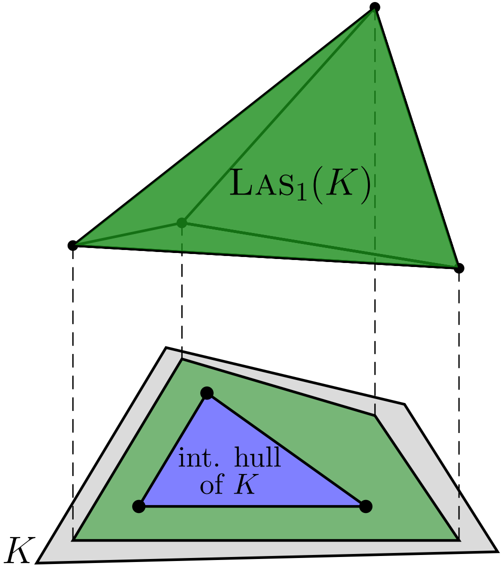


Try to convince you: Option II is better!

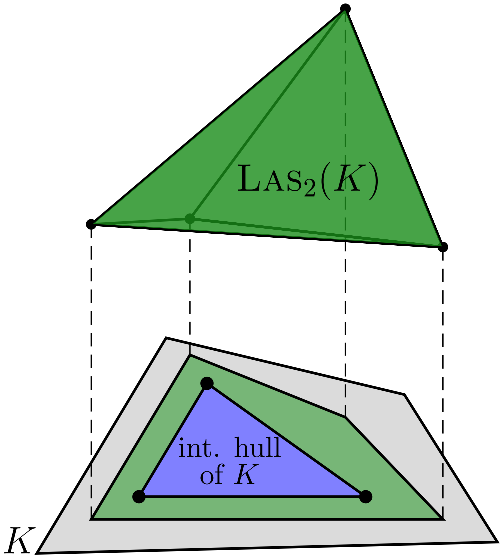
Lift-and-project methods



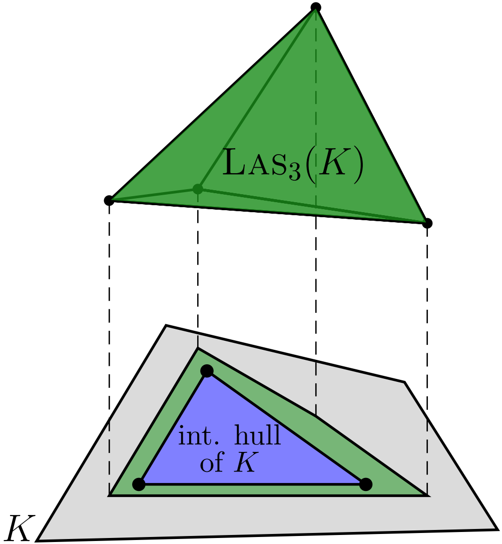
Lift-and-project methods



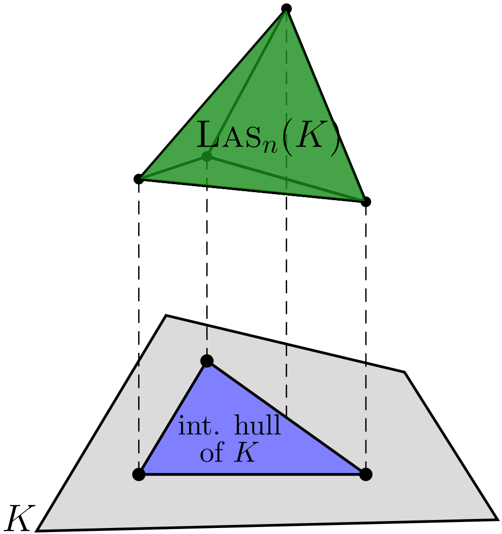
Lift-and-project methods



Lift-and-project methods



Lift-and-project methods



Positive semidefinitness

For a symmetric matrix M the following is equivalent

- **positive semidefinite**

Positive semidefinitness

For a symmetric matrix M the following is equivalent

- **positive semidefinite**
- $\boldsymbol{x}^T M \boldsymbol{x} \geq 0 \; \forall \boldsymbol{x}$

Positive semidefinitness

For a symmetric matrix M the following is equivalent

- **positive semidefinite**
- $\boldsymbol{x}^T M \boldsymbol{x} \geq 0 \; \forall \boldsymbol{x}$
- Any principal submatrix has det ≥ 0

Positive semidefinitness

For a symmetric matrix M the following is equivalent

- **positive semidefinite**
- $\boldsymbol{x}^T M \boldsymbol{x} \geq 0 \; \forall \boldsymbol{x}$
- Any principal submatrix has det ≥ 0
- $\exists$ vectors $\boldsymbol{v}_i$ with $M_{ij} = \langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

- $y_{\{i\}} = x_i$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

- $y_{\{i\}} = x_i$
- $y_\emptyset = 1$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

- $y_{\{i\}} = x_i$
- $y_\emptyset = 1$

Round-t Lasserre relaxation

$$\begin{aligned} (y_{I \cup J})_{|I|,|J| \leq t} &\succeq 0 \\ \left(\sum_{i \in [n]} A_{\ell i} y_{I \cup J \cup \{i\}} - b_\ell y_{I \cup J} \right)_{|I|,|J| \leq t} &\succeq 0 \quad \forall \ell \in [m] \\ y_\emptyset &= 1 \end{aligned}$$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

- $y_{\{i\}} = x_i$
- $y_\emptyset = 1$

moment matrix

Round-t Lasserre relaxation

$$\begin{aligned}
\boxed{(y_{I \cup J})_{|I|,|J| \leq t}} &\succeq 0 \\
\left(\sum_{i \in [n]} A_{\ell i} y_{I \cup J \cup \{i\}} - b_\ell y_{I \cup J} \right)_{|I|,|J| \leq t} &\succeq 0 \quad \forall \ell \in [m] \\
y_\emptyset &= 1
\end{aligned}$$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

slack moment matrix

- $y_{\{i\}} = x_i$
- $y_\emptyset = 1$

moment matrix

Round-t Lasserre relaxation

$$\begin{array}{rcll} \boxed{(y_{I \cup J})_{|I|,|J| \leq t}} & \succeq & 0 & \\ \boxed{\left(\sum_{i \in [n]} A_{\ell i} y_{I \cup J \cup \{i\}} - b_\ell y_{I \cup J} \right)_{|I|,|J| \leq t}} & \succeq & 0 & \forall \ell \in [m] \\ y_\emptyset & = & 1 & \end{array}$$

Round-t Lasserre relaxation

- Given: $K = \{x \in \mathbb{R}^n \mid Ax \geq b\}$.
- Introduce variables for $I \subseteq [n]$, $|I| \leq t$

$$y_I \equiv \bigwedge_{i \in I} (x_i = 1)$$

slack moment matrix

- $y_{\{i\}} = x_i$
- $y_\emptyset = 1$

moment matrix

Round-t Lasserre relaxation

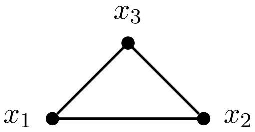
$$\begin{aligned}
\boxed{(y_{I \cup J})_{|I|,|J| \leq t}} &\succeq 0 \\
\boxed{\left(\sum_{i \in [n]} A_{\ell i} y_{I \cup J \cup \{i\}} - b_\ell y_{I \cup J} \right)_{|I|,|J| \leq t}} &\succeq 0 \quad \forall \ell \in [m] \\
y_\emptyset &= 1
\end{aligned}$$

- Solvable in time $n^{O(t)} m^{O(1)}$

Example: Independent Set

Moment matrix for $t = 1$:

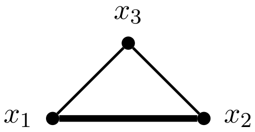
$$M_1(y) = \begin{array}{cc} \begin{array}{cccc} \emptyset & \{1\} & \{2\} & \{3\} \end{array} & \\ \begin{pmatrix} 1 & y_1 & y_2 & y_3 \\ y_1 & y_1 & y_{12} & y_{13} \\ y_2 & y_{12} & y_2 & y_{23} \\ y_3 & y_{13} & y_{23} & y_3 \end{pmatrix} & \begin{array}{c} \emptyset \\ \{1\} \\ \{2\} \\ \{3\} \end{array} \end{array}$$



Example: Independent Set

Moment matrix for $t = 1$:

$$M_1(y) = \begin{array}{cc} \begin{matrix} \emptyset & \{1\} & \{2\} & \{3\} \end{matrix} & \\ \begin{pmatrix} 1 & y_1 & y_2 & y_3 \\ y_1 & y_1 & y_{12} & y_{13} \\ y_2 & y_{12} & y_2 & y_{23} \\ y_3 & y_{13} & y_{23} & y_3 \end{pmatrix} & \begin{matrix} \emptyset \\ \{1\} \\ \{2\} \\ \{3\} \end{matrix} \end{array}$$



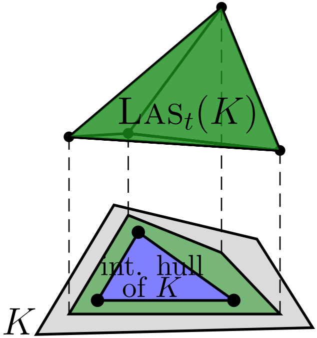
Moment matrix for edge $(1, 2)$ for $t = 1$:

$$\begin{array}{cc} \begin{matrix} \emptyset & \{1\} & \{2\} & \{3\} \end{matrix} & \\ \begin{pmatrix} 1 - y_1 - y_2 & y_1 - y_1 - y_{12} & y_2 - y_{12} - y_2 & y_3 - y_{13} - y_{23} \\ y_1 - y_1 - y_{12} & y_1 - y_1 - y_{12} & y_{12} - y_{12} - y_{12} & y_{13} - y_{13} - y_{123} \\ y_2 - y_{12} - y_2 & y_{12} - y_{12} - y_{12} & y_2 - y_{12} - y_2 & y_{23} - y_{123} - y_{23} \\ y_3 - y_{13} - y_{23} & y_{13} - y_{13} - y_{123} & y_{23} - y_{123} - y_{23} & y_3 - y_{13} - y_{23} \end{pmatrix} & \begin{matrix} \emptyset \\ \{1\} \\ \{2\} \\ \{3\} \end{matrix} \end{array}$$

Basic properties (1)

Lemma

$conv(K \cap \{0,1\}^n) \subseteq \text{LAS}_t^{proj}(K)$



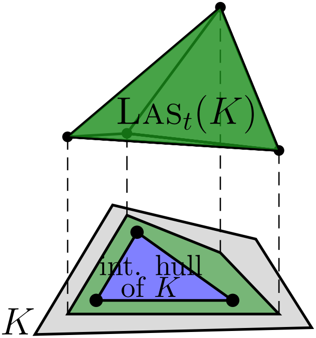
Basic properties (1)

Lemma

$conv(K \cap \{0,1\}^n) \subseteq \textsc{Las}_t^{proj}(K)$

- For $x \in K \cap \{0,1\}^n$ define

$$y_I := \begin{cases} 1 & x_i = 1 \ \forall i \in I \\ 0 & \text{otherwise} \end{cases}$$



Basic properties (1)

Lemma

$conv(K \cap \{0,1\}^n) \subseteq \textsc{Las}_t^{proj}(K)$

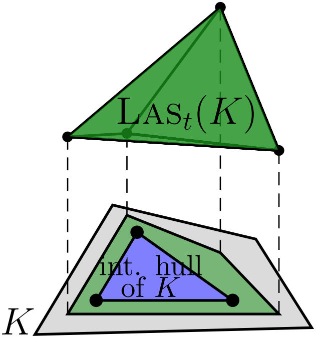
- For $x \in K \cap \{0,1\}^n$ define

$$y_I := \begin{cases} 1 & x_i = 1 \; \forall i \in I \\ 0 & \text{otherwise} \end{cases}$$

- Then

$$(M_t(y))_{I,J} = y_{I \cup J} = y_I \cdot y_J = (yy^T)_{I,J}$$

and $yy^T \succeq 0$.



Basic properties (1)

Lemma

$conv(K \cap \{0,1\}^n) \subseteq \textsc{Las}_t^{proj}(K)$

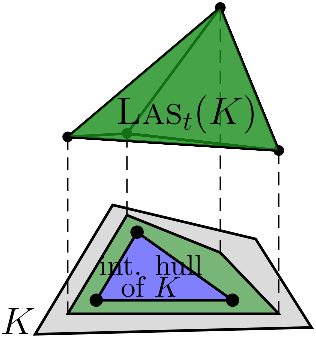
- For $x \in K \cap \{0,1\}^n$ define

$$y_I := \begin{cases} 1 & x_i = 1 \ \forall i \in I \\ 0 & \text{otherwise} \end{cases}$$

- Then

$$(M_t(y))_{I,J} = y_{I \cup J} = y_I \cdot y_J = (yy^T)_{I,J}$$

and $yy^T \succeq 0$.



- Similarly $\sum_{i=1}^n a_i y_{I \cup J \cup \{i\}} - \beta y_{I \cup J} = (ax - \beta) \cdot y_I \cdot y_J$ and $(ax - \beta) \cdot yy^T \succeq 0$. □

Basic properties (2)

Lemma

For $y \in \textsc{Las}_t(K)$*:*

a) $(y_1, \ldots, y_n) \in K$

Basic properties (2)

Lemma

For $y \in \textsc{Las}_t(K)$*:*

a) $(y_1, \ldots, y_n) \in K$

a) Slack moment matrix for $ax \geq \beta$ contains diagonal entry $(\emptyset, \emptyset)$: $\sum_{i=1}^{n} a_i y_i - y_\emptyset \beta \geq 0$

Basic properties (2)

Lemma

For $y \in \textsc{Las}_t(K)$*:*

a) $(y_1, \ldots, y_n) \in K$

b) $0 \leq y_I \leq 1$ *for all* $|I| \leq t$

a) Slack moment matrix for $ax \geq \beta$ contains diagonal entry $(\emptyset, \emptyset)$: $\sum_{i=1}^n a_i y_i - y_\emptyset \beta \geq 0$

Basic properties (2)

Lemma

For $y \in \textsc{Las}_t(K)$:

a) $(y_1, \ldots, y_n) \in K$

b) $0 \leq y_I \leq 1$ *for all* $|I| \leq t$

a) Slack moment matrix for $ax \geq \beta$ contains diagonal entry $(\emptyset, \emptyset)$: $\sum_{i=1}^{n} a_i y_i - y_\emptyset \beta \geq 0$

b) Submatrix of $M_t(y)$ indexed by $\{\emptyset, I\}$:

$$\begin{array}{c|c|} & \begin{array}{cc} \emptyset & I \end{array} \\ \begin{array}{c} \emptyset \\ I \end{array} & \begin{pmatrix} y_\emptyset & y_I \\ y_I & y_I \end{pmatrix} \end{array} = y_I(1 - y_I) \geq 0$$

Basic properties (2)

Lemma

For $y \in \textsc{Las}_t(K)$*:*

a) $(y_1, \ldots, y_n) \in K$

b) $0 \leq y_I \leq 1$ *for all* $|I| \leq t$

c) $y_J \leq y_I$ *for* $I \subseteq J$ *with* $|I|, |J| \leq t$*.*

a) Slack moment matrix for $ax \geq \beta$ contains diagonal entry $(\emptyset, \emptyset)$: $\sum_{i=1}^{n} a_i y_i - y_\emptyset \beta \geq 0$

b) Submatrix of $M_t(y)$ indexed by $\{\emptyset, I\}$:

$$\begin{array}{c|c|l} & \begin{array}{cc} \emptyset & I \end{array} & \\ \begin{array}{c} \emptyset \\ I \end{array} & \begin{pmatrix} y_\emptyset & y_I \\ y_I & y_I \end{pmatrix} & = y_I(1 - y_I) \geq 0 \end{array}$$

Basic properties (2)

Lemma

For $y \in \textsc{Las}_t(K)$:

a) $(y_1, \dots, y_n) \in K$

b) $0 \leq y_I \leq 1$ *for all* $|I| \leq t$

c) $y_J \leq y_I$ *for* $I \subseteq J$ *with* $|I|, |J| \leq t$.

a) Slack moment matrix for $ax \geq \beta$ contains diagonal entry $(\emptyset, \emptyset)$: $\sum_{i=1}^{n} a_i y_i - y_\emptyset \beta \geq 0$

b) Submatrix of $M_t(y)$ indexed by $\{\emptyset, I\}$:

$$\begin{array}{c|c|} & \begin{array}{cc} \emptyset & I \end{array} \\ \begin{array}{c} \emptyset \\ I \end{array} & \begin{pmatrix} y_\emptyset & y_I \\ y_I & y_I \end{pmatrix} \end{array} = y_I(1 - y_I) \geq 0$$

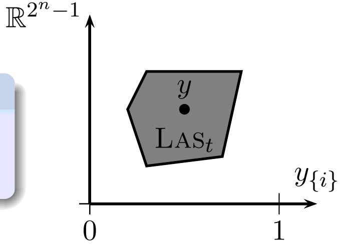
c) Submatrix of $M_t(y)$ indexed by $\{I, J\}$:

$$\begin{array}{c|c|} & \begin{array}{cc} I & J \end{array} \\ \begin{array}{c} I \\ J \end{array} & \begin{pmatrix} y_I & y_J \\ y_J & y_J \end{pmatrix} \end{array} = y_J(y_I - y_J) \geq 0$$

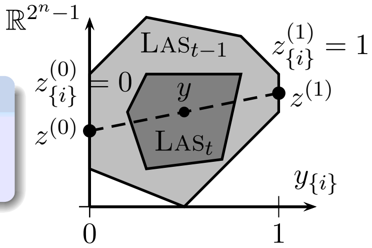
Inducing on one variable

Lemma

For $y \in \textsc{Las}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \textsc{Las}_{t-1}(K) \mid z_i \in \{0,1\}\}$.



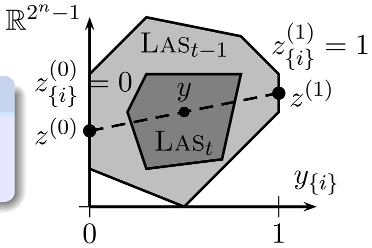
Inducing on one variable



Lemma

For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in \textit{conv}\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

Inducing on one variable

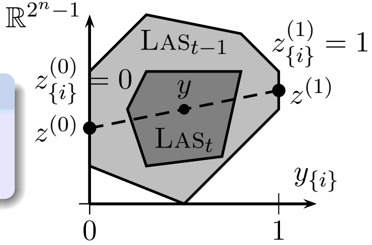


Lemma

For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

- Define $z_I^{(1)} := \frac{y_{I \cup \{i\}}}{y_i}$ and $z_I^{(0)} := \frac{y_I - y_{I\cup\{i\}}}{1-y_i}$

Inducing on one variable

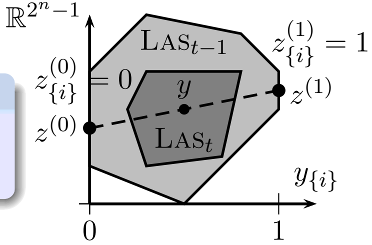


Lemma

For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

- Define $z^{(1)}_I := \frac{y_{I\cup\{i\}}}{y_i}$ and $z^{(0)}_I := \frac{y_I - y_{I\cup\{i\}}}{1-y_i}$
- Clearly $y = y_i \cdot z^{(1)} + (1-y_i) \cdot z^{(0)}$

Inducing on one variable

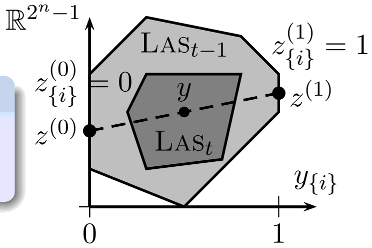


Lemma

For $y \in \text{Las}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{Las}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

- Define $z_I^{(1)} := \frac{y_{I \cup \{i\}}}{y_i}$ and $z_I^{(0)} := \frac{y_I - y_{I\cup\{i\}}}{1-y_i}$
- Clearly $y = y_i \cdot z^{(1)} + (1-y_i) \cdot z^{(0)}$
- Moreover $z_i^{(0)} = 0, z_i^{(1)} = 1$. Remains to show psdness.

Inducing on one variable

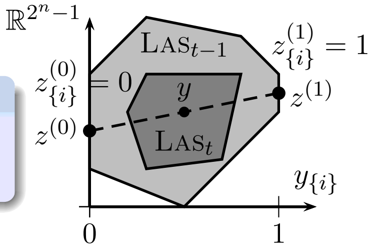


Lemma

For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

- Define $z_I^{(1)} := \frac{y_{I \cup \{i\}}}{y_i}$ and $z_I^{(0)} := \frac{y_I - y_{I \cup \{i\}}}{1 - y_i}$
- Clearly $y = y_i \cdot z^{(1)} + (1 - y_i) \cdot z^{(0)}$
- Moreover $z_i^{(0)} = 0, z_i^{(1)} = 1$. Remains to show psdness.
- Take $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle = y_{I \cup J}$ for $|I|, |J| \leq t$.

Inducing on one variable



Lemma

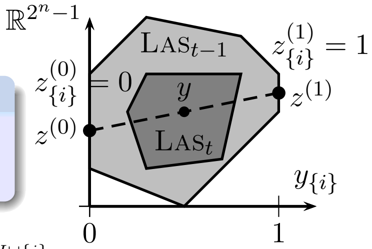
For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

- Define $z_I^{(1)} := \frac{y_{I\cup\{i\}}}{y_i}$ and $z_I^{(0)} := \frac{y_I - y_{I\cup\{i\}}}{1-y_i}$
- Clearly $y = y_i \cdot z^{(1)} + (1-y_i)\cdot z^{(0)}$
- Moreover $z_i^{(0)} = 0, z_i^{(1)} = 1$. Remains to show psdness.
- Take $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J\rangle = y_{I\cup J}$ for $|I|,|J| \leq t$.
- Then

$$\left\langle \frac{\boldsymbol{v}_{I\cup\{i\}}}{\sqrt{y_i}}, \frac{\boldsymbol{v}_{J\cup\{i\}}}{\sqrt{y_i}} \right\rangle = \frac{y_{I\cup J\cup\{i\}}}{y_i} = z^{(1)}_{I\cup J}$$

and $M_{t-1}(z^{(1)}) \succeq 0$.

Inducing on one variable



Lemma

For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

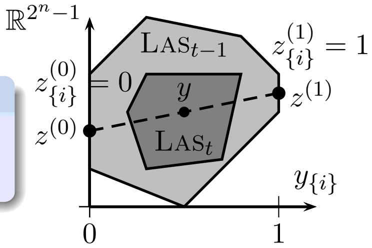
- Define $z_I^{(1)} := \frac{y_{I\cup\{i\}}}{y_i}$ and $z_I^{(0)} := \frac{y_I - y_{I\cup\{i\}}}{1-y_i}$
- Clearly $y = y_i \cdot z^{(1)} + (1-y_i)\cdot z^{(0)}$
- Moreover $z_i^{(0)} = 0, z_i^{(1)} = 1$. Remains to show psdness.
- Take $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J\rangle = y_{I\cup J}$ for $|I|,|J| \leq t$.
- Moreover

$$\left\langle \frac{\boldsymbol{v}_I - \boldsymbol{v}_{I\cup\{i\}}}{\sqrt{1-y_i}}, \frac{\boldsymbol{v}_J - \boldsymbol{v}_{J\cup\{i\}}}{\sqrt{1-y_i}} \right\rangle = \frac{\boldsymbol{v}_I\boldsymbol{v}_J - \boldsymbol{v}_I\boldsymbol{v}_{J\cup\{i\}} - \boldsymbol{v}_J\boldsymbol{v}_{I\cup\{i\}} + \boldsymbol{v}_{I\cup\{i\}}\boldsymbol{v}_{J\cup\{i\}}}{1-y_i}$$

$$= \frac{y_{I\cup J} - y_{I\cup J\cup\{i\}}}{1-y_i} = z^{(0)}_{I\cup J}$$

Thus $M_{t-1}(z^{(0)}) \succeq 0$.

Inducing on one variable



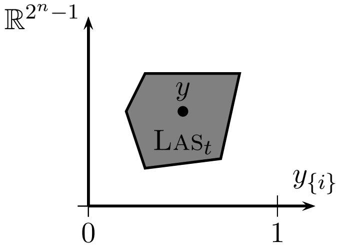
Lemma

For $y \in \text{LAS}_t(K)$, $t \geq 1$, $i \in [n]$,
$y \in conv\{z \in \text{LAS}_{t-1}(K) \mid z_i \in \{0,1\}\}$.

- Define $z^{(1)}_I := \frac{y_{I\cup\{i\}}}{y_i}$ and $z^{(0)}_I := \frac{y_I - y_{I\cup\{i\}}}{1-y_i}$
- Clearly $y = y_i \cdot z^{(1)} + (1-y_i)\cdot z^{(0)}$
- Moreover $z^{(0)}_i = 0, z^{(1)}_i = 1$. Remains to show psdness.
- Take $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J\rangle = y_{I\cup J}$ for $|I|,|J| \leq t$.
- Similar for slack moment matrices.

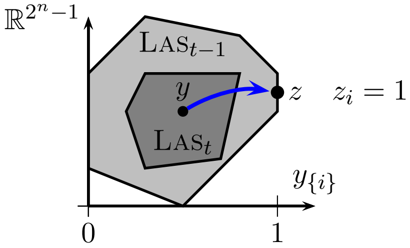
Consequences (1)

Operation: "Induce on $x_i = 1$"



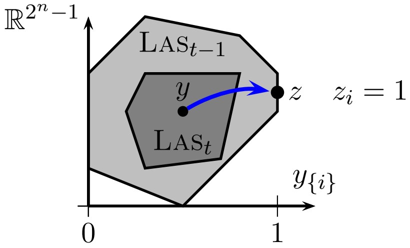
Consequences (1)

Operation: "Induce on $x_i = 1$"



Consequences (1)

Operation: “Induce on $x_i = 1$”



Lemma

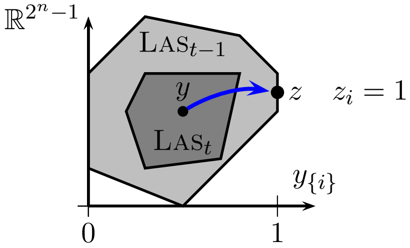
For $y \in \textsc{Las}_t(K)$, pick a set $|S| \leq t$:

$$y \in conv\{z \in \textsc{Las}_{t-|S|}(K) \mid z_i \in \{0,1\} \;\; \forall i \in S\}$$

- Explicit formula for each z via **inclusion exclusion formula**

Consequences (1)

Operation: "Induce on $x_i = 1$"



Lemma

For $y \in \text{LAS}_t(K)$, pick a set $|S| \leq t$:

$$y \in conv\{z \in \text{LAS}_{t-|S|}(K) \mid z_i \in \{0,1\} \;\; \forall i \in S\}$$

- Explicit formula for each z via **inclusion exclusion formula**
- Gap closed after n rounds

Consequences (2)

Lemma (Locally consistent probability distribution)

For $y \in \textsc{Las}_t(K)$ and $|S| \le t$, there is a random variable $X \in \{0,1\}^S$ with

$$\Pr\Big[\bigwedge_{i \in I} (X_i = 1)\Big] = y_I \quad \forall I \subseteq S$$

- For example for Independent Set, for $|S| \le t$ nodes, y gives a probability distribution of independent sets in $G[S]$

Application 1:

Scheduling On 2 Machines With Precedence Constraints

$$P2 \mid \text{prec}, p_j = 1 \mid C_{\max}$$

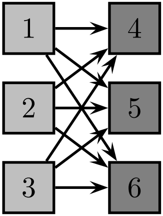
Source: [Svensson, unpublished, 2011]

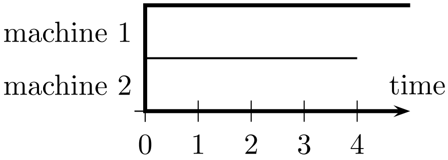
$P2 \mid \textbf{prec}, p_j = 1 \mid C_{\max}$

Input:

- **jobs** J with **unit processing time**
- **precedence constraints**
- 2 identical machines

Goal: minimize makespan



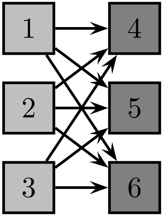


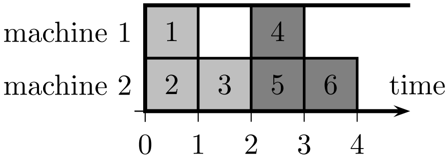
$P2 \mid \textbf{prec}, p_j = 1 \mid C_{\max}$

Input:

- **jobs** J with **unit processing time**
- **precedence constraints**
- 2 identical machines

Goal: minimize makespan



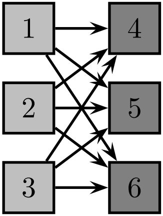


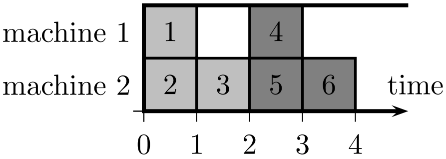
$P2 \mid \textbf{prec}, p_j = 1 \mid C_{\max}$

Input:

- **jobs** J with **unit processing time**
- **precedence constraints**
- 2 identical machines

Goal: minimize makespan





Known results:

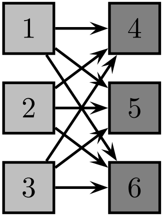
- **NP**-hard for general m

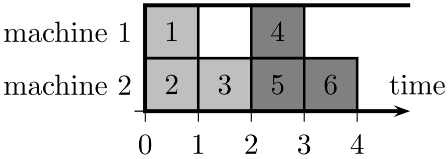
$P2 \mid \textbf{prec}, p_j = 1 \mid C_{\max}$

Input:

- **jobs** J with **unit processing time**
- **precedence constraints**
- 2 identical machines

Goal: minimize makespan





Known results:

- **NP**-hard for general m
- poly-time for 2 machines [Coffman, Graham '72]

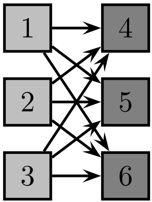
Time indexed LP

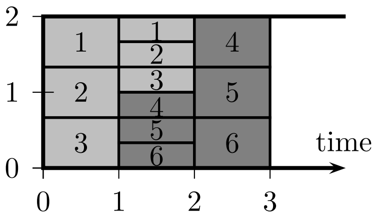
$$
\begin{array}{rcll}
\sum_{t=1}^{T} x_{jt} & = & 1 & \forall j \in J \\
\sum_{j \in J} x_{jt} & \leq & 2 & \forall t \in [T] \\
\sum_{t' \leq t} x_{it'} & \geq & \sum_{t' \leq t+1} x_{jt'} & \forall i \prec j \;\; \forall t \in [T] \\
x_{jt} & \geq & 0 & \forall j \in J \;\; \forall t \in [T]
\end{array}
$$

Time indexed LP

$$
\begin{array}{rcll}
\sum_{t=1}^{T} x_{jt} & = & 1 & \forall j \in J \\
\sum_{j \in J} x_{jt} & \leq & 2 & \forall t \in [T] \\
\sum_{t' \leq t} x_{it'} & \geq & \sum_{t' \leq t+1} x_{jt'} & \forall i \prec j \;\; \forall t \in [T] \\
x_{jt} & \geq & 0 & \forall j \in J \;\; \forall t \in [T]
\end{array}
$$

- Integrality gap is $\geq \frac{4}{3}$

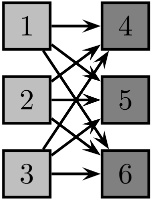


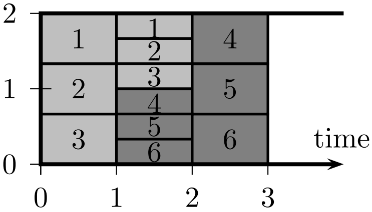


Time indexed LP

$$\begin{array}{rcll} \displaystyle\sum_{t=1}^{T} x_{jt} & = & 1 & \forall j \in J \\ \displaystyle\sum_{j \in J} x_{jt} & \leq & 2 & \forall t \in [T] \\ \displaystyle\sum_{t' \leq t} x_{it'} & \geq & \displaystyle\sum_{t' \leq t+1} x_{jt'} & \forall i \prec j \;\; \forall t \in [T] \\ x_{jt} & \geq & 0 & \forall j \in J \;\; \forall t \in [T] \end{array}$$

- Integrality gap is $\geq \frac{4}{3}$





- **Claim:** $y \in \textsc{Las}_1(LP(T)) \Rightarrow \exists$ schedule of length T

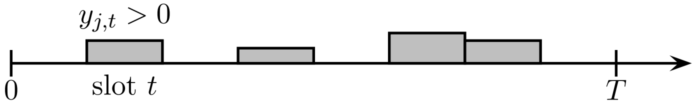
Scheduling algorithm

(1) Find $y \in \textsc{Las}_1(LP(T))$

Scheduling algorithm

(1) Find $y \in \textsc{Las}_1(LP(T))$

Fractional schedule for j:

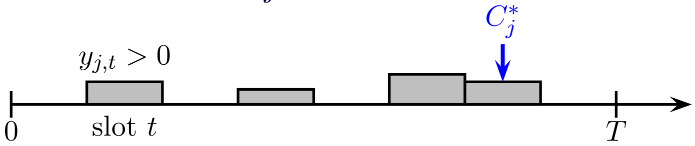


Scheduling algorithm

(1) Find $y \in \textsc{Las}_1(LP(T))$

(2) Fractional completion time $C_j^* := \max\{t \mid y^*_{\{(j,t)\}} > 0\}$

Fractional schedule for j:



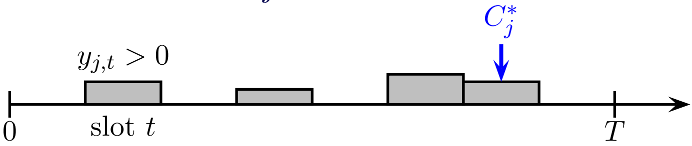
Scheduling algorithm

(1) Find $y \in \textsc{Las}_1(LP(T))$

(2) Fractional completion time $C_j^* := \max\{t \mid y^*_{\{(j,t)\}} > 0\}$

(3) Sort jobs s.t. $C_1^* \leq C_2^* \leq \ldots \leq C_n^*$

Fractional schedule for j:



Scheduling algorithm

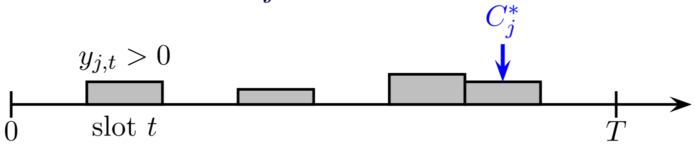
(1) Find $y \in \textsc{Las}_1(LP(T))$

(2) Fractional completion time $C_j^* := \max\{t \mid y^*_{\{(j,t)\}} > 0\}$

(3) Sort jobs s.t. $C_1^* \leq C_2^* \leq \ldots \leq C_n^*$

(4) Run list schedule w.r.t. ordering $\Rightarrow \sigma : J \to \mathbb{N}$

Fractional schedule for j:



Analysis

Lemma

For any job $\sigma_j \leq C_j^*$.

schedule σ:

0

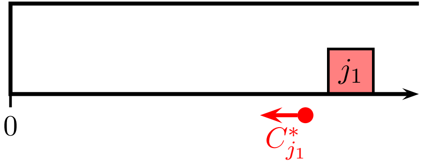
Analysis

Lemma

For any job $\sigma_j \leq C_j^*$.

- Consider job j_1 with smallest index, s.t. $\sigma_{j_1} > C^*_{j_1}$ (delete jobs $> j_1$)

schedule σ:

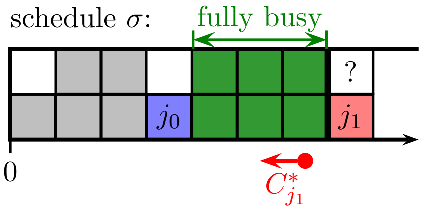


Analysis

Lemma

For any job $\sigma_j \leq C^*_j$.

- Consider job j_1 with smallest index, s.t. $\sigma_{j_1} > C^*_{j_1}$ (delete jobs $> j_1$)
- Let $j_0 < j_1$ highest index job scheduled alone

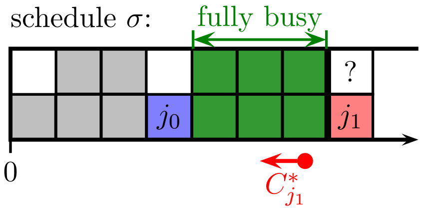


Analysis

Lemma

For any job $\sigma_j \leq C^*_j$.

- Consider job j_1 with smallest index, s.t. $\sigma_{j_1} > C^*_{j_1}$ (delete jobs $> j_1$)
- Let $j_0 < j_1$ highest index job scheduled alone
- Jobs in $J_0 := \{j < j_1 \mid \sigma_j > \sigma_{j_0}\} \cup \{j_1\}$ depend on j_0!

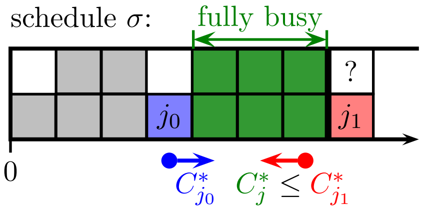


Analysis

Lemma

For any job $\sigma_j \leq C^*_j$.

- Consider job j_1 with smallest index, s.t. $\sigma_{j_1} > C^*_{j_1}$ (delete jobs $> j_1$)
- Let $j_0 < j_1$ highest index job scheduled alone
- Jobs in $J_0 := \{j < j_1 \mid \sigma_j > \sigma_{j_0}\} \cup \{j_1\}$ depend on j_0!

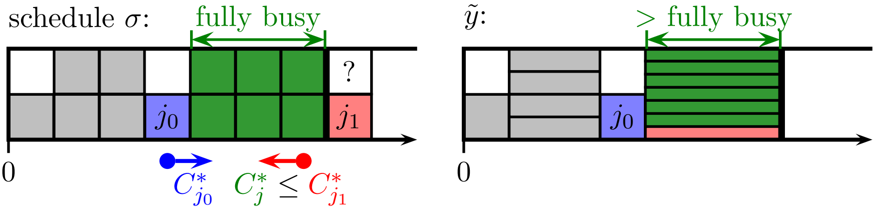


Analysis

Lemma

For any job $\sigma_j \leq C_j^*$.

- Consider job j_1 with smallest index, s.t. $\sigma_{j_1} > C_{j_1}^*$ (delete jobs $> j_1$)
- Let $j_0 < j_1$ highest index job scheduled alone
- Jobs in $J_0 := \{j < j_1 \mid \sigma_j > \sigma_{j_0}\} \cup \{j_1\}$ depend on j_0!
- Induce on $x_{j_0, C_j^*} = 1 \to \tilde{y} \in \text{LAS}_0(LP(T))$

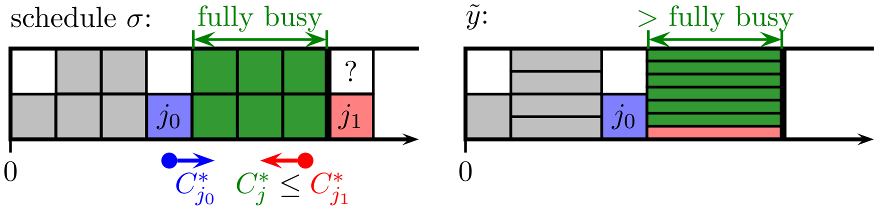


Analysis

Lemma

For any job $\sigma_j \leq C_j^*$.

- Consider job j_1 with smallest index, s.t. $\sigma_{j_1} > C_{j_1}^*$ (delete jobs $> j_1$)
- Let $j_0 < j_1$ highest index job scheduled alone
- Jobs in $J_0 := \{j < j_1 \mid \sigma_j > \sigma_{j_0}\} \cup \{j_1\}$ depend on j_0!
- Induce on $x_{j_0, C_j^*} = 1 \rightarrow \tilde{y} \in \textsc{Las}_0(LP(T))$
- For LP infeasible! **Contradiction!**



A PTAS for 3 machines?

Open problem

For $m = 3$ machines, is the gap for $f(\varepsilon)$-round Lasserre at most $1 + \varepsilon$?

- Neither known to be **NP**-hard, nor is a PTAS known!

Application 2:

Set Cover

Source: [Chlamtac, Friggstad, Georgiou 2012]

Subexp. Set Cover Approximation

Set Cover:

- **Input:** Family of sets $S_1, \dots, S_m \subseteq [n]$ with cost c_{S_j}
- **Goal:** Cover elements at minimum cost

Subexp. Set Cover Approximation

Set Cover:

- **Input:** Family of sets $S_1, \ldots, S_m \subseteq [n]$ with cost c_{S_j}
- **Goal:** Cover elements at minimum cost

Known:

- $\ln(n)$-apx [Chvátal '79]

Subexp. Set Cover Approximation

Set Cover:

- **Input:** Family of sets $S_1, \ldots, S_m \subseteq [n]$ with cost c_{S_j}
- **Goal:** Cover elements at minimum cost

Known:

- $\ln(n)$-apx [Chvátal '79]
- Reduction SAT to SET COVER with gap $(1-\varepsilon)\ln(n)$ in time $n^{O(1/\varepsilon)}$ [Dinur, Steurer '13]

Subexp. Set Cover Approximation

Set Cover:

- **Input:** Family of sets $S_1, \ldots, S_m \subseteq [n]$ with cost c_{S_j}
- **Goal:** Cover elements at minimum cost

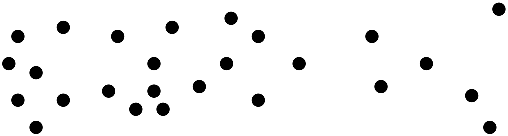
Known:

- $\ln(n)$-apx [Chvátal '79]
- Reduction SAT to SET COVER with gap $(1-\varepsilon)\ln(n)$ in time $n^{O(1/\varepsilon)}$ [Dinur, Steurer '13]

Theorem

There is a $(1-\varepsilon)\ln(n)$-apx in time $2^{\tilde{O}(n^{\varepsilon})}$.

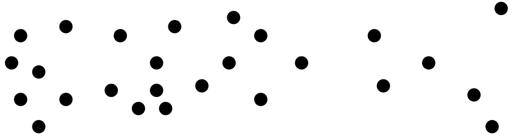
The algorithm



The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \left\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \right\}$$



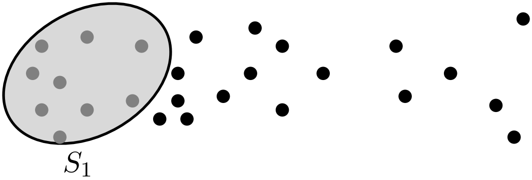
The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \;\forall \text{element } j;\; c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

  (3) Find set S_i covering most new elements



The algorithm

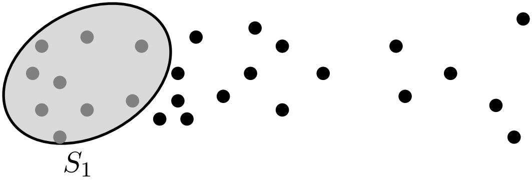
(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \left\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \right\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$



The algorithm

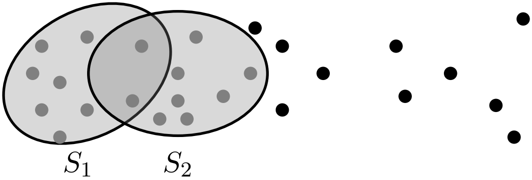
(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \left\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \right\}$$

(2) FOR $i = 1, \dots, n^\varepsilon$ DO

  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$



The algorithm

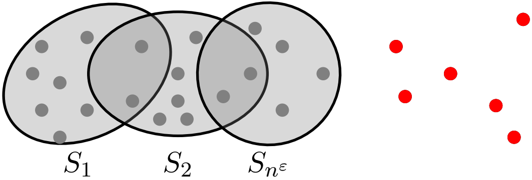
(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i: j \in S_i} x_i \geq 1 \; \forall \text{element } j;\; c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

(3) Find set S_i covering most new elements

(4) Induce on $x_i = 1 \rightarrow \tilde{y}$



The algorithm

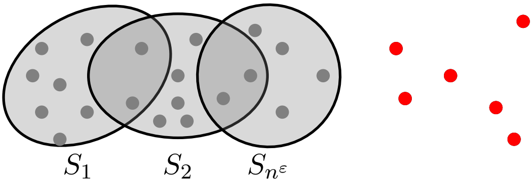
(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with
$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i : j \in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$

(5) Run **ln(|largest set|)-apx** for rest



The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \;\forall \text{element } j;\; c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

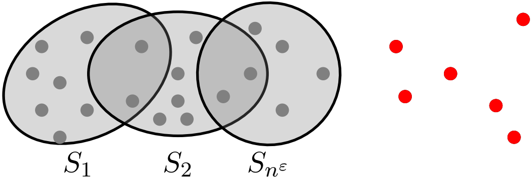
  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$

(5) Run **ln(|largest set|)-apx** for rest

Final $\tilde{y}$ has:

- $\tilde{y}_1 = \ldots = \tilde{y}_{n^\varepsilon} = 1$



The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \left\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \right\}$$

(2) FOR $i = 1, \dots, n^\varepsilon$ DO

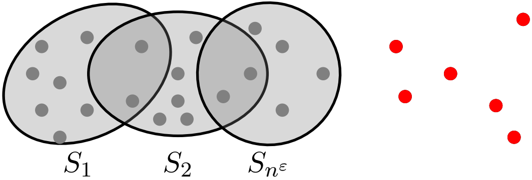
  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$

(5) Run **ln(|largest set|)-apx** for rest

Final $\tilde{y}$ has:

- $\tilde{y}_1 = \ldots = \tilde{y}_{n^\varepsilon} = 1$
- $\tilde{y} \in \textsc{Las}_0(K)$



The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i: j \in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

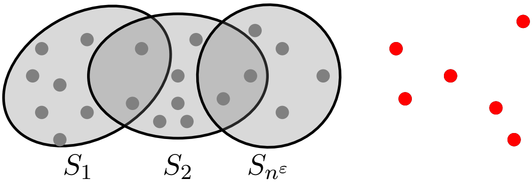
  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$

(5) Run **ln(|largest set|)-apx** for rest

Final $\tilde{y}$ has:

- $\tilde{y}_1 = \ldots = \tilde{y}_{n^\varepsilon} = 1$
- $\tilde{y} \in \textsc{Las}_0(K)$
- $c^T \tilde{y} \leq OPT$



The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j\in S_i} x_i \geq 1 \ \forall \text{element } j;\ c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \dots, n^\varepsilon$ DO

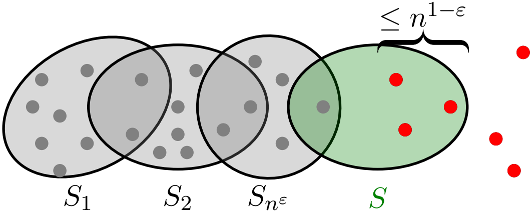
  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \to \tilde{y}$

(5) Run **ln(|largest set|)-apx** for rest

Final $\tilde{y}$ has:

- $\tilde{y}_1 = \ldots = \tilde{y}_{n^\varepsilon} = 1$
- $\tilde{y} \in \textsc{Las}_0(K)$
- $c^T \tilde{y} \leq OPT$



- $0 < \tilde{y}_S < 1 \Rightarrow |S \cap \text{remain. elem}| \leq n^{1-\varepsilon}$

The algorithm

(1) Compute $y \in \textsc{Las}_{n^\varepsilon}(K)$ with

$$K := \Big\{ x \in \mathbb{R}^m_{\geq 0} \mid \sum_{i:j \in S_i} x_i \geq 1 \; \forall \text{element } j; \; c^T x \leq OPT \Big\}$$

(2) FOR $i = 1, \ldots, n^\varepsilon$ DO

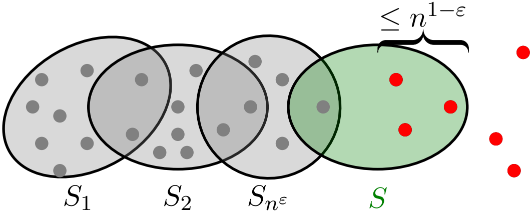
  (3) Find set S_i covering most new elements

  (4) Induce on $x_i = 1 \rightarrow \tilde{y}$

(5) Run **ln(|largest set|)-apx** for rest

Final $\tilde{y}$ has:

- $\tilde{y}_1 = \ldots = \tilde{y}_{n^\varepsilon} = 1$
- $\tilde{y} \in \textsc{Las}_0(K)$
- $c^T \tilde{y} \leq OPT$



- $0 < \tilde{y}_S < 1 \Rightarrow |S \cap \text{remain. elem}| \leq n^{1-\varepsilon}$
- **Apx-ratio:** $\ln(n^{1-\varepsilon}) = (1-\varepsilon)\ln(n)$

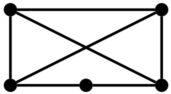
Application 3:

Max Cut

Source: [Goemans, Williamson '95]

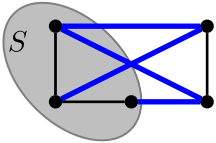
MaxCut

MaxCut: Given $G = (V, E)$. Maximize $|\delta(S)|$



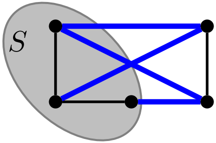
MaxCut

MaxCut: Given $G = (V, E)$. Maximize $|\delta(S)|$



MaxCut

MaxCut: Given $G = (V, E)$. Maximize $|\delta(S)|$

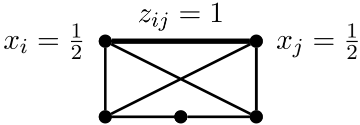


LP:

$$\max\left\{\sum_{e\in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\}\ \forall (i,j) \in E\right\}$$

MaxCut

MaxCut: Given $G = (V, E)$. Maximize $|\delta(S)|$



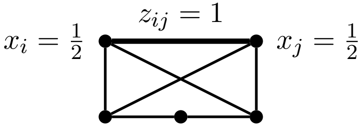
LP:

$$\max \left\{ \sum_{e \in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\} \; \forall (i,j) \in E \right\}$$

- Always $OPT_f = |E|$

MaxCut

MaxCut: Given $G = (V, E)$. Maximize $|\delta(S)|$



LP:

$$\max\left\{ \sum_{e \in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\} \ \forall (i,j) \in E \right\}$$

- Always $OPT_f = |E|$
- Consider $(x, z) \in \textsc{Las}_3(K)$

Vector representation

Observation

For $y \in \textsc{Las}_t(K)$, there are $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle = y_{I \cup J}$ for $|I|, |J| \leq t$.

Vector representation

Observation

For $y \in \textsc{Las}_t(K)$, there are $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle = y_{I \cup J}$ for $|I|, |J| \leq t$.

- $\|\boldsymbol{v}_I\|_2^2 = y_I$

Vector representation

Observation

For $y \in \textsc{Las}_t(K)$, there are $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle = y_{I \cup J}$ for $|I|, |J| \leq t$.

- $\|\boldsymbol{v}_I\|_2^2 = y_I$
- $\|\boldsymbol{v}_\emptyset\|_2^2 = 1$

Vector representation

Observation

For $y \in \textsc{Las}_t(K)$, there are $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle = y_{I \cup J}$ for $|I|, |J| \leq t$.

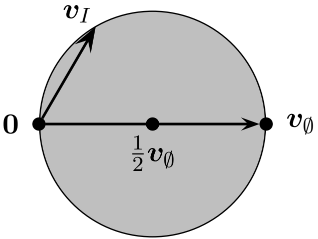
- $\|\boldsymbol{v}_I\|_2^2 = y_I$
- $\|\boldsymbol{v}_\emptyset\|_2^2 = 1$
- $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle \geq 0$

Vector representation

Observation

For $y \in \textsc{Las}_t(K)$, there are $\boldsymbol{v}_I$ with $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle = y_{I \cup J}$ for $|I|, |J| \leq t$.

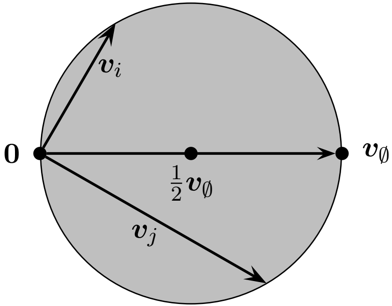
- $\|\boldsymbol{v}_I\|_2^2 = y_I$
- $\|\boldsymbol{v}_\emptyset\|_2^2 = 1$
- $\langle \boldsymbol{v}_I, \boldsymbol{v}_J \rangle \geq 0$



- $\boldsymbol{v}_I$ lies on the **sphere** with radius $\frac{1}{2}$ and center $\frac{1}{2}\boldsymbol{v}_\emptyset$ (since $\|\boldsymbol{v}_I - \frac{1}{2}\boldsymbol{v}_\emptyset\|_2^2 = \|\boldsymbol{v}_I\|_2^2 - 2 \cdot \frac{1}{2}\boldsymbol{v}_I\boldsymbol{v}_\emptyset + \frac{1}{4}\|\boldsymbol{v}_\emptyset\|_2^2 = \frac{1}{4}$)

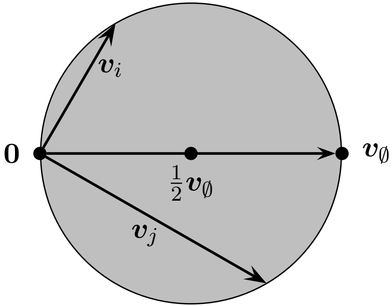
Vector representation for MaxCut

MaxCut: $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = x_{\{i,j\}}$



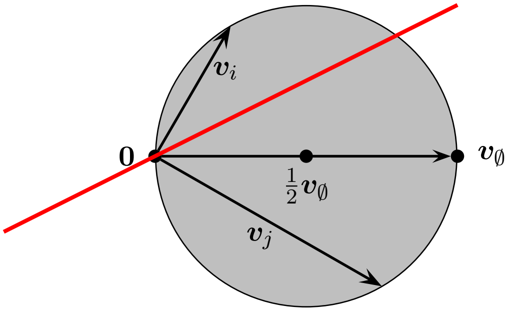
Vector representation for MaxCut

MaxCut: $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = x_{\{i,j\}}$ and $z_{ij} = x_i + x_j - 2x_{\{i,j\}}$



Vector representation for MaxCut

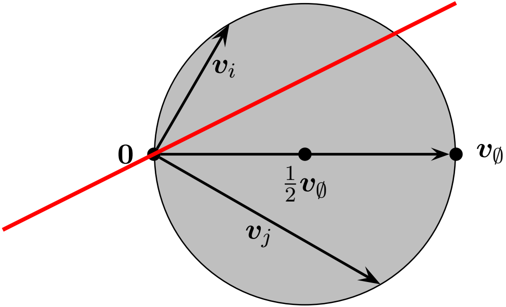
MaxCut: $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = x_{\{i,j\}}$ and $z_{ij} = x_i + x_j - 2x_{\{i,j\}}$



- Now **Hyperplane rounding**?

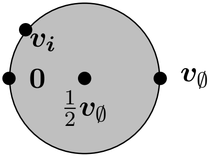
Vector representation for MaxCut

MaxCut: $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = x_{\{i,j\}}$ and $z_{ij} = x_i + x_j - 2x_{\{i,j\}}$

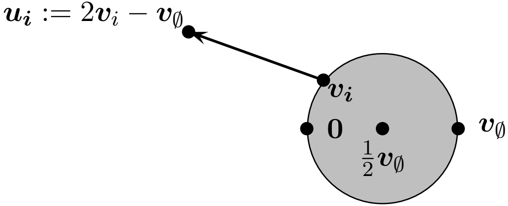


- Now **Hyperplane rounding**?
- **Problem:** Angles are in $[0^o, 90^o]$, $\Pr[\text{cut } (i,j)] \leq \frac{1}{2}$

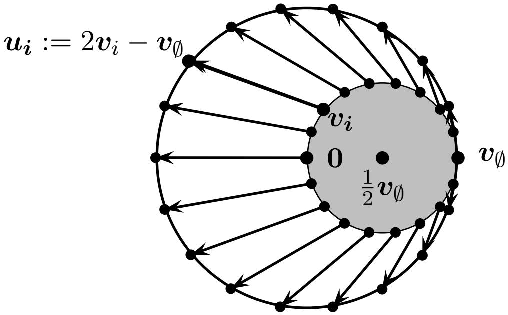
Vector transformation



Vector transformation



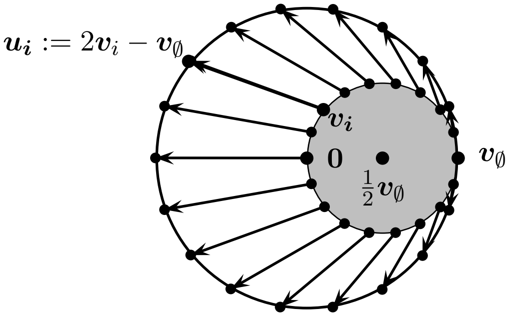
Vector transformation



Observations:

- u_i are unit vectors

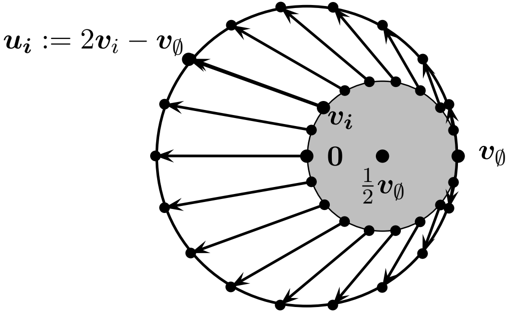
Vector transformation



Observations:

- $\boldsymbol{u}_i$ are unit vectors
- $\langle \boldsymbol{u}_i, \boldsymbol{u}_j \rangle = 1 - 2z_{ij}$ (angles are in $[0^o, 180^o]$)

Vector transformation

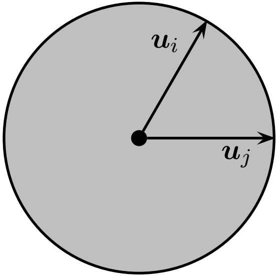


Observations:

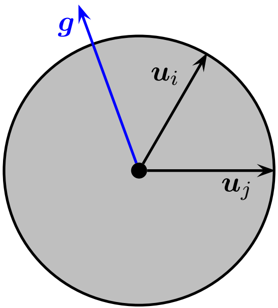
- $\boldsymbol{u}_i$ are unit vectors
- $\langle \boldsymbol{u}_i, \boldsymbol{u}_j \rangle = 1 - 2z_{ij}$ (angles are in $[0^o, 180^o]$)
- $\boldsymbol{u}_i$'s are solutions to **Goemans Williamson SDP**

$$\max \Big\{ \sum_{(i,j)\in E} c_{ij} \cdot \frac{1 - \boldsymbol{u}_i \boldsymbol{u}_j}{2} \mid \|\boldsymbol{u}_i\|_2 = 1 \ \ \forall i \in V \Big\}$$

Hyperplane rounding



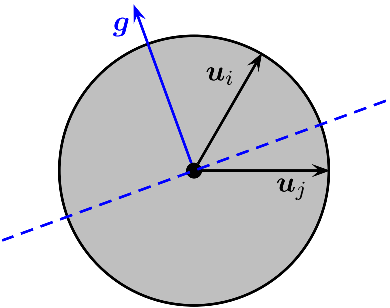
Hyperplane rounding



Algorithm:

(1) Pick a random Gaussian $\boldsymbol{g}$

Hyperplane rounding

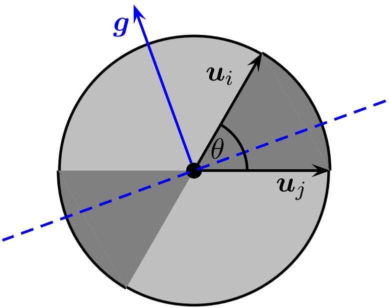


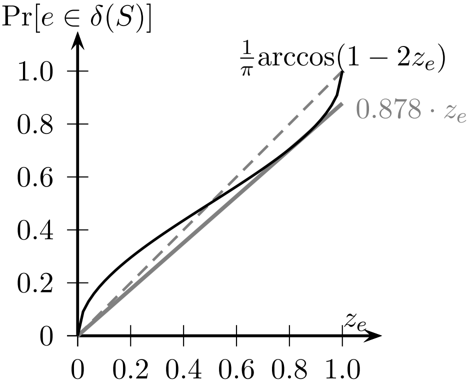
Algorithm:

(1) Pick a random Gaussian $\boldsymbol{g}$

(2) Set $S := \{i \in V \mid \langle \boldsymbol{g}, \boldsymbol{u}_i \rangle \geq 0\}$

Hyperplane rounding





Algorithm:

(1) Pick a random Gaussian $\boldsymbol{g}$

(2) Set $S := \{i \in V \mid \langle \boldsymbol{g}, \boldsymbol{u}_i \rangle \geq 0\}$

Analysis:

$$\Pr[(i,j) \in \delta(S)] = \frac{\text{angle of } \boldsymbol{u}_i \text{ and } \boldsymbol{u}_j}{\pi} = \frac{\arccos(1-2z_e)}{\pi} \geq 0.87 \cdot z_e$$

Theory:

Global Correlation Rounding

Source:

- [Barak, Raghavendra, Steurer '11]
- [Guruswami, Sinop '11]

Global Correlation Rounding

Rand. Var. $X_1, X_2 \in \{0,1\}$ are **uncorrelated / independent**

$$\Leftrightarrow$$

$$\mathrm{Cov}[X_1, X_2] = \Pr[X_1 = X_2 = 1] - \Pr[X_1 = 1] \cdot \Pr[X_2 = 1] \stackrel{!}{=} 0$$

Global Correlation Rounding

$$\text{Rand. Var. } X_1, X_2 \in \{0,1\} \text{ are } \textbf{uncorrelated / independent}$$

$$\Leftrightarrow$$

$$\mathrm{Cov}[X_1, X_2] = \Pr[X_1 = X_2 = 1] - \Pr[X_1 = 1] \cdot \Pr[X_2 = 1] \stackrel{!}{=} 0$$

Theorem

For any $y \in \textsc{Las}_t(K)$ can induce on $\leq O(\frac{1}{\varepsilon^3})$ variables to obtain $y' \in \textsc{Las}_{t-O(1/\varepsilon^3)}(K)$ s.t.

$$\Pr_{i,j \in [n]} \left[\left| y'_i \cdot y'_j - y'_{\{i,j\}} \right| \geq \varepsilon \right] \leq \varepsilon$$

Proof outline

- Consider random variable $X \in \{0,1\}^n$.

Proof outline

- Consider random variable $X \in \{0,1\}^n$.
- Small calculation:

$$\mathop{\mathbb{E}}_{X_2}[\mathrm{Var}[X_1 \mid X_2]] = \ldots \leq \mathrm{Var}[X_1] - 4 \cdot \mathrm{Cov}[X_1, X_2]^2$$

Proof outline

- Consider random variable $X \in \{0,1\}^n$.
- Small calculation:

$$\mathop{\mathbb{E}}_{X_2}[\mathrm{Var}[X_1 \mid X_2]] = \ldots \leq \mathrm{Var}[X_1] - 4 \cdot \mathrm{Cov}[X_1, X_2]^2$$

- Pick a random variable and condition on it $\rightarrow X' \in \{0,1\}^n$

$$\mathop{\mathbb{E}}_{i \in [n]}[\mathrm{Var}[X_i] - \mathrm{Var}[X_i']] \geq 4 \cdot \mathop{\mathbb{E}}_{i,j}[\mathrm{Cov}[X_i, X_j]^2] \overset{\text{if correlated}}{\geq} 4\varepsilon^3$$

Proof outline

- Consider random variable $X \in \{0,1\}^n$.
- Small calculation:

$$\mathop{\mathbb{E}}_{X_2}[\mathrm{Var}[X_1 \mid X_2]] = \ldots \leq \mathrm{Var}[X_1] - 4 \cdot \mathrm{Cov}[X_1, X_2]^2$$

- Pick a random variable and condition on it $\rightarrow X' \in \{0,1\}^n$

$$\mathop{\mathbb{E}}_{i \in [n]}[\mathrm{Var}[X_i] - \mathrm{Var}[X_i']] \geq 4 \cdot \mathop{\mathbb{E}}_{i,j}[\mathrm{Cov}[X_i, X_j]^2] \overset{\text{if correlated}}{\geq} 4\varepsilon^3$$

- Cannot go on for more than $O(\frac{1}{\varepsilon^3})$ rounds

Proof outline

- Consider random variable $X \in \{0,1\}^n$.
- Small calculation:

$$\mathop{\mathbb{E}}_{X_2}[\mathrm{Var}[X_1 \mid X_2]] = \ldots \leq \mathrm{Var}[X_1] - 4 \cdot \mathrm{Cov}[X_1, X_2]^2$$

- Pick a random variable and condition on it $\rightarrow X' \in \{0,1\}^n$

$$\mathop{\mathbb{E}}_{i \in [n]}[\mathrm{Var}[X_i] - \mathrm{Var}[X'_i]] \geq 4 \cdot \mathop{\mathbb{E}}_{i,j}[\mathrm{Cov}[X_i, X_j]^2] \overset{\text{if correlated}}{\geq} 4\varepsilon^3$$

- Cannot go on for more than $O(\frac{1}{\varepsilon^3})$ rounds
- Variance also exists for Lasserre!

Application 4:

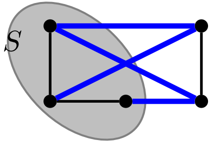
Max Cut in dense Graphs

Source: [de la Vega, Kenyon-Mathieu '95]

PTAS for MaxCut in dense graphs

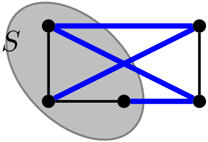
Problem:

- Given $G = (V, E)$ with $|E| \geq \varepsilon n^2$.
- Maximize $|\delta(S)|$



PTAS for MaxCut in dense graphs

Problem:

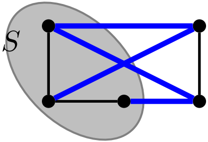


- Given $G = (V, E)$ with $|E| \geq \varepsilon n^2$.
- Maximize $|\delta(S)|$

LP : $$\max \Big\{ \sum_{e \in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\} \; \forall (i,j) \in E \Big\}$$

PTAS for MaxCut in dense graphs

Problem:



- Given $G = (V, E)$ with $|E| \geq \varepsilon n^2$.
- Maximize $|\delta(S)|$

LP :
$$\max\Big\{ \sum_{e\in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\}\ \forall (i,j) \in E \Big\}$$

PTAS:

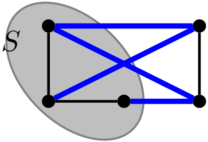
- Compute in $n^{O(1/\varepsilon^4)}$ uncorrelated $(x, z) \in \textsc{Las}_3(K)$:

$$\Pr_{i,j\in V}[|x_i x_j - x_{\{i,j\}}| \geq \varepsilon] \leq \varepsilon^2$$

PTAS for MaxCut in dense graphs

Problem:

- Given $G = (V, E)$ with $|E| \geq \varepsilon n^2$.
- Maximize $|\delta(S)|$



$$\textbf{LP}: \quad \max\left\{ \sum_{e \in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\}\ \forall (i,j) \in E \right\}$$

PTAS:

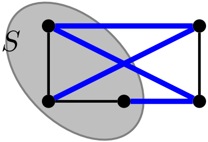
- Compute in $n^{O(1/\varepsilon^4)}$ uncorrelated $(x, z) \in \textsc{Las}_3(K)$:

$$\Pr_{i,j \in V}[|x_i x_j - x_{\{i,j\}}| \geq \varepsilon] \leq \varepsilon^2$$

- **Observe:** $1 - \varepsilon$ fraction of edges $\leq \varepsilon$ correlated.

PTAS for MaxCut in dense graphs

Problem:



- Given $G = (V, E)$ with $|E| \geq \varepsilon n^2$.
- Maximize $|\delta(S)|$

LP : $\max \left\{ \sum_{e \in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\} \; \forall (i,j) \in E \right\}$

PTAS:

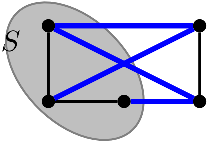
- Compute in $n^{O(1/\varepsilon^4)}$ uncorrelated $(x, z) \in \text{Las}_3(K)$:

$$\Pr_{i,j \in V}[|x_i x_j - x_{\{i,j\}}| \geq \varepsilon] \leq \varepsilon^2$$

- **Observe:** $1 - \varepsilon$ fraction of edges $\leq \varepsilon$ correlated.
- Take $S \subseteq V$ with $\Pr[i \in S] = x_i$ **independently!!**

PTAS for MaxCut in dense graphs

Problem:



- Given $G = (V, E)$ with $|E| \geq \varepsilon n^2$.
- Maximize $|\delta(S)|$

LP : $\max\left\{ \sum_{e \in E} z_e \mid z_{ij} \leq \min\{x_i + x_j, 2 - x_i - x_j\} \ \forall (i,j) \in E \right\}$

PTAS:

- Compute in $n^{O(1/\varepsilon^4)}$ uncorrelated $(x, z) \in \textsc{Las}_3(K)$:

$$\Pr_{i,j \in V}[|x_i x_j - x_{\{i,j\}}| \geq \varepsilon] \leq \varepsilon^2$$

- **Observe:** $1 - \varepsilon$ fraction of edges $\leq \varepsilon$ correlated.
- Take $S \subseteq V$ with $\Pr[i \in S] = x_i$ **independently!!**
- Then

$$\Pr[(i,j) \in \delta(S)] = x_i + x_j - 2x_i x_j \overset{(i,j)\text{ uncor.}}{\approx} x_i + x_j - 2x_{\{i,j\}} = z_e$$

The end

Open problems:

- $f(\varepsilon, k)$ rounds solve Unique Games?
- $O(1)$ rounds give a $O(\log n)$-apx for coloring 3-colorable graphs?
- $f(\varepsilon)$-rounds give PTAS for $P3 \mid \text{prec}, p_j = 1 \mid C_{\max}$?
- $O(1)$ rounds give $(2 - \varepsilon)$-apx for Unrelated Machine Scheduling $Q \mid p_{ij} \mid C_{\max}$?

The end

Open problems:

- $f(\varepsilon, k)$ rounds solve Unique Games?
- $O(1)$ rounds give a $O(\log n)$-apx for coloring 3-colorable graphs?
- $f(\varepsilon)$-rounds give PTAS for $P3 \mid \text{prec}, p_j = 1 \mid C_{\max}$?
- $O(1)$ rounds give $(2 - \varepsilon)$-apx for Unrelated Machine Scheduling $Q \mid p_{ij} \mid C_{\max}$?

Thanks for your attention

Slides and lecture notes can be found under
`http://www-math.mit.edu/~rothvoss/lecturenotes.html`