Max Cut without SDP

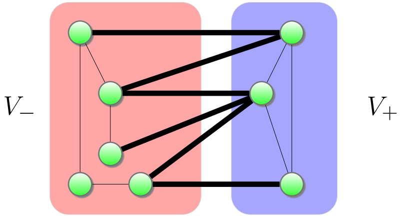
J.A. Soto, Universidad de Chile.

ADGO 2016. Santiago, January 26, 2016

Maximum cut problem

Given a weighted graph $G = (V, E, w)$, with $w\colon E \to \mathbb{R}^+$.
Find bipartition (V_-, V_+) of V that maximizes

$$w(E(V_- \ : \ V_+)) = \sum_{e=uv: u \in V_-, v \in V_+} w(e).$$

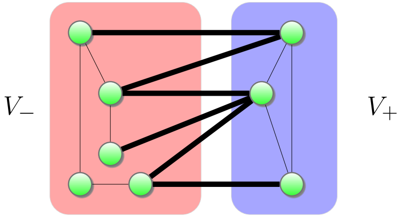


$$\mathrm{maxcut}(G) := \max w(E(V_- \ : \ V_+))$$

Maximum cut problem

Given a weighted graph $G = (V, E, w)$, with $w\colon E \to \mathbb{R}^+$.
Find bipartition (V_-, V_+) of V that maximizes

$$w(E(V_- \ : \ V_+)) = \sum_{e=uv: u \in V_-, v \in V_+} w(e).$$



$$\mathrm{maxcut}(G) := \max w(E(V_- \ : \ V_+))$$

Fact: Max-Cut is NP-hard even for unit weights.

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X : Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X : Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

- Random-cut: Assigning vertices unif. at random.
  0.5 - aproximation.

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X : Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

- Random-cut: Assigning vertices unif. at random.
  0.5 - aproximation.
- Local search (move vertices between X and Y according to local rules).

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X : Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

- Random-cut: Assigning vertices unif. at random.
  0.5 - aproximation.
- Local search (move vertices between X and Y according to local rules).
  0.5 - aproximation.

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X:Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

- Random-cut: Assigning vertices unif. at random.
  0.5 - aproximation.
- Local search (move vertices between X and Y according to local rules).
  0.5 - aproximation.
- Linear Programming approach.

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X:Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

- Random-cut: Assigning vertices unif. at random.
  0.5 - aproximation.
- Local search (move vertices between X and Y according to local rules).
  0.5 - aproximation.
- Linear Programming approach. **0.5 - aproximation**.

Aproximating Max-Cut

A cut (X, Y) is an α-aproximation if
$$w(E(X : Y)) \geq \alpha \cdot \mathrm{maxcut}(G)$$

- Random-cut: Assigning vertices unif. at random.
  0.5 - aproximation.
- Local search (move vertices between X and Y according to local rules).
  0.5 - aproximation.
- Linear Programming approach. **0.5 - aproximation**.
  Any polynomial-size LP for MaxCUT has integrality gap of 2. [Chan, Lee, Ragavendra, Steurer, FOCS 2013]

Approximating Max-Cut

Goemans-Williamson

There is a 0.87856 approximation algorithm for Max-Cut based on semidefinite programming (SDP)

Approximating Max-Cut

Goemans-Williamson

There is a 0.87856 approximation algorithm for Max-Cut based on semidefinite programming (SDP)

- Algorithm is relatively simple (Solve SDP + randomized projection) and can be derandomized.
- Optimal guarantee assuming Unique Games Conjecture.

Approximating Max-Cut

Goemans-Williamson

There is a 0.87856 approximation algorithm for Max-Cut based on semidefinite programming (SDP)

- Algorithm is relatively simple (Solve SDP + randomized projection) and can be derandomized.
- Optimal guarantee assuming Unique Games Conjecture.
- (Approximate solving) SDP is in P, but in general, it is slow.
- Task: Find fast algorithms for Max-Cut achieving an approximation bigger than 0.5

Spectral Partitioning: Recursion Idea

[Trevisan, STOC 2008], [S, 2015 - new analysis]

Idea: Compute a vector $x \in \mathbb{R}^V$ (an eigenvector of certain matrix) to decide the partition recursively.

Spectral Information

For $e = \{i, j\} \in E$, define $M^e, D^e \in \mathbb{R}^{V \times V}$ with quadratic forms

$$x^T M^e x = w_e \cdot (x_i - x_j)^2$$
$$x^T D^e x = w_e \cdot (x_i^2 + x_j^2)$$

If $x \in \{-1, 1\}^V$: $\quad x^T M^e x = \begin{cases} 4w_e & \text{if } e \text{ is cut by } x \\ 0 & \text{if not.} \end{cases}$

$$x^T D^e x = 2w_e$$

Spectral Information

For $e = \{i, j\} \in E$, define $M^e, D^e \in \mathbb{R}^{V \times V}$ with quadratic forms

$$x^T M^e x = w_e \cdot (x_i - x_j)^2$$
$$x^T D^e x = w_e \cdot (x_i^2 + x_j^2)$$

$$\text{If } x \in \{-1,1\}^V : \quad x^T M^e x = \begin{cases} 4w_e & \text{if } e \text{ is cut by } x \\ 0 & \text{if not.} \end{cases}$$
$$x^T D^e x = 2w_e$$

For $M = \sum_{e \in E} M^e, D = \sum_{e \in E} D^e$

$$w(E) = \frac{1}{2}\, x^T D x, \quad \forall x \in \{-1,1\}^V$$
$$\mathrm{maxcut}(G) = \max_{x \in \{-1,1\}^V} \frac{1}{4}\, x^T M x.$$

Important eigenvector

Lemma:

$$\exists x \in \mathbb{R}^V, \|x\|_\infty = 1, \quad \frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)}.$$

Important eigenvector

Lemma:

$$\exists x \in \mathbb{R}^V, \|x\|_\infty = 1, \quad \frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)}.$$

Proof.

- y : eigenvector for maximum eigenvalue of $D^{1/2} M D^{1/2}$.
- y maximizes $\frac{y^T D^{1/2} M D^{1/2} y}{y^T y}$ then $x = D^{-1/2} y$ maximizes $\frac{x^T M x}{x^T D x}$.
- Let $\bar{x} \in \{-1, 1\}^E$ be the indicator of a max cut.

Important eigenvector

Lemma:

$$\exists x \in \mathbb{R}^V, \|x\|_\infty = 1, \quad \frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)}.$$

Proof.

- y : eigenvector for maximum eigenvalue of $D^{1/2} M D^{1/2}$.
- y maximizes $\frac{y^T D^{1/2} M D^{1/2} y}{y^T y}$ then $x = D^{-1/2} y$ maximizes $\frac{x^T M x}{x^T D x}$.
- Let $\bar{x} \in \{-1, 1\}^E$ be the indicator of a max cut. Then

$$\frac{x^T M x}{x^T D x} \geq \frac{\bar{x}^T M \bar{x}}{\bar{x}^T D \bar{x}} = \frac{4\text{maxcut}(G)}{2w(E)}. \quad \square$$

Idea to find a bipartition (V_-, V_+).

- Find x such that $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.

Idea to find a bipartition (V_-, V_+).

- Find x such that $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.
- If x_i is close to 1, put $i \in V_+$.

Idea to find a bipartition (V_-, V_+).

- Find x such that $\frac{x^T Mx}{x^T Dx} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.
- If x_i is close to 1, put $i \in V_+$.
- If x_i is close to -1, put $i \in V_-$.

Idea to find a bipartition (V_-, V_+).

- Find x such that $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.
- If x_i is close to 1, put $i \in V_+$.
- If x_i is close to -1, put $i \in V_-$.
- Else, put i in a set of undecided vertices V_0.

Idea to find a bipartition (V_-, V_+).

- Find x such that $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.
- If x_i is close to 1, put $i \in V_+$.
- If x_i is close to -1, put $i \in V_-$.
- Else, put i in a set of undecided vertices V_0.
- Recurse in $G[V_0]$ to find a bipartition (W_-, W_+) of V_0.

Idea to find a bipartition (V_-, V_+).

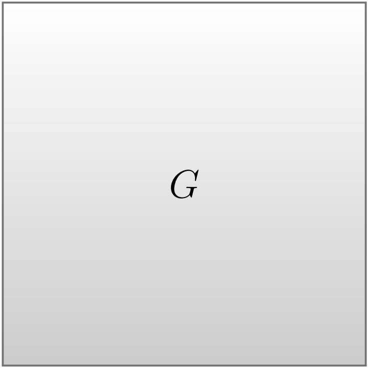
- Find x such that $\frac{x^T M x}{x^T D x} \geq 2 \frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.
- If x_i is close to 1, put $i \in V_+$.
- If x_i is close to -1, put $i \in V_-$.
- Else, put i in a set of undecided vertices V_0.
- Recurse in $G[V_0]$ to find a bipartition (W_-, W_+) of V_0.
- Return the best of $(V_- \cup W_-, V_+ \cup W_+)$ and $(V_- \cup W_+, V_+ \cup W_-)$.

Idea to find a bipartition (V_-, V_+).

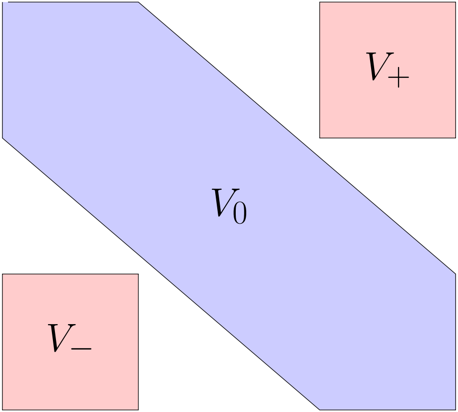
- Find x such that $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$, $\|x\|_\infty = 1$.
- If x_i is close to 1, put $i \in V_+$.
- If x_i is close to -1, put $i \in V_-$.
- Else, put i in a set of undecided vertices V_0.
- Recurse in $G[V_0]$ to find a bipartition (W_-, W_+) of V_0.
- Return the best of $(V_- \cup W_-, V_+ \cup W_+)$ and $(V_- \cup W_+, V_+ \cup W_-)$.

How good is this idea?

Spectral Partitioning: Algorithm $\mathcal{A}(G)$

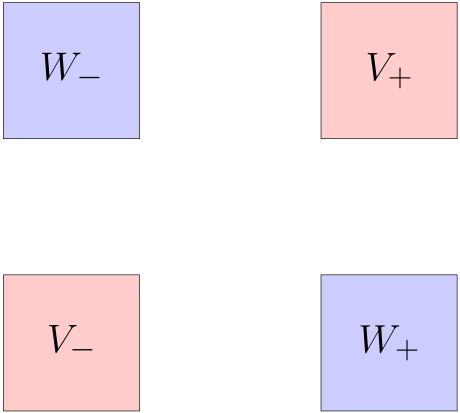


Spectral Partitioning: Algorithm $\mathcal{A}(G)$



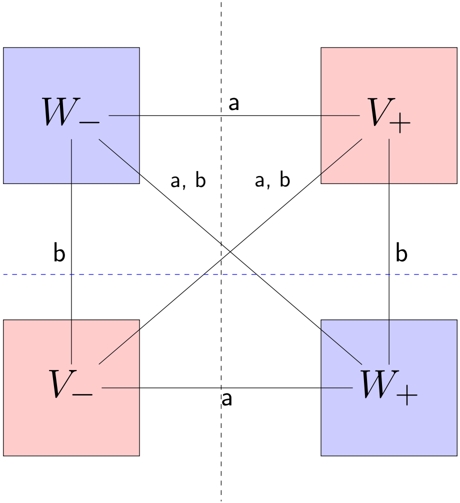
- Use x to find a tripartition of G.

Spectral Partitioning: Algorithm $\mathcal{A}(G)$



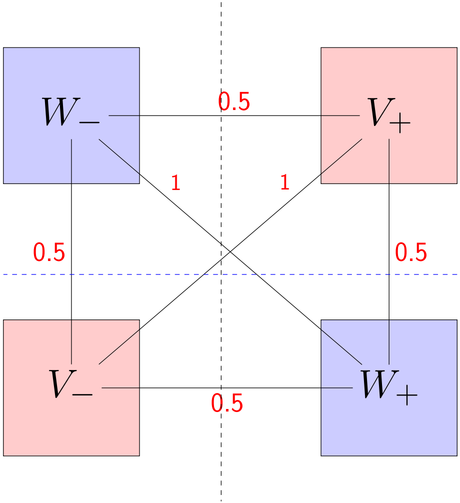
- Use x to find a tripartition of G.
- Recursively partition V_0 in (W_+, W_-).

Spectral Partitioning: Algorithm $\mathcal{A}(G)$



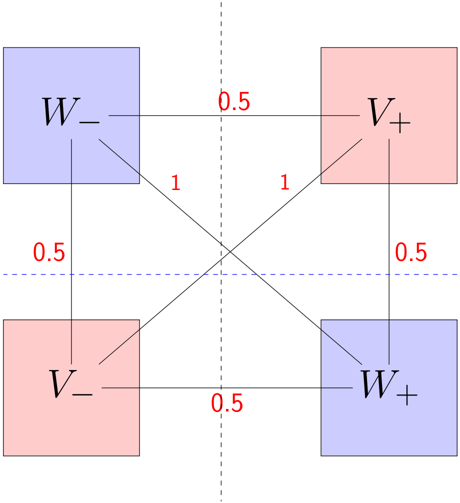
- Use x to find a tripartition of G.
- Recursively partition V_0 in (W_+, W_-).
- Two possible choices
  (a) $(V_- \cup W_-, V_+, W_+)$
  (b) $(V_- \cup W_+, V_+, W_-)$

Spectral Partitioning: Algorithm $\mathcal{A}(G)$



- Use x to find a tripartition of G.
- Recursively partition V_0 in (W_+, W_-).
- Two possible choices
  (a) $(V_- \cup W_-, V_+, W_+)$
  (b) $(V_- \cup W_+, V_+, W_-)$
- Average cut.

Spectral Partitioning: Algorithm $\mathcal{A}(G)$



- Use x to find a tripartition of G.
- Recursively partition V_0 in (W_+, W_-).
- Two possible choices
  (a) $(V_- \cup W_-, V_+, W_+)$
  (b) $(V_- \cup W_+, V_+, W_-)$
- Average cut.

Total cut by $\mathcal{A}(G)$:

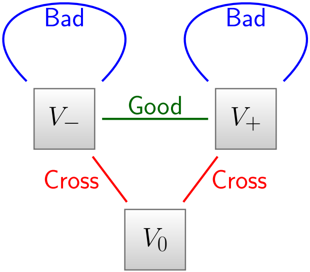
$$\geq w(E(V_+ \ : V_-)) + \frac{1}{2} w(E(V_+ \cup V_- \ : V_0)) + \mathcal{A}(G[V_0])$$

Good tripartitions.

Want $z \in \{-1,0,1\}^V$ maximizing

$$w(E(V_+ : V_-)) + \frac{1}{2} w(E(V_+ \cup V_- : V_0))$$

$$\begin{aligned}
\mathrm{Good}(z) &= w(E(V_- : V_+)) \\
\mathrm{Bad}(z) &= w(E[V_-]) + w(E[V_+]) \\
\mathrm{Cross}(z) &= w(E(V_- \cup V_+ : V_0)) \\
\mathrm{Inc}(z) &= w(E) - w(E[V_0]).
\end{aligned}$$

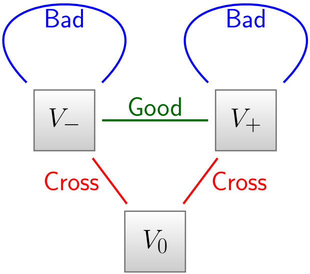


Good tripartitions.

Want $z \in \{-1,0,1\}^V$ maximizing

$$\frac{\mathrm{Good}(z) + \frac{1}{2}\mathrm{Cross}(z)}{\mathrm{Inc}(z)}$$

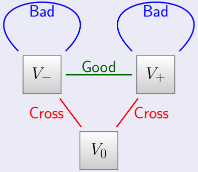
$$\begin{aligned}
\mathrm{Good}(z) &= w(E(V_- : V_+)) \\
\mathrm{Bad}(z) &= w(E[V_-]) + w(E[V_+]) \\
\mathrm{Cross}(z) &= w(E(V_- \cup V_+ : V_0)) \\
\mathrm{Inc}(z) &= w(E) - w(E[V_0]).
\end{aligned}$$



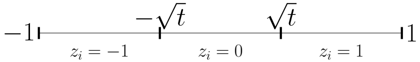
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)}$

need $z \in \{-1, 0, 1\}^V$ with $\frac{\text{Good}(z)+\frac{1}{2}\text{Cross}(z)}{\text{Inc}(z)}$



Choose t uniformly in $[0, 1]$.



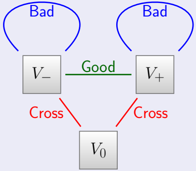
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and

$$\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$$

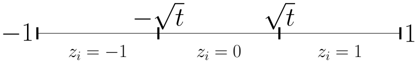
need $z \in \{-1, 0, 1\}^V$ with

$$\frac{\mathrm{Good}(z) + \frac{1}{2}\mathrm{Cross}(z)}{\mathrm{Inc}(z)}$$



$G(i,j) = \Pr(\{i,j\}$ good)
$B(i,j) = \Pr(\{i,j\}$ bad)
$C(i,j) = \Pr(\{i,j\}$ crossing)

Choose t uniformly in $[0,1]$.



Lemma: For all $\beta \in [0,1]$

$$G(i,j) + \beta C(i,j) \geq \beta(1-\beta)(x_i - x_j)^2$$

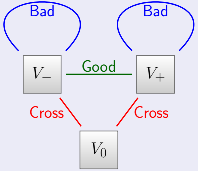
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and

$\frac{x^T M x}{x^T D x} \geq 2 \frac{\text{maxcut}(G)}{w(E)}$

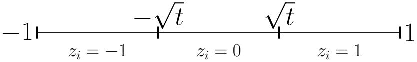
need $z \in \{-1, 0, 1\}^V$ with

$\frac{\text{Good}(z) + \frac{1}{2}\text{Cross}(z)}{\text{Inc}(z)}$



$G(i,j) = \Pr(\{i,j\}$ good)
$B(i,j) = \Pr(\{i,j\}$ bad)
$C(i,j) = \Pr(\{i,j\}$ crossing)

Choose t uniformly in $[0,1]$.



Lemma: For all $\beta \in [0,1]$

$$G(i,j) + \beta C(i,j) \geq \beta(1-\beta)(x_i - x_j)^2$$

Assume $|x_i| \leq |x_j|$.

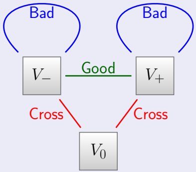
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and

$\frac{x^T Mx}{x^T Dx} \geq 2\frac{\text{maxcut}(G)}{w(E)}$

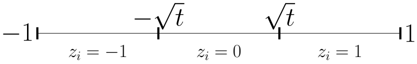
need $z \in \{-1, 0, 1\}^V$ with

$\frac{\text{Good}(z)+\frac{1}{2}\text{Cross}(z)}{\text{Inc}(z)}$



$G(i,j) = \Pr(\{i,j\}$ good)
$B(i,j) = \Pr(\{i,j\}$ bad)
$C(i,j) = \Pr(\{i,j\}$ crossing)

Choose t uniformly in $[0,1]$.



Lemma: For all $\beta \in [0,1]$

$$G(i,j) + \beta C(i,j) \geq \beta(1-\beta)(x_i - x_j)^2$$

Assume $|x_i| \leq |x_j|$.

- $x_i x_j > 0$: $G(i,j) = 0$, $C(i,j) = x_j^2 - x_i^2$.

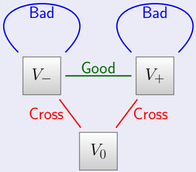
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and

$$\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)}$$

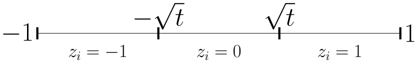
need $z \in \{-1, 0, 1\}^V$ with

$$\frac{\mathrm{Good}(z) + \frac{1}{2}\mathrm{Cross}(z)}{\mathrm{Inc}(z)}$$



$G(i,j) = \Pr(\{i,j\}$ good)
$B(i,j) = \Pr(\{i,j\}$ bad)
$C(i,j) = \Pr(\{i,j\}$ crossing)

Choose t uniformly in $[0,1]$.



Lemma: For all $\beta \in [0,1]$

$$G(i,j) + \beta C(i,j) \geq \beta(1-\beta)(x_i - x_j)^2$$

Assume $|x_i| \leq |x_j|$.

- $x_i x_j > 0$: $G(i,j) = 0$, $C(i,j) = x_j^2 - x_i^2$. Get $\beta(x_j^2 - x_i^2) \geq \beta(1-\beta)(x_i - x_j)^2$.

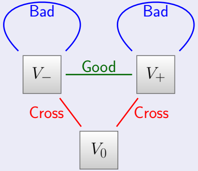
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and

$$\frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)}$$

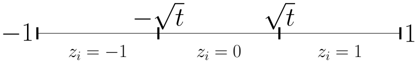
need $z \in \{-1, 0, 1\}^V$ with

$$\frac{\text{Good}(z) + \frac{1}{2}\text{Cross}(z)}{\text{Inc}(z)}$$



$G(i,j) = \Pr(\{i,j\}$ good)
$B(i,j) = \Pr(\{i,j\}$ bad)
$C(i,j) = \Pr(\{i,j\}$ crossing)

Choose t uniformly in $[0,1]$.



Lemma: For all $\beta \in [0,1]$

$$G(i,j) + \beta C(i,j) \geq \beta(1-\beta)(x_i - x_j)^2$$

Assume $|x_i| \leq |x_j|$.

- $x_i x_j > 0$: $G(i,j) = 0$, $C(i,j) = x_j^2 - x_i^2$.
  Get $\beta(x_j^2 - x_i^2) \geq \beta(1-\beta)(x_i - x_j)^2$.
- $x_i x_j < 0$: $G(i,j) = x_i^2$, $C(i,j) = x_j^2 - x_i^2$.

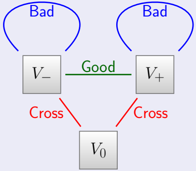
Getting z from x using dependent rounding

Have x with $\|x\|_\infty = 1$ and

$$\frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)}$$

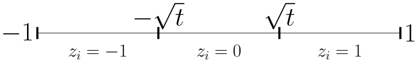
need $z \in \{-1, 0, 1\}^V$ with

$$\frac{\text{Good}(z) + \frac{1}{2}\text{Cross}(z)}{\text{Inc}(z)}$$



$G(i,j) = \Pr(\{i,j\}$ good)
$B(i,j) = \Pr(\{i,j\}$ bad)
$C(i,j) = \Pr(\{i,j\}$ crossing)

Choose t uniformly in $[0,1]$.



Lemma: For all $\beta \in [0,1]$

$$G(i,j) + \beta C(i,j) \geq \beta(1-\beta)(x_i - x_j)^2$$

Assume $|x_i| \leq |x_j|$.

- $x_i x_j > 0$: $G(i,j) = 0$, $C(i,j) = x_j^2 - x_i^2$.
  Get $\beta(x_j^2 - x_i^2) \geq \beta(1-\beta)(x_i - x_j)^2$.
- $x_i x_j < 0$: $G(i,j) = x_i^2$, $C(i,j) = x_j^2 - x_i^2$.
  Get $\beta x_j^2 + (1-\beta)x_i^2 \geq \beta(1-\beta)(x_i - x_j)^2$.

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\mathbb{E}[\mathrm{Good}(z)] + \beta\mathbb{E}[\mathrm{Cross}(z)] \qquad = \sum_{ij\in E} w_{ij}(G(i,j) + \beta C(i,j))$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\mathbb{E}[\text{Good}(z)] + \beta\mathbb{E}[\text{Cross}(z)] = \sum_{ij \in E} w_{ij}(G(i,j) + \beta C(i,j))$$

$$\geq \sum_{ij \in E} \beta(1-\beta) w_{ij}(x_i - x_j)^2$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\begin{aligned} \mathbb{E}[\text{Good}(z)] + \beta\mathbb{E}[\text{Cross}(z)] &= \sum_{ij\in E} w_{ij}(G(i,j) + \beta C(i,j)) \\ \geq \sum_{ij\in E} \beta(1-\beta) w_{ij}(x_i - x_j)^2 &= \beta(1-\beta)x^T M x \end{aligned}$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\begin{aligned}
\mathbb{E}[\mathrm{Good}(z)] + \beta\mathbb{E}[\mathrm{Cross}(z)] \qquad & = \sum_{ij\in E} w_{ij}(G(i,j) + \beta C(i,j)) \\
\geq \sum_{ij\in E} \beta(1-\beta) w_{ij}(x_i - x_j)^2 \qquad & = \beta(1-\beta)x^T M x \\
\geq 2(1-\varepsilon)\beta(1-\beta)x^T D x &
\end{aligned}$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\text{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$
\begin{aligned}
\mathbb{E}[\text{Good}(z)] + \beta\mathbb{E}[\text{Cross}(z)] &= \sum_{ij \in E} w_{ij}(G(i,j) + \beta C(i,j)) \\
\geq \sum_{ij\in E} \beta(1-\beta) w_{ij}(x_i - x_j)^2 &= \beta(1-\beta)x^T M x \\
\geq 2(1-\varepsilon)\beta(1-\beta)x^T D x &= 2(1-\varepsilon)\beta(1-\beta)\sum_{ij\in E} w_{ij}(x_i^2 + x_j^2)
\end{aligned}
$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\begin{aligned}
\mathbb{E}[\mathrm{Good}(z)] + \beta\mathbb{E}[\mathrm{Cross}(z)] &= \sum_{ij\in E} w_{ij}(G(i,j) + \beta C(i,j)) \\
\geq \sum_{ij\in E} \beta(1-\beta) w_{ij}(x_i - x_j)^2 &= \beta(1-\beta)x^T M x \\
\geq 2(1-\varepsilon)\beta(1-\beta)x^T D x &= 2(1-\varepsilon)\beta(1-\beta)\sum_{ij\in E} w_{ij}(x_i^2 + x_j^2) \\
= 2(1-\varepsilon)\beta(1-\beta)\mathbb{E}[2\mathrm{Inc}(z) - \mathrm{Cross}(z)].
\end{aligned}$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\mathbb{E}[\mathrm{Good}(z)] + \beta\mathbb{E}[\mathrm{Cross}(z)] \geq 2(1-\varepsilon)\beta(1-\beta)\mathbb{E}[2\mathrm{Inc}(z) - \mathrm{Cross}(z)].$$

How good is z?

Suppose $\|x\|_\infty = 1$ and $\frac{x^T M x}{x^T D x} \geq 2 \frac{\mathrm{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, with $\varepsilon \leq 1/2$ and z is chosen (at random) as before.

$$\mathbb{E}[\mathrm{Good}(z)] + \beta \mathbb{E}[\mathrm{Cross}(z)] \geq 2(1-\varepsilon)\beta(1-\beta)\mathbb{E}[2\mathrm{Inc}(z) - \mathrm{Cross}(z)].$$

Using this and

$$\mathbb{E}[\mathrm{Good}(z)] + \mathbb{E}[\mathrm{Cross}(z)] \leq \mathbb{E}[\mathrm{Inc}(z)],$$

we can optimize β to maximize

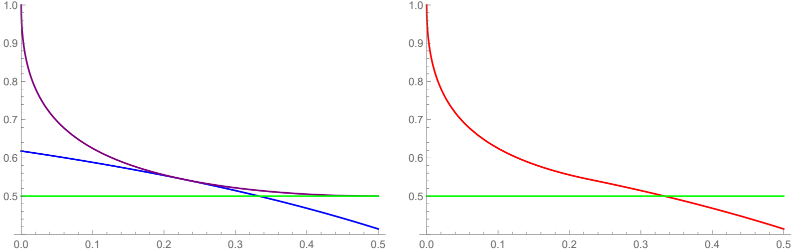
$$\frac{\mathbb{E}[\mathrm{Good}(z)] + \frac{1}{2}\mathbb{E}[\mathrm{Cross}(z)]}{\mathbb{E}[\mathrm{Inc}(z)]}.$$

Theorem

Let $\|x\|_\infty = 1$ *and* $\frac{x^T M x}{x^T D x} \geq 2\frac{\mathrm{maxcut}(G)}{w(E)} = 2(1-\varepsilon)$, *with* $\varepsilon \leq 1/2$.
Can find $z \in \{-1,0,1\}^V \setminus \{0\}^V$ *with* $\rho(z) = \frac{\mathrm{Good}(z)+\frac{1}{2}\mathrm{Cross}(z)}{\mathrm{Inc}(z)}$ *satisfying*

$$\rho(z) \geq f(\varepsilon) := \begin{cases} \frac{-1+\sqrt{4\varepsilon^2-8\varepsilon+5}}{2(1-\varepsilon)}, & \textit{if } \varepsilon \geq \varepsilon_0 \\ \frac{1}{1+2\sqrt{\varepsilon(1-\varepsilon)}}, & \textit{if } \varepsilon \leq \varepsilon_0 \end{cases}$$

where $\varepsilon_0 \approx 0.22815$ *is the unique value that makes both expressions equal.*

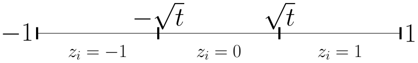


Deterministic rounding

In reality, optimizing β yields a random vector z such that

$$\frac{\mathbb{E}[\mathrm{Good}(z)] + \frac{1}{2}\mathbb{E}[\mathrm{Cross}(z)]}{\mathbb{E}[\mathrm{Inc}(z)]} \geq f(\varepsilon).$$

But recall how z was constructed: We chose t uniformly in $[0, 1]$, and checked x_i.

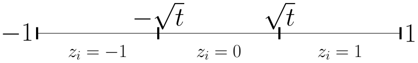


Deterministic rounding

In reality, optimizing β yields a random vector z such that

$$\frac{\mathbb{E}[\text{Good}(z)] + \frac{1}{2}\mathbb{E}[\text{Cross}(z)]}{\mathbb{E}[\text{Inc}(z)]} \geq f(\varepsilon).$$

But recall how z was constructed: We chose t uniformly in $[0, 1]$, and checked x_i.



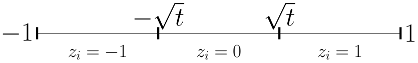
Only relevant values for t: $\{x_i^2 \colon i \in V\}$.

Deterministic rounding

In reality, optimizing β yields a random vector z such that

$$\frac{\mathbb{E}[\mathrm{Good}(z)] + \frac{1}{2}\mathbb{E}[\mathrm{Cross}(z)]}{\mathbb{E}[\mathrm{Inc}(z)]} \geq f(\varepsilon).$$

But recall how z was constructed: We chose t uniformly in $[0, 1]$, and checked x_i.



Only relevant values for t: $\{x_i^2 \colon i \in V\}$. Compute the n different possibilities for z and choose the one that maximizes

$$\rho(z) = \frac{\mathrm{Good}(z) + \frac{1}{2}\mathrm{Cross}(z)}{\mathrm{Inc}(z)}.$$

Complete algorithm

Given weighted graph $G(V, E, w)$.
Algorithm $\mathcal{A}(G)$ does the following.

- Compute x, $\|x\|_\infty = 1$ maximizing $\frac{x^T M x}{x^T D x}$.
- Use deterministic rounding to get $z \in \{-1, 0, 1\}^V \setminus \{0\}^V$ by Theorem.
- If $\rho(z) < 1/2$, return 0.5-approximation (V_-, V_+).
- Else, let (V_-, V_0, V_+) be the tripartition induced by z.
  - If $V_0 = \emptyset$ return (V_-, V_+).
  - Else compute $(W_-, W_+) = \mathcal{A}(G[V_0])$ and return the best of $(V_- \cup W_-, V_+ \cup W_+)$ and $(V_- \cup W_+, V_+ \cup W_-)$.

Complete algorithm

Given weighted graph $G(V, E, w)$.
Algorithm $\mathcal{A}(G)$ does the following.

- Compute x, $\|x\|_\infty = 1$ maximizing $\frac{x^T M x}{x^T D x}$.
- Use deterministic rounding to get $z \in \{-1, 0, 1\}^V \setminus \{0\}^V$ by Theorem.
- If $\rho(z) < 1/2$, return 0.5-approximation (V_-, V_+).
- Else, let (V_-, V_0, V_+) be the tripartition induced by z.
  - If $V_0 = \emptyset$ return (V_-, V_+).
  - Else compute $(W_-, W_+) = \mathcal{A}(G[V_0])$ and return the best of $(V_- \cup W_-, V_+ \cup W_+)$ and $(V_- \cup W_+, V_+ \cup W_-)$.

What is the guarantee?

Theorem (S.)

If $\mathrm{maxcut}(G) \geq (1-\varepsilon)w(E)$, the algorithm returns a cut (V_-, V_+) with

$$\frac{w(V_- : V_+)}{w(E)} \geq \int_0^1 \max\left(1/2, f(\varepsilon/r)\right) dr.$$

Theorem (S.)

If $\mathrm{maxcut}(G) \geq (1-\varepsilon)w(E)$, the algorithm returns a cut (V_-, V_+) with

$$\frac{w(V_- : V_+)}{w(E)} \geq \int_0^1 \max\left(1/2, f(\varepsilon/r)\right) dr.$$

Let $G = G_0, \ldots, G_T$ be the graphs on each recursive call.
$G_t = (V_t, E_t)$, $\delta_t = w(E_t)/w(E) < 1$.

- Note that $\mathrm{maxcut}(G_t) \geq (1 - \varepsilon/\delta_t)w(E_t)$.

Theorem (S.)

If $\mathrm{maxcut}(G) \geq (1-\varepsilon)w(E)$, the algorithm returns a cut (V_-, V_+) with

$$\frac{w(V_- : V_+)}{w(E)} \geq \int_0^1 \max\left(1/2, f(\varepsilon/r)\right) dr.$$

Let $G = G_0, \ldots, G_T$ be the graphs on each recursive call.
$G_t = (V_t, E_t)$, $\delta_t = w(E_t)/w(E) < 1$.

- Note that $\mathrm{maxcut}(G_t) \geq (1 - \varepsilon/\delta_t)w(E_t)$.

$$\begin{aligned} w(E_t) - w(\mathrm{maxcut}(G_t)) = \mathrm{Bad}(\bar{x}_t) \leq \mathrm{Bad}(\bar{x}) &= w(E) - w(\mathrm{maxcut}(G)) \\ &= w(E)\varepsilon = w(E_t)\varepsilon/\delta_t \end{aligned}$$

Theorem (S.)

If $\mathrm{maxcut}(G) \geq (1-\varepsilon)w(E)$, the algorithm returns a cut (V_-, V_+) with

$$\frac{w(V_- : V_+)}{w(E)} \geq \int_0^1 \max\left(1/2, f(\varepsilon/r)\right) dr.$$

Let $G = G_0, \ldots, G_T$ be the graphs on each recursive call.
$G_t = (V_t, E_t)$, $\delta_t = w(E_t)/w(E) < 1$.

- Note that $\mathrm{maxcut}(G_t) \geq (1 - \varepsilon/\delta_t)w(E_t)$.
- Weight cut in $E_t \setminus E_{t+1}$ is at least $\max\left(1/2, f(\varepsilon/r)\right) \cdot w(E_t \setminus E_{t+1})$.

Theorem (S.)

If $\mathrm{maxcut}(G) \geq (1-\varepsilon)w(E)$, the algorithm returns a cut (V_-, V_+) with

$$\frac{w(V_- : V_+)}{w(E)} \geq \int_0^1 \max\left(1/2, f(\varepsilon/r)\right) dr.$$

Let $G = G_0, \ldots, G_T$ be the graphs on each recursive call.
$G_t = (V_t, E_t)$, $\delta_t = w(E_t)/w(E) < 1$.

- Note that $\mathrm{maxcut}(G_t) \geq (1 - \varepsilon/\delta_t)w(E_t)$.
- Weight cut in $E_t \setminus E_{t+1}$ is at least $\max\left(1/2, f(\varepsilon/r)\right) \cdot w(E_t \setminus E_{t+1})$.

$$\begin{aligned}\frac{w(V_- : V_+)}{w(E)} &\geq \sum_{t=0}^{T} \max\left(1/2, f(\varepsilon/r)\right) \cdot (\delta_t - \delta_{t+1}) \\ &\geq \sum_{i=0}^{T} \int_{\delta_{t+1}}^{\delta_t} \max\left(1/2, f(\varepsilon/r)\right) dr.\end{aligned}$$

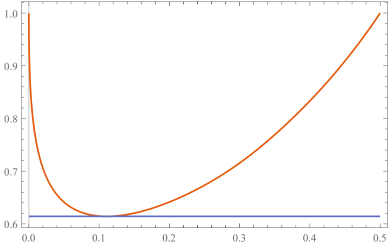
Aproximation Guarantee

Corollary:

Let $0 \leq \varepsilon \leq 1/2$ be such that $\operatorname{maxcut}(G) \geq (1-\varepsilon)w(E)$. The algorithm returns a cut with approximation guarantee equal to

$$F(\varepsilon) := \frac{1}{(1-\varepsilon)} \int_0^1 \max\left(1/2, f(\varepsilon/r)\right) dr.$$

In particular, the guarantee is at least 0.614247



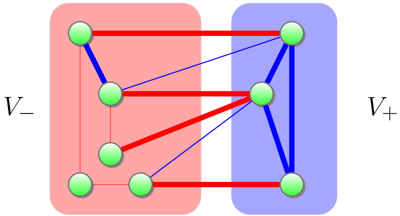
Extensions

We get same guarantees for the following variant:

Maximum colored cut problem

Given a weighted graph $G = (V, E = R \cup B, w)$, with $w\colon E \to \mathbb{R}^+$.
Find bipartition (V_-, V_+) of V that maximizes

$$\underbrace{w(R(V_- \ : \ V_+))}_{\text{red cut}} + \underbrace{w(B[V_-]) + w(B[V_+])}_{\text{blue uncut}}.$$



Thanks!
Questions?