Low stretch trees.

Another cool approach:

Low stretch trees.

Another cool approach:
Low stretch spanning tree.

Low stretch trees.

Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Low stretch trees.

Another cool approach:
- Low stretch spanning tree.
  - Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.
  - Example: expander.

Low stretch trees.

Another cool approach:
- Low stretch spanning tree.
  - Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.
  - Example: expander. Any short tree.

Low stretch trees.

Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

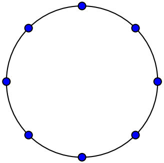
Low stretch trees.

Another cool approach:

- Low stretch spanning tree.
  - Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.
  - Example: expander. Any short tree. Diameter is $O(\log n)$.
  - Example: cycle.

Low stretch trees.

Another cool approach:
  Low stretch spanning tree.
    Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.
    Example: expander. Any short tree. Diameter is $O(\log n)$.
    Example: cycle.



Low stretch trees.

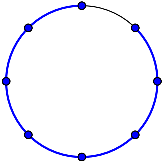
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e = (u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Low stretch trees.

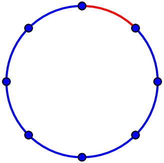
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

Low stretch trees.

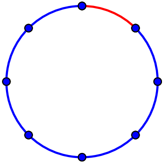
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.

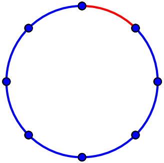


Distance 1 goes to $n-1$!

Low stretch trees.

Another cool approach:

- Low stretch spanning tree.
  - Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.
  - Example: expander. Any short tree. Diameter is $O(\log n)$.
  - Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge

Low stretch trees.

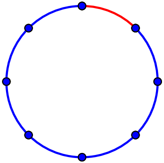
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Low stretch trees.

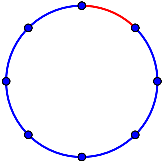
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1$

Low stretch trees.

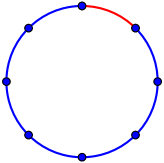
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1)$

Low stretch trees.

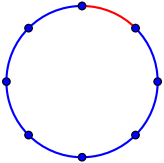
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx$

Low stretch trees.

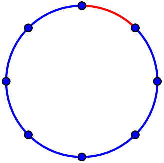
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx 2$

Low stretch trees.

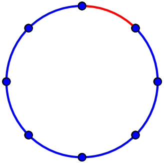
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e = (u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx 2$

In general: $\tilde{O}(m)$

Low stretch trees.

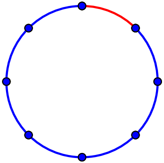
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e = (u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx 2$

In general: $\tilde{O}(m) = O(m \log n \log\log n)$.

Low stretch trees.

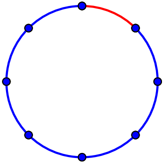
Another cool approach:

Low stretch spanning tree.

Spanning tree T with small $\ell_T(u,v)$ for average edge $e=(u,v)$.

Example: expander. Any short tree. Diameter is $O(\log n)$.

Example: cycle.



Distance 1 goes to $n-1$!

For cycle, remove a random edge get a tree.

Average Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx 2$

In general: $\tilde{O}(m) = O(m \log n \log\log n)$.

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u, v).$$

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v)$.
(Assume resistances $r(e) = 1$.)

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v)$.
(Assume resistances $r(e) = 1$.)

KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u, v)$.
(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$

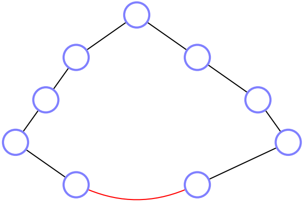
KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$
(Assume resistances $r(e) = 1.$)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v) - \phi(u,v)}{\ell_T(e)+1}$.

KOSZ

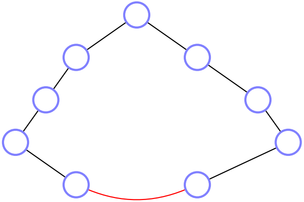
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$

(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)} (f(u,v) - \phi(u,v))^2$

KOSZ

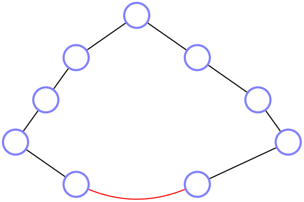
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$

(Assume resistances $r(e) = 1.$)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)}(f(u,v) - \phi(u,v))^2$

Increases a bit on $\ell_T(e)$ (no violation) edges)

KOSZ

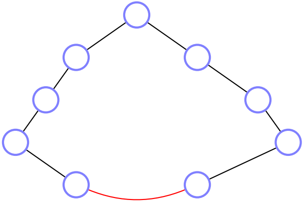
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$

(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v) - \phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)} (f(u,v) - \phi(u,v))^2$

Increases a bit on $\ell_T(e)$ (no violation) edges)

reduces a lot on e since large violation

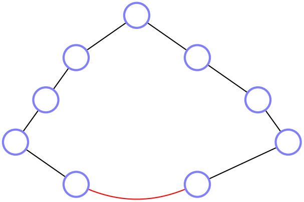
KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v)$.
(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.
Reduces $\sum_{e=(u,v)}(f(u,v) - \phi(u,v))^2$
Increases a bit on $\ell_T(e)$ (no violation) edges)
reduces a lot on e since large violation

Decrease on one edge dominate increase on many edges?

KOSZ

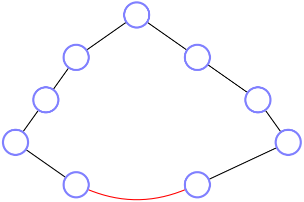
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u, v).$$

(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u, v)$

Change flow by $\frac{f(u,v) - \phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)} (f(u, v) - \phi(u, v))^2$

Increases a bit on $\ell_T(e)$ (no violation) edges)

reduces a lot on e since large violation

Decrease on one edge dominate increase on many edges?

No violation on many, large violation on one.

KOSZ

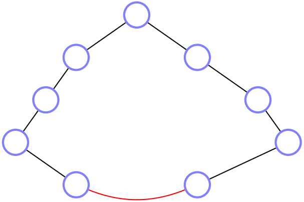
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$

(Assume resistances $r(e) = 1.$)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)}(f(u,v) - \phi(u,v))^2$

Increases a bit on $\ell_T(e)$ (no violation) edges)

reduces a lot on e since large violation

Decrease on one edge dominate increase on many edges?

No violation on many, large violation on one. Quadratic.

KOSZ

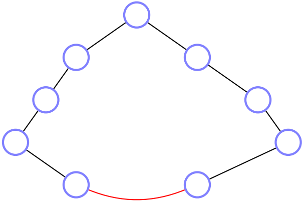
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$

(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)}(f(u,v) - \phi(u,v))^2$

Increases a bit on $\ell_T(e)$ (no violation) edges)

reduces a lot on e since large violation

Decrease on one edge dominate increase on many edges?

No violation on many, large violation on one. Quadratic.

Increase: $\ell \times (\varepsilon^2)$, decrease $1^2 - (1-\varepsilon)^2 = -2\varepsilon + \varepsilon^2$.

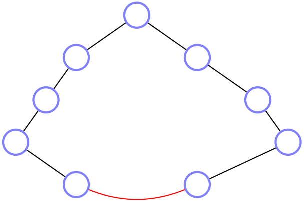
KOSZ

Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v)$.
(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.
Reduces $\sum_{e=(u,v)}(f(u,v) - \phi(u,v))^2$
Increases a bit on $\ell_T(e)$ (no violation) edges)
reduces a lot on e since large violation

Decrease on one edge dominate increase on many edges?
No violation on many, large violation on one. Quadratic.
Increase: $\ell \times (\varepsilon^2)$, decrease $1^2 - (1-\varepsilon)^2 = -2\varepsilon + \varepsilon^2$.
Change: $-2\varepsilon + (\ell+1)\varepsilon^2$.

KOSZ

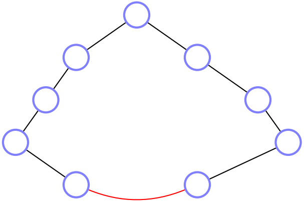
Laplacian systems $\equiv$ to routing electrical flow with sources and sinks.

Solve $\phi(u)$ and currents, $f(\cdot)$, on spanning tree.

$$f(e) = (\phi(u) - \phi(v))/r(e) = \phi(u) - \phi(v) = \phi(u,v).$$

(Assume resistances $r(e) = 1$.)

Pick non-tree edge with probability $\propto \ell_T(e) + 1$



$f(e) \neq \phi(u,v)$

Change flow by $\frac{f(u,v)-\phi(u,v)}{\ell_T(e)+1}$.

Reduces $\sum_{e=(u,v)}(f(u,v) - \phi(u,v))^2$

Increases a bit on $\ell_T(e)$ (no violation) edges)

reduces a lot on e since large violation

Decrease on one edge dominate increase on many edges?

No violation on many, large violation on one. Quadratic.

Increase: $\ell \times (\varepsilon^2)$, decrease $1^2 - (1-\varepsilon)^2 = -2\varepsilon + \varepsilon^2$.

Change: $-2\varepsilon + (\ell+1)\varepsilon^2$. $\quad \varepsilon = \frac{1}{\ell+1} \implies -\varepsilon$ decrease.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$

Analysis Idea

Total violation: $\sum_{e=(u,v)} (f(e) - \phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e) - \phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e) - \phi(u,v))^2$
$\alpha = \Theta(1/\ell)$

Analysis Idea

Total violation: $\sum_{e=(u,v)} (f(e) - \phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e) - \phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e) - \phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha=\Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$$

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality:

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $=\tilde{O}(m)$

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities:

Analysis Idea

Total violation: $\sum_{e=(u,v)} (f(e) - \phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e) - \phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e) - \phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e) - \phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)} (f(e) - \phi(u,v))^2$.

The constant of proportionality: $\sum_e (\ell_T(e) + 1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e + 1}{\sum_e (\ell_T(e) + 1)}$

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $=\tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$

tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$

$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$

$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$

Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

$\tilde{O}(m)$ iterations halves error.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

$\tilde{O}(m)$ iterations halves error.

Analysis Idea

Total violation: $\sum_{e=(u,v)} (f(e) - \phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e) - \phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e) - \phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e) - \phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e) - \phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)} (f(e) - \phi(u,v))^2$.

The constant of proportionality: $\sum_e (\ell_T(e) + 1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e + 1}{\sum_e (\ell_T(e) + 1)}$

$(1 - \tilde{\Omega}(1/m))$ expected reduction.

$\tilde{O}(m)$ iterations halves error.

Again: used exact solve on tree to speed up iterative method.

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

$\tilde{O}(m)$ iterations halves error.

Again: used exact solve on tree to speed up iterative method.
Shout out to "Algebra".

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

$\tilde{O}(m)$ iterations halves error.

Again: used exact solve on tree to speed up iterative method.
Shout out to “Algebra”.

Update efficiently?

Analysis Idea

Total violation: $\sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

Edge w/stretch ℓ selected: reduce violation by $\approx \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Non-tree edge reduces $\propto \alpha(f(e)-\phi(u,v))^2$
tree edges increase $\propto \ell(\alpha^2(f(e)-\phi(u,v))^2$
$\alpha = \Theta(1/\ell) \implies$ reduction $\propto \frac{1}{\ell}(f(e)-\phi(u,v))^2$.

Selected $\propto \ell$.

$(f(e)-\phi(u,v))^2 \times \frac{1}{\ell} \times \ell$
$\implies$ expected reduction $\propto \sum_{e=(u,v)}(f(e)-\phi(u,v))^2$.

The constant of proportionality: $\sum_e(\ell_T(e)+1)$. $= \tilde{O}(m)$
Probabilities: $\approx \frac{\ell_e+1}{\sum_e(\ell_T(e)+1)}$

$(1-\tilde{\Omega}(1/m))$ expected reduction.

$\tilde{O}(m)$ iterations halves error.

Again: used exact solve on tree to speed up iterative method.
Shout out to "Algebra".

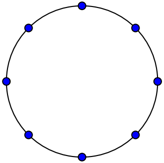
Update efficiently? Update on paths, path decomposition of trees.

Poll

What happens on the cycle?

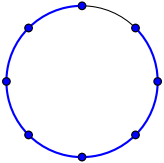
Poll

What happens on the cycle?



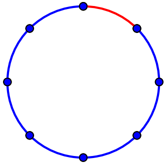
Poll

What happens on the cycle?



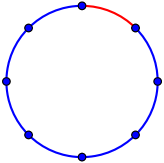
Poll

What happens on the cycle?



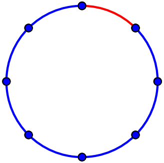
Poll

What happens on the cycle?



Poll

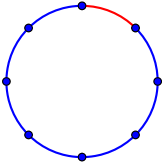
What happens on the cycle?



Keep choosing non-tree edge.

Poll

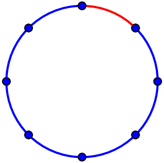
What happens on the cycle?



Keep choosing non-tree edge. $\ell \approx m \implies \Omega(1/m)$ progress

Poll

What happens on the cycle?

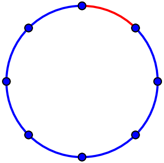


Keep choosing non-tree edge. $\ell \approx m \implies \Omega(1/m)$ progress

$\tilde{O}(m)$ iterations.

Poll

What happens on the cycle?



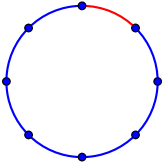
Keep choosing non-tree edge. $\ell \approx m \implies \Omega(1/m)$ progress

$\tilde{O}(m)$ iterations.

Need Data structure to update flow along path.

Poll

What happens on the cycle?



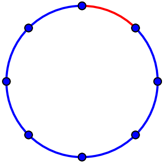
Keep choosing non-tree edge. $\ell \approx m \implies \Omega(1/m)$ progress

$\tilde{O}(m)$ iterations.

Need Data structure to update flow along path.
    Exercise: Use Binary tree on path.

Poll

What happens on the cycle?



Keep choosing non-tree edge. $\ell \approx m \implies \Omega(1/m)$ progress

$\tilde{O}(m)$ iterations.

Need Data structure to update flow along path.

Exercise: Use Binary tree on path.

$O(\log n)$ time per update.

Electrical Flow and Laplacian Systems.

A graph $G = (V, E)$.

Electrical Flow and Laplacian Systems.

A graph $G = (V, E)$.
Circuit: nodes V, resistors E, value 1 (for today.)

Electrical Flow and Laplacian Systems.

A graph $G = (V, E)$.
Circuit: nodes V, resistors E, value 1 (for today.)

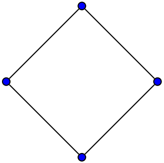
Given $\chi : V \to \Re$
Find flow that routes χ and minimizes
$\sum_e f(e)^2$.

Electrical Flow and Laplacian Systems.

A graph $G = (V, E)$.
Circuit: nodes V, resistors E, value 1 (for today.)

Given $\chi : V \to \Re$
Find flow that routes χ and minimizes
$\sum_e f(e)^2$.

+1

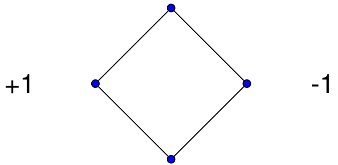


-1

Electrical Flow and Laplacian Systems.

A graph $G = (V, E)$.
Circuit: nodes V, resistors E, value 1 (for today.)

Given $\chi : V \rightarrow \Re$
Find flow that routes χ and minimizes
$\sum_e f(e)^2$.

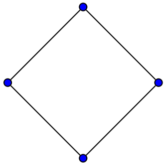


Claim: Minimizer is electrical flow.

Electrical Flow and Laplacian Systems.

A graph $G = (V, E)$.
Circuit: nodes V, resistors E, value 1 (for today.)

Given $\chi : V \rightarrow \Re$
Find flow that routes χ and minimizes
$\sum_e f(e)^2$.

+1  -1

Claim: Minimizer is electrical flow.

Flow corresponds to flow induced by a set of potentials.

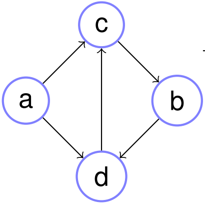
Some Matrices.

Given $G = (V, E)$, arbitrarily orient edges.

Some Matrices.

Given $G = (V, E)$, arbitrarily orient edges.

$$B_{v,e} = \begin{cases} -1 & e = (u,v) \\ 1 & e = (v,u) \\ 0 & \text{otherwise} \end{cases} \qquad L_{u,v} = \begin{cases} d & u = v \\ -1 & (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$



B

	a	b	c	d
(a,b)	1	-1	0	0
(a,c)	1	0	-1	0
(c,d)	0	0	1	-1
(d,b)	0	-1	0	1
(b,c)	0	1	-1	1

L

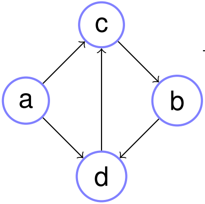
	a	b	c	d
a	2	-1	-1	0
b	-1	2	0	-1
c	-1	-1	3	-1
d	-1	-1	-1	3

Some Matrices.

Given $G = (V, E)$, arbitrarily orient edges.

$$B_{v,e} = \begin{cases} -1 & e = (u,v) \\ 1 & e = (v,u) \\ 0 & \text{otherwise} \end{cases}$$

$$L_{u,v} = \begin{cases} d & u = v \\ -1 & (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$



B

	a	b	c	d
(a,b)	1	-1	0	0
(a,c)	1	0	-1	0
(c,d)	0	0	1	-1
(d,b)	0	-1	0	1
(b,c)	0	1	-1	1

L

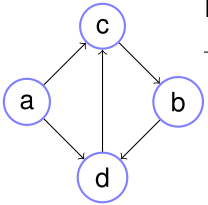
	a	b	c	d
a	2	-1	-1	0
b	-1	2	0	-1
c	-1	-1	3	-1
d	-1	-1	-1	3

Fun facts: $\mathbf{f} \in \Re^{|E|}$

Some Matrices.

Given $G = (V, E)$, arbitrarily orient edges.

$$B_{v,e} = \begin{cases} -1 & e = (u,v) \\ 1 & e = (v,u) \\ 0 & \text{otherwise} \end{cases} \qquad L_{u,v} = \begin{cases} d & u = v \\ -1 & (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$



B

	a	b	c	d
(a,b)	1	-1	0	0
(a,c)	1	0	-1	0
(c,d)	0	0	1	-1
(d,b)	0	-1	0	1
(b,c)	0	1	-1	1

L

	a	b	c	d
a	2	-1	-1	0
b	-1	2	0	-1
c	-1	-1	3	-1
d	-1	-1	-1	3

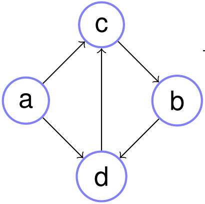
Fun facts: $\mathbf{f} \in \Re^{|E|}$

$[B^T f]_u = \sum_{e=(u,v)} f_e - \sum_{e=(v,u)} f_e$

Some Matrices.

Given $G = (V, E)$, arbitrarily orient edges.

$$B_{v,e} = \begin{cases} -1 & e = (u,v) \\ 1 & e = (v,u) \\ 0 & \text{otherwise} \end{cases} \qquad L_{u,v} = \begin{cases} d & u = v \\ -1 & (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$



B

	a	b	c	d
(a,b)	1	-1	0	0
(a,c)	1	0	-1	0
(c,d)	0	0	1	-1
(d,b)	0	-1	0	1
(b,c)	0	1	-1	1

L

	a	b	c	d
a	2	-1	-1	0
b	-1	2	0	-1
c	-1	-1	3	-1
d	-1	-1	-1	3

Fun facts: $\mathbf{f} \in \Re^{|E|}$

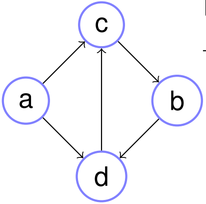
$[B^T f]_u = \sum_{e=(u,v)} f_e - \sum_{e=(v,u)} f_e$

$B^T B = L$

Some Matrices.

Given $G = (V,E)$, arbitrarily orient edges.

$$B_{v,e} = \begin{cases} -1 & e=(u,v) \\ 1 & e=(v,u) \\ 0 & \text{otherwise} \end{cases} \qquad L_{u,v} = \begin{cases} d & u=v \\ -1 & (u,v)\in E \\ 0 & \text{otherwise} \end{cases}$$



B

	a	b	c	d
(a,b)	1	-1	0	0
(a,c)	1	0	-1	0
(c,d)	0	0	1	-1
(d,b)	0	-1	0	1
(b,c)	0	1	-1	1

L

	a	b	c	d
a	2	-1	-1	0
b	-1	2	0	-1
c	-1	-1	3	-1
d	-1	-1	-1	3

Fun facts: $\mathbf{f} \in \Re^{|E|}$

$[B^T f]_u = \sum_{e=(u,v)} f_e - \sum_{e=(v,u)} f_e$

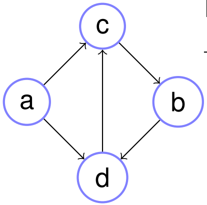
$B^T B = L$

$[Bx]_{e=(u,v)} = x_u - x_v$

Some Matrices.

Given $G = (V, E)$, arbitrarily orient edges.

$$B_{v,e} = \begin{cases} -1 & e = (u,v) \\ 1 & e = (v,u) \\ 0 & \text{otherwise} \end{cases} \qquad L_{u,v} = \begin{cases} d & u = v \\ -1 & (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$



B

	a	b	c	d
(a,b)	1	-1	0	0
(a,c)	1	0	-1	0
(c,d)	0	0	1	-1
(d,b)	0	-1	0	1
(b,c)	0	1	-1	1

L

	a	b	c	d
a	2	-1	-1	0
b	-1	2	0	-1
c	-1	-1	3	-1
d	-1	-1	-1	3

Fun facts: $\mathbf{f} \in \Re^{|E|}$

$[B^T f]_u = \sum_{e=(u,v)} f_e - \sum_{e=(v,u)} f_e$

$B^T B = L$

$[Bx]_{e=(u,v)} = x_u - x_v$

$x^T L x = \sum_{e=(u,v)} (x_u - x_v)^2$

Duality..

Given G, χ, $\chi \perp 1$

Duality..

Given G, χ, $\chi \perp 1$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian:

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual:

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$?

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Dual problem:

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Dual problem: Find ϕ that maximizes ...

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Dual problem: Find ϕ that maximizes ...
$\max L(\phi, B\phi) = \phi^T B^T B\phi + 2\phi^T(\chi - B^T B\phi)$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Dual problem: Find ϕ that maximizes ...
$\max L(\phi, B\phi) = \phi^T B^T B\phi + 2\phi^T(\chi - B^T B\phi) \implies \max_\phi 2\phi^T \chi - \phi^T L\phi$

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Dual problem: Find ϕ that maximizes ...
$\max L(\phi, B\phi) = \phi^T B^T B\phi + 2\phi^T(\chi - B^T B\phi) \implies \max_\phi 2\phi^T \chi - \phi^T L\phi$
Note: want $\phi^T L\phi = \sum_e (\phi_u - \phi_v)^2$ to be small.

Duality..

Given G, χ, $\chi \perp 1$

Minimize $|f|^2$ subject to $B^T f = \chi$.

Lagrangian: $L(\phi, f) = \sum_e f(e)^2 + 2\phi^T(\chi - B^T f)$

Lagrangian Dual: Find ϕ that maximizes $\min_f L(\phi, f)$.

Given ϕ, minimize $L(\phi, f)$? Calculus.

For $e = (u, v)$
$2f(e) + 2(\phi_v - \phi_u) = 0$ (Minimum when partial derivatives = 0.)

$\rightarrow f(e) = (\phi_u - \phi_v)$ Potential differences!!!

Matrix Form: $f = B\phi$ Again, flows should be potential differences.

Dual problem: Find ϕ that maximizes ...
$\max L(\phi, B\phi) = \phi^T B^T B\phi + 2\phi^T(\chi - B^T B\phi) \implies \max_\phi 2\phi^T \chi - \phi^T L\phi$
Note: want $\phi^T L\phi = \sum_e (\phi_u - \phi_v)^2$ to be small.
Minimize Squared Potential differences while routing current!

Why did we take dual?

Dual problem:

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_{\phi} 2\phi\chi - \phi L\phi$

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_\phi 2\phi\chi - \phi L\phi$

Take the derivative:

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_\phi 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_{\phi} 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_{\phi} 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Optimal potential is solution to a Laplacian linear system.

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_\phi 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Optimal potential is solution to a Laplacian linear system.

Also useful for convergence.

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_{\phi} 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Optimal potential is solution to a Laplacian linear system.

Also useful for convergence.

Algorithm maintains feasible ϕ, f,

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_{\phi} 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Optimal potential is solution to a Laplacian linear system.

Also useful for convergence.

Algorithm maintains feasible ϕ, f,

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi^T L\phi$

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_\phi 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Optimal potential is solution to a Laplacian linear system.

Also useful for convergence.

Algorithm maintains feasible ϕ, f,

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi^T L\phi$

Duality gap is “distance” from optimal!

Why did we take dual?

Dual problem:
Find ϕ that maximizes ...
$\max_\phi 2\phi\chi - \phi L\phi$

Take the derivative:
$L\phi - \chi$
$L\phi = \chi$ at optimal point!

Optimal potential is solution to a Laplacian linear system.

Also useful for convergence.

Algorithm maintains feasible ϕ, f,

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi^T L\phi$

Duality gap is “distance” from optimal!

Algorithm: Work on flow and potentials.
To drive gap to 0.

Alg.

Given: χ, G

Alg.

Given: χ, G

Take a spanning tree T of G.

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log\log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1 + \varepsilon)$ approximation in $O(m \log n \log \log n \log(\frac{n}{\varepsilon}))$

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1 + \varepsilon)$ approximation in $O(m \log n \log \log n \log(\frac{n}{\varepsilon}))$!

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log\log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1 + \varepsilon)$ approximation in $O(m \log n \log\log n \log(\frac{n}{\varepsilon}))$! !

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1 + \varepsilon)$ approximation in $O(m \log n \log \log n \log(\frac{n}{\varepsilon}))$! ! !

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log\log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1+\varepsilon)$ approximation in $O(m \log n \log\log n \log(\frac{n}{\varepsilon}))$! ! ! !

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1 + \varepsilon)$ approximation in $O(m \log n \log \log n \log(\frac{n}{\varepsilon}))$! ! ! ! !

Alg.

Given: χ, G

Take a spanning tree T of G. (Which tree?)

Route flow, f, to satisfy χ through T
Compute, ϕ, using tree ; $\phi_s = 0$, add f_e through T

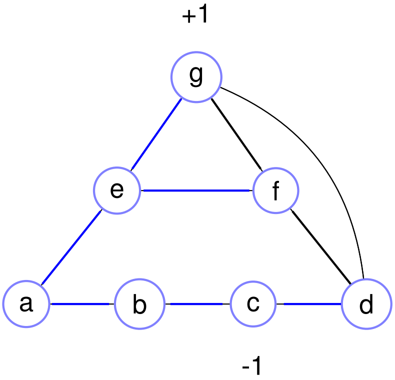
Repeat:
Choose non-tree edge $e = (u, v)$ (Which non-tree edge?)
$f(e) = (\phi_u - \phi_v)/(\ell_T(u, v) + 1)$
($\ell_T(u, v)$ path length in T)
Route excess on path through tree.

Which Tree?
Claim: Linear time algorithm for T w/ stretch $O(m \log n \log \log n)$!
Stretch: $\sum_{e=(u,v)} \ell_T(u, v)$

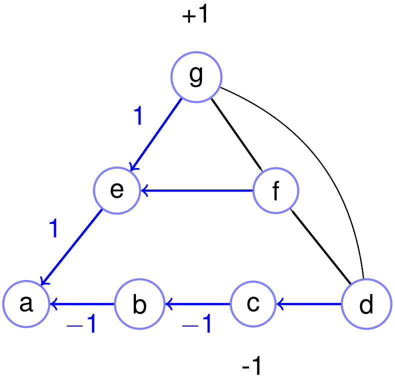
Which non-tree edge?
Choose an edge w/prob. proportional to $\ell_T(e)$.

Finds $(1 + \varepsilon)$ approximation in $O(m \log n \log \log n \log(\frac{n}{\varepsilon}))$! ! ! ! ! !

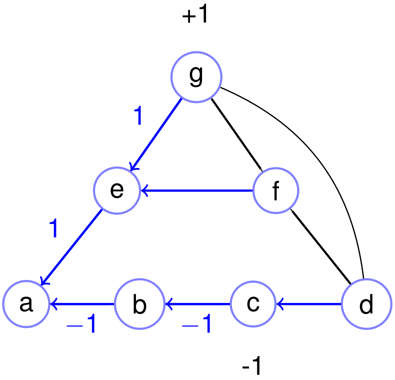
Animation



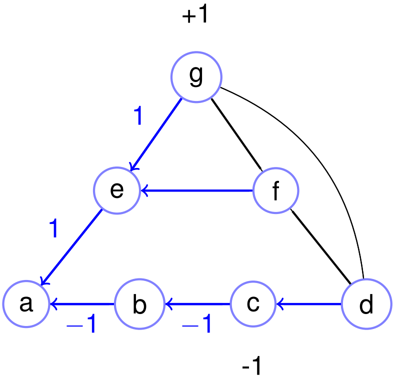
Animation



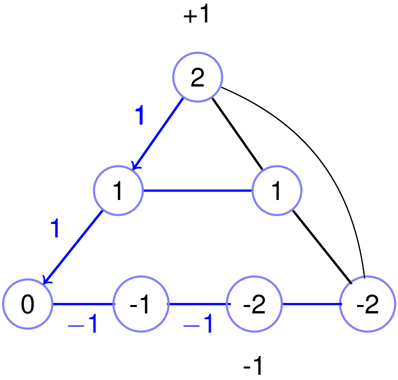
Animation



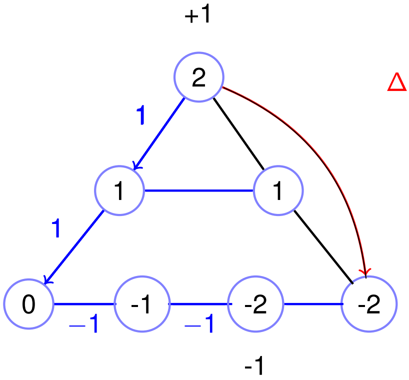
Animation



Animation

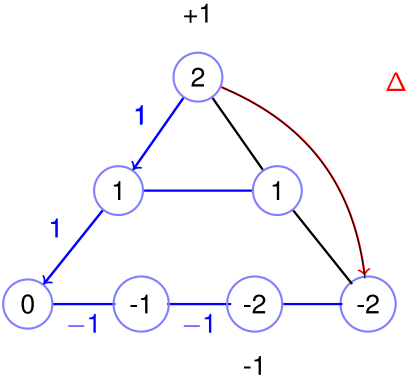


Animation



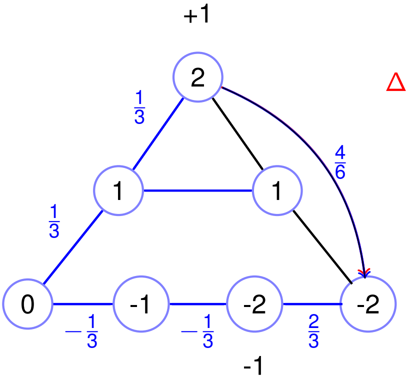
$\Delta_c = 4, R_e = 6$

Animation



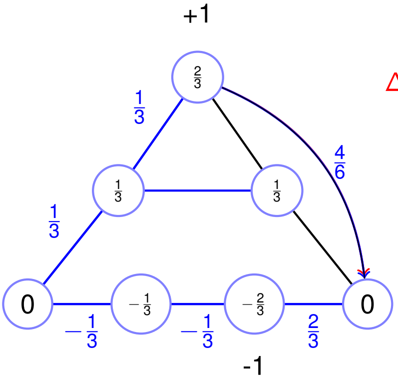
$\Delta_c = 4, R_e = 6$

Animation



$\Delta_c = 4, R_e = 6$

Animation



$\Delta_c = 4, R_e = 6$

Energy reduction.

Given T,$e = (u, v)$, let $R_e = \ell_T(u, v) + 1$.

Energy reduction.

Given T, $e = (u, v)$, let $R_e = \ell_T(u, v) + 1$.

Algorithm:

Energy reduction.

Given T, $e = (u, v)$, let $R_e = \ell_T(u, v) + 1$.

Algorithm:

Energy reduction.

Given T,$e = (u, v)$, let $R_e = \ell_T(u, v) + 1$.

Algorithm:

Repeatedly "Fix" edge $e = (u, v)$.

Energy reduction.

Given T,$e = (u,v)$, let $R_e = \ell_T(u,v) + 1$.

Algorithm:

Repeatedly “Fix” edge $e = (u,v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e
(assume e' are oriented around cycle.)

Energy reduction.

Given T,$e = (u,v)$, let $R_e = \ell_T(u,v) + 1$.

Algorithm:

Repeatedly “Fix” edge $e = (u,v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e

(assume e' are oriented around cycle.)

Difference in energy from f and f'.

Energy reduction.

Given T,$e = (u,v)$, let $R_e = \ell_T(u,v)+1$.

Algorithm:

Repeatedly "Fix" edge $e = (u,v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e
(assume e' are oriented around cycle.)

Difference in energy from f and f'.
$\sum_{e' \in C_e}(f(e') - \delta)^2 - (f(e'))^2$

Energy reduction.

Given T,$e=(u,v)$, let $R_e=\ell_T(u,v)+1$.

Algorithm:

Repeatedly “Fix” edge $e=(u,v)$.

Route $-\delta=-\frac{\sum_{e'\in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e
(assume e' are oriented around cycle.)

Difference in energy from f and f'.
$\sum_{e'\in C_e}(f(e')-\delta)^2-(f(e'))^2=\sum_{e'\in C_e}-2f(e')\delta+\delta^2$

Energy reduction.

Given T,$e = (u, v)$, let $R_e = \ell_T(u, v) + 1$.

Algorithm:

Repeatedly "Fix" edge $e = (u, v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e

(assume e' are oriented around cycle.)

Difference in energy from f and f'.

$\sum_{e' \in C_e} (f(e') - \delta)^2 - (f(e'))^2 = \sum_{e' \in C_e} -2f(e')\delta + \delta^2$

$= -(2\delta \sum_{e' \in C_e} f(e')) + R_E \delta^2$

Energy reduction.

Given T,$e = (u,v)$, let $R_e = \ell_T(u,v) + 1$.

Algorithm:

Repeatedly "Fix" edge $e = (u,v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e

(assume e' are oriented around cycle.)

Difference in energy from f and f'.

$\sum_{e' \in C_e} (f(e') - \delta)^2 - (f(e'))^2 = \sum_{e' \in C_e} -2f(e')\delta + \delta^2$

$= -(2\delta \sum_{e' \in C_e} f(e')) + R_E \delta^2$

Note: $\sum_{e' \in C_e} f(e') = R_e \delta$

Energy reduction.

Given T,$e = (u, v)$, let $R_e = \ell_T(u, v) + 1$.

Algorithm:

Repeatedly “Fix” edge $e = (u, v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e

(assume e' are oriented around cycle.)

Difference in energy from f and f'.

$\sum_{e' \in C_e}(f(e') - \delta)^2 - (f(e'))^2 = \sum_{e' \in C_e} -2f(e')\delta + \delta^2$

$= -(2\delta \sum_{e' \in C_e} f(e')) + R_E \delta^2$

Note: $\sum_{e' \in C_e} f(e') = R_e \delta$

$\rightarrow -\Delta^2_{C_e}/R_e$ where $\Delta_{C_e} = \sum_{e' \in C_e} f(e')$.

Energy reduction.

Given T,$e=(u,v)$, let $R_e=\ell_T(u,v)+1$.

Algorithm:

Repeatedly “Fix” edge $e=(u,v)$.

Route $-\delta=-\frac{\sum_{e'\in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e
(assume e' are oriented around cycle.)

Difference in energy from f and f'.
$\sum_{e'\in C_e}(f(e')-\delta)^2-(f(e'))^2=\sum_{e'\in C_e}-2f(e')\delta+\delta^2$
$=-(2\delta\sum_{e'\in C_e} f(e'))+R_E\delta^2$

Note: $\sum_{e'\in C_e} f(e')=R_e\delta$

$\rightarrow -\Delta_{C_e}^2/R_e$ where $\Delta_{C_e}=\sum_{e'\in C_e} f(e')$.

Fix a part of the potential difference, Δ_{c_e} around cyle!!

Energy reduction.

Given T,$e = (u,v)$, let $R_e = \ell_T(u,v) + 1$.

Algorithm:

Repeatedly "Fix" edge $e = (u,v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e

(assume e' are oriented around cycle.)

Difference in energy from f and f'.

$\sum_{e' \in C_e} (f(e') - \delta)^2 - (f(e'))^2 = \sum_{e' \in C_e} -2f(e')\delta + \delta^2$

$= -(2\delta \sum_{e' \in C_e} f(e')) + R_E \delta^2$

Note: $\sum_{e' \in C_e} f(e') = R_e \delta$

$\rightarrow -\Delta_{C_e}^2 / R_e$ where $\Delta_{C_e} = \sum_{e' \in C_e} f(e')$.

Fix a part of the potential difference, Δ_{c_e} around cyle!!

$\rightarrow$ reduction of $\Delta_{C_e}^2 / R_e$ in energy!

Energy reduction.

Given T,$e = (u,v)$, let $R_e = \ell_T(u,v) + 1$.

Algorithm:

Repeatedly "Fix" edge $e = (u,v)$.

Route $-\delta = -\frac{\sum_{e' \in C_e} f(e')}{R_e}$ flow around cycle induced in T: C_e

(assume e' are oriented around cycle.)

Difference in energy from f and f'.

$\sum_{e' \in C_e} (f(e') - \delta)^2 - (f(e'))^2 = \sum_{e' \in C_e} -2f(e')\delta + \delta^2$

$= -(2\delta \sum_{e' \in C_e} f(e')) + R_E \delta^2$

Note: $\sum_{e' \in C_e} f(e') = R_e \delta$

$\rightarrow -\Delta_{C_e}^2 / R_e$ where $\Delta_{C_e} = \sum_{e' \in C_e} f(e')$.

Fix a part of the potential difference, Δ_{c_e} around cyle!!

$\rightarrow$ reduction of $\Delta_{C_e}^2 / R_e$ in energy!

Fix $1/R_e$ of a cycle violation!

Duality Gap?

Algorithm maintains feasible ϕ, f, $(B^T f = \chi)$

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L \phi$

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L \phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Duality Gap?

Algorithm maintains feasible ϕ, f, $(B^T f = \chi)$

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T(f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$

Duality Gap?

Algorithm maintains feasible ϕ, f, $(B^T f = \chi)$

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$ Difference between ϕ flow and f.

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$ Difference between ϕ flow and f.

$\Delta_\phi(u,v) = \sum_{e \in P_{u,v}} -f(e)$. assume $f(e)$ is oriented around cycle.

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$ Difference between ϕ flow and f.

$\Delta_\phi(u,v) = \sum_{e \in P_{u,v}} -f(e)$. assume $f(e)$ is oriented around cycle.

For $e \in T$, $\Delta_\phi(e) = f(e)$.

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$ Difference between ϕ flow and f.

$\Delta_\phi(u,v) = \sum_{e \in P_{u,v}} -f(e)$. assume $f(e)$ is oriented around cycle.

For $e \in T$, $\Delta_\phi(e) = f(e)$. For $e \notin T$ $\Delta_{C_e} = f(e) + \sum_{e \in P_e} f(e)$

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.
$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$ Difference between ϕ flow and f.

$\Delta_\phi(u,v) = \sum_{e \in P_{u,v}} -f(e)$. assume $f(e)$ is oriented around cycle.

For $e \in T$, $\Delta_\phi(e) = f(e)$. For $e \notin T$ $\Delta_{C_e} = f(e) + \sum_{e \in P_e} f(e)$

Duality Gap: $\sum_{e \notin T} (\Delta_{C_e}(f))^2$

Duality Gap?

Algorithm maintains feasible ϕ, f, ($B^T f = \chi$)

Primal value: $|f|^2$.
Dual value: $2\phi\chi - \phi L\phi$

ϕ is tree induced voltages.

Total Duality Gap?

Gap: $|f|^2 - (2\phi^T\chi - \phi^T L\phi)$.

$= |f|^2 - 2\phi^T B^T f + \phi^T B^T B\phi$ where $B^T f = \chi$ and $L = B^T B$.
$= (f - B\phi)^T (f - B\phi)$.

Gap $= \sum_e (f(e) - \Delta_\phi(e))^2$ Difference between ϕ flow and f.

$\Delta_\phi(u,v) = \sum_{e \in P_{u,v}} -f(e)$. assume $f(e)$ is oriented around cycle.

For $e \in T$, $\Delta_\phi(e) = f(e)$. For $e \notin T$ $\Delta_{C_e} = f(e) + \sum_{e \in P_e} f(e)$

Duality Gap: $\sum_{e \notin T} (\Delta_{C_e}(f))^2$

Total distance from optimal is cycle violations!

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Polishing off.

Claim: E[change in energy$|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Polishing off.

Claim: E[change in energy$|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta^2_{C_e}}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e}$

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{c_e}^2}{R_e} = -\sum_e \frac{\Delta_{c_e}^2}{\tau}$

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{c_e}^2}{R_e} = -\sum_e \frac{\Delta_{c_e}^2}{\tau}$ □

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{c_e}^2}{R_e} = -\sum_e \frac{\Delta_{c_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{c_e}^2}{R_e} = -\sum_e \frac{\Delta_{c_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau}\frac{\Delta_{c_e}^2}{R_e} = -\sum_e \frac{\Delta_{c_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e} = -\sum_e \frac{\Delta_{C_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta^2_{C_e}}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta^2_{c_e}}{R_e} = -\sum_e \frac{\Delta^2_{c_e}}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta^2_{C_e}}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau}\frac{\Delta^2_{C_e}}{R_e} = -\sum_e \frac{\Delta^2_{C_e}}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e} = -\sum_e \frac{\Delta_{C_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time!

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e} = -\sum_e \frac{\Delta_{C_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time! !

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e} = -\sum_e \frac{\Delta_{C_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time! ! !

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e} = -\sum_e \frac{\Delta_{C_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time! ! ! !

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta^2_{C_e}}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta^2_{C_e}}{R_e} = -\sum_e \frac{\Delta^2_{C_e}}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time! ! ! ! !

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta^2_{C_e}}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta^2_{C_e}}{R_e} = -\sum_e \frac{\Delta^2_{C_e}}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time! ! ! ! ! !
Enough with the exclamations already

Polishing off.

Claim: $E[\text{change in energy}|Gap] = \frac{Gap}{\tau}$
($\tau = \sum_e \ell_T(e)$ is stretch of E in T.)

Duality Gap: $\sum_{e \notin T} \Delta_{C_e}(f)^2$

Choose edge e reduce energy by $-\frac{\Delta_{C_e}^2}{R_e}$.

Choose edge with probability $\frac{R_e}{\tau}$.

Expected reduction $-\sum_e \frac{R_e}{\tau} \frac{\Delta_{C_e}^2}{R_e} = -\sum_e \frac{\Delta_{C_e}^2}{\tau}$ □

Duality Gap reduces by $(1 - 1/\tau)$ every iteration on expectation.

$O(\tau \log(n/\varepsilon))$ iterations gives $(1+\varepsilon)$ approximation.

$\tau = O(m \log n \log\log n)$ ...

$\tilde{O}(m)$ iterations

Iteration in $O(\log^2 n)$ time using balanced binary trees.

$\rightarrow \tilde{O}(m)$ time! ! ! ! ! !
Enough with the exclamations already !

Wrapup...

How to get low stretch tree?

Wrapup...

How to get low stretch tree?
Answer: Elkin-Spielman-Teng, ...,

Wrapup...

How to get low stretch tree?
Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Wrapup...

How to get low stretch tree?

Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Open: Get $O(m \log n)$ stretch?

Wrapup...

How to get low stretch tree?

Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Open: Get $O(m \log n)$ stretch? Like for HSTs. (Later)

Wrapup...

How to get low stretch tree?

Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Open: Get $O(m \log n)$ stretch? Like for HSTs. (Later)

Ideas: Similar to HST construction, but much more subtle.

Wrapup...

How to get low stretch tree?

Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Open: Get $O(m \log n)$ stretch? Like for HSTs. (Later)

Ideas: Similar to HST construction, but much more subtle.

How to do update along cycle?

Wrapup...

How to get low stretch tree?
Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.
Open: Get $O(m \log n)$ stretch? Like for HSTs. (Later)
Ideas: Similar to HST construction, but much more subtle.

How to do update along cycle?
Answer: Data Structures.

Wrapup...

How to get low stretch tree?

Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Open: Get $O(m \log n)$ stretch? Like for HSTs. (Later)

Ideas: Similar to HST construction, but much more subtle.

How to do update along cycle?

Answer: Data Structures.

Idea: Use a binary tree on paths.

Wrapup...

How to get low stretch tree?

Answer: Elkin-Spielman-Teng, ...,Abraham, Newman.

Open: Get $O(m \log n)$ stretch? Like for HSTs. (Later)

Ideas: Similar to HST construction, but much more subtle.

How to do update along cycle?

Answer: Data Structures.

Idea: Use a binary tree on paths.

Decompose tree into paths.

Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Update(i, j, δ)

$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Update(i, j, δ)

$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)

Return $\sum_{k=i}^{j-1} f(k, k+1)$.

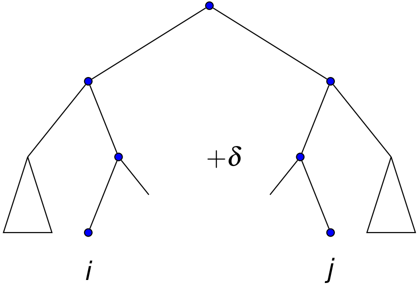
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0, \forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.

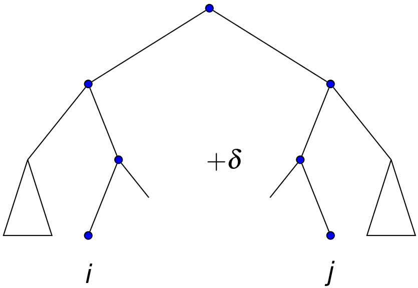
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:

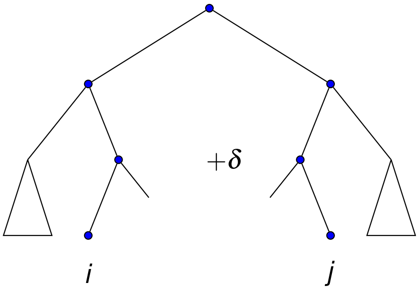
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:
Right children, add δ to edge.

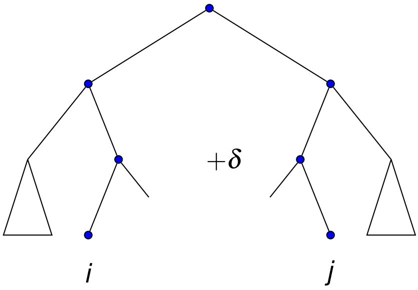
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0, \forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:
Right children, add δ to edge.
Reflect for right path.

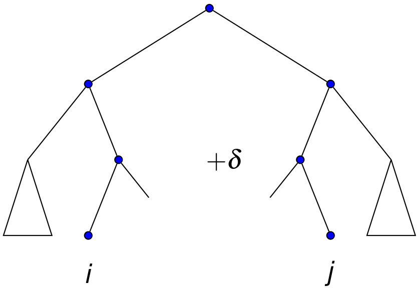
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:
Right children, add δ to edge.
Reflect for right path.

Lookup: Exercise.

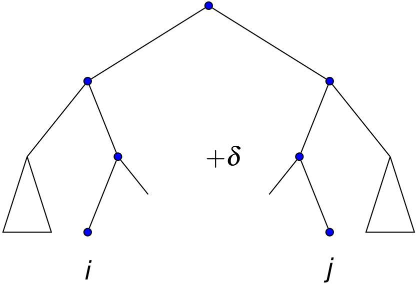
Interval Tree

Path: $G = (V, E)$, $V = \{1, \dots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0, \forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \dots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:
Right children, add δ to edge.
Reflect for right path.

Lookup: Exercise.
Also, need number of descendants.

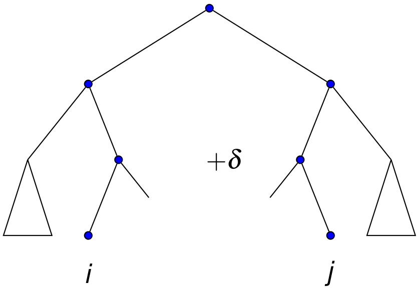
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0, \forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:
Right children, add δ to edge.
Reflect for right path.

Lookup: Exercise.
Also, need number of descendants.

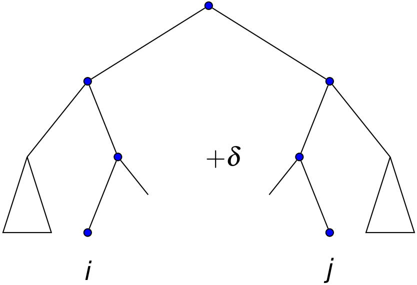
Interval Tree

Path: $G = (V, E)$, $V = \{1, \ldots, n\}$, $E = \{(i, i+1) : i \in \{1, n-1\}\}$

Init: $f(e) = 0$, $\forall e \in G$.

Update(i, j, δ)
$\forall k \in \{i, \ldots, j-1\}$, $f(i, i+1) = f(i, i+1) + \delta$

Lookup(i, j)
Return $\sum_{k=i}^{j-1} f(k, k+1)$.



Update: find common ancestor.
For left path:
Right children, add δ to edge.
Reflect for right path.

Lookup: Exercise.
Also, need number of descendants.

$O(\log n)$ update/lookup.

Path Decomposition of Tree

Given tree T.

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove.

Path Decomposition of Tree

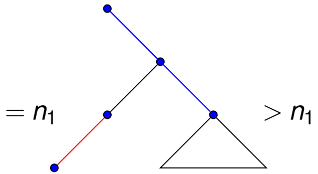
Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse

Path Decomposition of Tree

Given tree T.

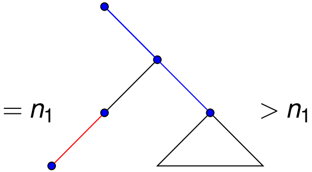
Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse



Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse

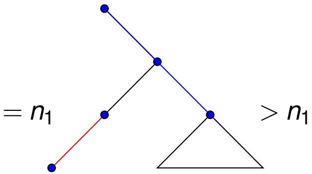


Decomposition Property:

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse

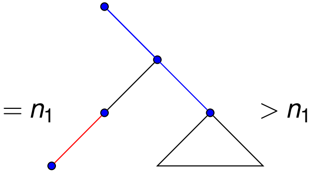


Decomposition Property:
Where a root to leaf path in tree only sees $O(\log n)$ paths.

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse



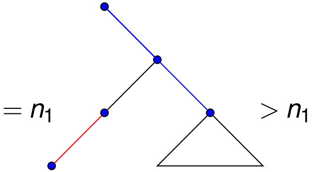
Decomposition Property:
Where a root to leaf path in tree only sees $O(\log n)$ paths.

Proof Idea:

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse



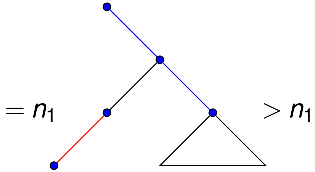
Decomposition Property:
Where a root to leaf path in tree only sees $O(\log n)$ paths.

Proof Idea:
Every path change, doubles number of vertices.

Path Decomposition of Tree

Given tree T.

Decomposition Procedure:
Longest path from root to leaf.
From root, go towards heavier side.
Remove. Recurse



Decomposition Property:
Where a root to leaf path in tree only sees $O(\log n)$ paths.

Proof Idea:
Every path change, doubles number of vertices.

$O(\log^2 n)$ update time for updating flow values on tree edges!

What's going on?

Geometric View.

What's going on?

Geometric View.

Cycles are constraints.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection into subspace defined by constraint.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.

What's going on?

Geometric View.
- Cycles are constraints.
- Flow around cycle $= 0$.
  - Each cycle update is approximate projection into subspace defined by constraint.
  - Solution is intersection of cycle constraints and flow conservation.
    - Kirchhoff's Laws.

What's going on?

Geometric View.

- Cycles are constraints.
- Flow around cycle $= 0$.
  - Each cycle update is approximate projection into subspace defined by constraint.
  - Solution is intersection of cycle constraints and flow conservation.
    Kirchhoff's Laws.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.

Kirchhoff's Laws.

Algorithmic Power.

What's going on?

Geometric View.

- Cycles are constraints.
- Flow around cycle $= 0$.
  - Each cycle update is approximate projection
    into subspace defined by constraint.
  - Solution is intersection of cycle constraints and flow conservation.
    Kirchhoff's Laws.

Algorithmic Power.

- Algebra: Solving exactly on tree.

What's going on?

Geometric View.

- Cycles are constraints.
- Flow around cycle $= 0$.
  - Each cycle update is approximate projection
    into subspace defined by constraint.
  - Solution is intersection of cycle constraints and flow conservation.
    Kirchhoff's Laws.

Algorithmic Power.

- Algebra: Solving exactly on tree.

What's going on?

Geometric View.

- Cycles are constraints.
- Flow around cycle $= 0$.
  - Each cycle update is approximate projection
    into subspace defined by constraint.
  - Solution is intersection of cycle constraints and flow conservation.
    Kirchhoff's Laws.

Algorithmic Power.

- Algebra: Solving exactly on tree.
- Calculus: make a local move th decreases potential.

What's going on?

Geometric View.

- Cycles are constraints.
- Flow around cycle $= 0$.
  - Each cycle update is approximate projection
    into subspace defined by constraint.
  - Solution is intersection of cycle constraints and flow conservation.
    Kirchhoff's Laws.

Algorithmic Power.

- Algebra: Solving exactly on tree.
- Calculus: make a local move th decreases potential.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.

Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.
Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.

Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.

Preconditioner uses tree to make sparse version of graph.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.
Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.
Preconditioner uses tree to make sparse version of graph.

Correspondence to Practice:

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.
Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.

Preconditioner uses tree to make sparse version of graph.

Correspondence to Practice:

Random sparsification of Cholesky factorization.

(Kyung-Sachdeva)

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.
Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.
Preconditioner uses tree to make sparse version of graph.

Correspondence to Practice:

Random sparsification of Cholesky factorization.

(Kyung-Sachdeva)

Tree pre-conditioner.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.
Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.

Preconditioner uses tree to make sparse version of graph.

Correspondence to Practice:

Random sparsification of Cholesky factorization.

(Kyung-Sachdeva)

Tree pre-conditioner.

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.

Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.

Preconditioner uses tree to make sparse version of graph.

Correspondence to Practice:

Random sparsification of Cholesky factorization.

(Kyung-Sachdeva)

Tree pre-conditioner.

Laplacian Systems are quite general:

What's going on?

Geometric View.

Cycles are constraints.

Flow around cycle $= 0$.

Each cycle update is approximate projection
into subspace defined by constraint.

Solution is intersection of cycle constraints and flow conservation.
Kirchhoff's Laws.

Algorithmic Power.

Algebra: Solving exactly on tree.

Calculus: make a local move th decreases potential.

Better Algorithm:

Recursive algorithm give $O(m\sqrt{\log n})$ iterations to halve error.
Preconditioner uses tree to make sparse version of graph.

Correspondence to Practice:

Random sparsification of Cholesky factorization.

(Kyung-Sachdeva)

Tree pre-conditioner.

Laplacian Systems are quite general:

Climate, physics, SDD-matrices.

Next

Rayleigh quotient.

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d}\lambda_{min}(L)$.

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d}\lambda_{min}(L)$.

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d} \lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d}\lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

$\frac{\mu}{2}$

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d} \lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2}$

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d} \lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G)$

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d}\lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)}$

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d}\lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Next

Rayleigh quotient.

$\lambda_2 = \max_{x \perp \mathbf{1}} \frac{x^T M x}{x^T x}$.

Eigenvalue gap: $\mu = \lambda_1 - \lambda_2 = \frac{1}{d}\lambda_{min}(L)$.

Recall: $h(G) = \min_{S, |S| \leq |V|/2} \frac{|E(S, V-S)|}{|S|}$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Today: the right hand inequality.

Hypercube

$V = \{0, 1\}^d$

Hypercube

$V = \{0, 1\}^d \quad (x, y) \in E$

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

Hypercube

$V = \{0, 1\}^d \quad (x, y) \in E$ when x and y differ in one bit.

$|V| = 2^d$

Hypercube

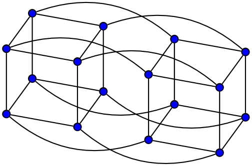
$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.

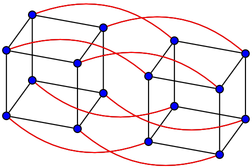


Good cuts?

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d \ |E| = d2^{d-1}$.

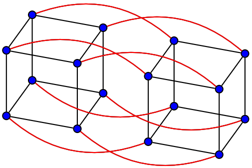


Good cuts? "Coordinate cut": d of them.

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d \ |E| = d2^{d-1}$.



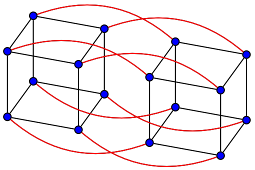
Good cuts? "Coordinate cut": d of them.

Edge expansion:

Hypercube

$V = \{0,1\}^d$ $\quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.



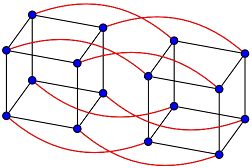
Good cuts? "Coordinate cut": d of them.

Edge expansion: $\frac{2^{d-1}}{d2^{d-1}}$

Hypercube

$V = \{0,1\}^d$ $\quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.



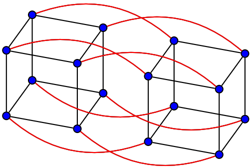
Good cuts? "Coordinate cut": d of them.

Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Hypercube

$V = \{0,1\}^d$ $\quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.



Good cuts? "Coordinate cut": d of them.

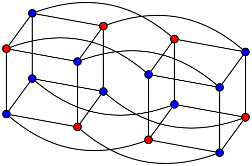
Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d \ |E| = d2^{d-1}$.



Good cuts? "Coordinate cut": d of them.

Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

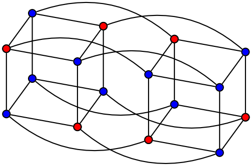
Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.

Vertex cut size:

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d \; |E| = d2^{d-1}$.



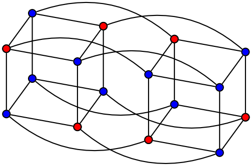
Good cuts? "Coordinate cut": d of them.
Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.
Vertex cut size: $\binom{d}{d/2}$ bit strings with $d/2$ 1's.

Hypercube

$V = \{0,1\}^d$ $\quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.



Good cuts? "Coordinate cut": d of them.

Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.

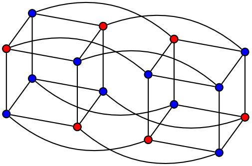
Vertex cut size: $\binom{d}{d/2}$ bit strings with $d/2$ 1's.

$\approx \frac{2^d}{\sqrt{d}}$

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d \; |E| = d2^{d-1}$.



Good cuts? "Coordinate cut": d of them.

Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.

Vertex cut size: $\binom{d}{d/2}$ bit strings with $d/2$ 1's.

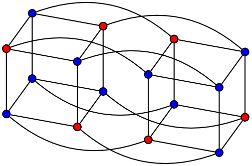
$$\approx \frac{2^d}{\sqrt{d}}$$

Vertex expansion: $\approx \frac{1}{\sqrt{d}}$.

Hypercube

$V = \{0,1\}^d \quad (x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d \; |E| = d2^{d-1}$.



Good cuts? "Coordinate cut": d of them.

Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.

Vertex cut size: $\binom{d}{d/2}$ bit strings with $d/2$ 1's.

$\approx \frac{2^d}{\sqrt{d}}$

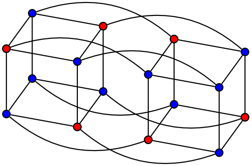
Vertex expansion: $\approx \frac{1}{\sqrt{d}}$.

Edge expansion: $d/2$ edges to next level. $\approx \frac{1}{2\sqrt{d}}$

Hypercube

$V = \{0,1\}^d$ $(x,y) \in E$ when x and y differ in one bit.

$|V| = 2^d$ $|E| = d2^{d-1}$.



Good cuts? "Coordinate cut": d of them.
Edge expansion: $\frac{2^{d-1}}{d2^{d-1}} = \frac{1}{d}$

Ball cut: All nodes within $d/2$ of node, say $00\cdots0$.
Vertex cut size: $\binom{d}{d/2}$ bit strings with $d/2$ 1's.

$$\approx \frac{2^d}{\sqrt{d}}$$

Vertex expansion: $\approx \frac{1}{\sqrt{d}}$.
Edge expansion: $d/2$ edges to next level. $\approx \frac{1}{2\sqrt{d}}$

Worse by a factor of $\sqrt{d}$

Eigenvalues of hypercube.

Anyone see any symmetry?

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1 - 2/d)v_i$.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite. Color ± 1

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite. Color ± 1

Eigenvalue:

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite. Color ± 1

Eigenvalue: $1-4/d$.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1 - 2/d)v_i$.

Eigenvalue $1 - 2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite. Color ± 1

Eigenvalue: $1 - 4/d$. $\binom{d}{2}$ eigenvectors.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite. Color ± 1

Eigenvalue: $1-4/d$. $\binom{d}{2}$ eigenvectors.

Eigenvalues: $1-2k/d$.

Eigenvalues of hypercube.

Anyone see any symmetry?

Coordinate cuts. +1 on one side, -1 on other.

$(Mv)_i = (1-2/d)v_i$.

Eigenvalue $1-2/d$. d Eigenvectors. Why orthogonal?

Next eigenvectors?

Delete edges in two dimensions.

Four subcubes: bipartite. Color ± 1

Eigenvalue: $1-4/d$. $\binom{d}{2}$ eigenvectors.

Eigenvalues: $1-2k/d$. $\binom{d}{k}$ eigenvectors.

Back to Cheeger.

Coordinate Cuts:

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1 - 2/d$.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1 - 2/d$. d Eigenvectors.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$\frac{\mu}{2}$

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1 - 2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2}$

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1 - 2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G)$

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)}$

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \le h(G) \le \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \le h(G) \le \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose "names" in hypercube, find coordinate cut?

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose “names” in hypercube, find coordinate cut?

Find coordinate cut?

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose “names” in hypercube, find coordinate cut?

Find coordinate cut?

Eigenvector v maps to line.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose “names” in hypercube, find coordinate cut?

Find coordinate cut?

Eigenvector v maps to line.
Cut along line.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose "names" in hypercube, find coordinate cut?

Find coordinate cut?

Eigenvector v maps to line.
Cut along line.

Eigenvector algorithm gets a linear combination of coordinate cuts.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose "names" in hypercube, find coordinate cut?

Find coordinate cut?

Eigenvector v maps to line.
Cut along line.

Eigenvector algorithm gets a linear combination of coordinate cuts.

Something like ball cut.

Back to Cheeger.

Coordinate Cuts:
Eigenvalue $1-2/d$. d Eigenvectors.

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

For hypercube: $h(G) = \frac{1}{d}$ $\lambda_1 - \lambda_2 = 2/d$.
Left hand side is tight.

Note: hamming weight vector also in first eigenspace.

Lose “names” in hypercube, find coordinate cut?

Find coordinate cut?

Eigenvector v maps to line.
Cut along line.

Eigenvector algorithm gets a linear combination of coordinate cuts.

Something like ball cut.

Find coordinate cut?

Cycle

Tight example for Other side of Cheeger?

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle on n nodes.

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle on n nodes.

Will show other side of Cheeger is tight.

Cycle

Tight example for Other side of Cheeger?

$$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$$

Cycle on n nodes.

Will show other side of Cheeger is tight.

Edge expansion:Cut in half.

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle on n nodes.

Will show other side of Cheeger is tight.

Edge expansion:Cut in half.
$|S| = n/2$, $|E(S,\overline{S})| = 2$

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle on n nodes.

Will show other side of Cheeger is tight.

Edge expansion:Cut in half.
$|S| = n/2$, $|E(S, \overline{S})| = 2$
$\rightarrow h(G) = \frac{2}{n}$.

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle on n nodes.

Will show other side of Cheeger is tight.

Edge expansion:Cut in half.
$|S| = n/2$, $|E(S,\overline{S})| = 2$
$\rightarrow h(G) = \frac{2}{n}$.

Show eigenvalue gap $\mu \leq \frac{1}{n^2}$.

Cycle

Tight example for Other side of Cheeger?

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Cycle on n nodes.

Will show other side of Cheeger is tight.

Edge expansion:Cut in half.
$|S| = n/2$, $|E(S,\overline{S})| = 2$
$\rightarrow h(G) = \frac{2}{n}$.

Show eigenvalue gap $\mu \leq \frac{1}{n^2}$.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x (1 - O(\frac{1}{n^2}))$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \le n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x (1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \ge 1 - O(\frac{1}{n^2})$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x(1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \geq 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x(1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \geq 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T Mx}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \le n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T Mx = x^T x(1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \ge 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

$\frac{\mu}{2}$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x (1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \geq 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2}$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \le n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x (1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \ge 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \le h(G)$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \le n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x(1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \ge 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \le h(G) \le \sqrt{2(1-\lambda_2)}$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T Mx}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \leq n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T Mx = x^T x(1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \geq 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \leq h(G) \leq \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Slow vector.

Find $x \perp \mathbf{1}$ with Rayleigh quotient, $\frac{x^T M x}{x^T x}$ close to 1.

$$x_i = \begin{cases} i - n/4 & \text{if } i \le n/2 \\ 3n/4 - i & \text{if } i > n/2 \end{cases}$$

Hit with M.

$$(Mx)_i = \begin{cases} -n/4 + 1/2 & \text{if } i = 1, n \\ n/4 - 1 & \text{if } i = n/2 \\ x_i & \text{otherwise} \end{cases}$$

$\rightarrow x^T M x = x^T x(1 - O(\frac{1}{n^2})) \quad \rightarrow \lambda_2 \ge 1 - O(\frac{1}{n^2})$

$\mu = \lambda_1 - \lambda_2 = O(\frac{1}{n^2})$

$h(G) = \frac{2}{n} = \Theta(\sqrt{\mu})$

$\frac{\mu}{2} = \frac{1-\lambda_2}{2} \le h(G) \le \sqrt{2(1-\lambda_2)} = \sqrt{2\mu}$

Tight example for upper bound for Cheeger.

Eigenvalues of cycle?

Eigenvalues: $\cos \frac{2\pi k}{n}$.

Eigenvalues of cycle?

Eigenvalues: $\cos \frac{2\pi k}{n}$.

$x_i = \cos \frac{2\pi k i}{n}$

Eigenvalues of cycle?

Eigenvalues: $\cos \frac{2\pi k}{n}$.

$x_i = \cos \frac{2\pi k i}{n}$

$(Mx)_i$

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi ki}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right)$

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi k i}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi k i}{n}\right)$

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi k i}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi k i}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi ki}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi ki}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues:

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi k i}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi k i}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues:
vibration modes of system.

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi ki}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi ki}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues:
vibration modes of system.
Fourier basis.

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi k i}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi k i}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues:

vibration modes of system.

Fourier basis.

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi ki}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi ki}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues:
- vibration modes of system.
- Fourier basis.

$\cos x \approx 1 + \sin(0)x - \cos(0)\frac{x^2}{2}$.

Eigenvalues of cycle?

Eigenvalues: $\cos\frac{2\pi k}{n}$.

$x_i = \cos\frac{2\pi ki}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi ki}{n}\right)$

Eigenvalue: $\cos\frac{2\pi k}{n}$.

Eigenvalues:
vibration modes of system.
Fourier basis.

$\cos x \approx 1 + \sin(0)x - \cos(0)\frac{x^2}{2}$.

$\cos\frac{2\pi}{n} \approx 1 + \sin(0)x - \cos(0)\frac{x^2}{2}$.

Eigenvalues of cycle?

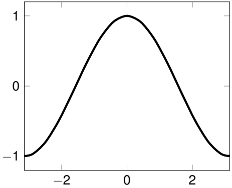
Eigenvalues: $\cos \frac{2\pi k}{n}$.

$x_i = \cos \frac{2\pi ki}{n}$

$(Mx)_i = \cos\left(\frac{2\pi k(i+1)}{n}\right) + \cos\left(\frac{2\pi k(i-1)}{n}\right) = 2\cos\left(\frac{2\pi k}{n}\right)\cos\left(\frac{2\pi ki}{n}\right)$

Eigenvalue: $\cos \frac{2\pi k}{n}$.

Eigenvalues:
- vibration modes of system.
- Fourier basis.



$\cos x \approx 1 + \sin(0)x - \cos(0)\frac{x^2}{2}$.

$\cos \frac{2\pi}{n} \approx 1 + \sin(0)x - \cos(0)\frac{x^2}{2}$.

Second biggest eigenvalue: $\approx 1 - \frac{2\pi^2}{n^2}$.

$h(G) = \frac{2}{n} \leq \frac{\pi}{n} \approx \sqrt{2\mu}$.

Random Walk.

p - probability distribution.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
  Mp.

Converge to uniform distribution.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

$M^t x = a_1 \lambda_1^t v_1 + a_2 \lambda_2 v_2 + \cdots$

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

$M^t x = a_1 \lambda_1^t v_1 + a_2 \lambda_2 v_2 + \cdots$

$\lambda_1 - \lambda_2$ - rate of convergence.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

$M^t x = a_1 \lambda_1^t v_1 + a_2 \lambda_2 v_2 + \cdots$

$\lambda_1 - \lambda_2$ - rate of convergence.

$\Omega(n^2)$ steps to get close to uniform.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

$M^t x = a_1 \lambda_1^t v_1 + a_2 \lambda_2 v_2 + \cdots$

$\lambda_1 - \lambda_2$ - rate of convergence.

$\Omega(n^2)$ steps to get close to uniform.

Start at node 0, probability distribution, $[1, 0, 0, \cdots, 0]$.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

$M^t x = a_1 \lambda_1^t v_1 + a_2 \lambda_2 v_2 + \cdots$

$\lambda_1 - \lambda_2$ - rate of convergence.

$\Omega(n^2)$ steps to get close to uniform.

Start at node 0, probability distribution, $[1, 0, 0, \cdots, 0]$.
Takes $\Omega(n^2)$ to get n steps away.

Random Walk.

p - probability distribution.

Probability distrubtion after choose a random neighbor.
Mp.

Converge to uniform distribution.

Power method: $M^t x$ goes to highest eigenvector.

$M^t x = a_1 \lambda_1^t v_1 + a_2 \lambda_2 v_2 + \cdots$

$\lambda_1 - \lambda_2$ - rate of convergence.

$\Omega(n^2)$ steps to get close to uniform.

Start at node 0, probability distribution, $[1, 0, 0, \cdots, 0]$.
Takes $\Omega(n^2)$ to get n steps away.

Recall drunken sailor.

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Alternatively: $\mu = \min_{x \perp 1} \frac{x^T (L/d) x}{x^t x}$.

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Alternatively: $\mu = \min_{x \perp 1} \frac{x^T (L/d) x}{x^t x}$.

$L = dI - A$ or $L/d = I - A/d = I - M$.

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Alternatively: $\mu = \min_{x \perp 1} \frac{x^T (L/d) x}{x^t x}$.

$L = dI - A$ or $L/d = I - A/d = I - M$.

Also:

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Alternatively: $\mu = \min_{x \perp 1} \frac{x^T (L/d) x}{x^t x}$.

$L = dI - A$ or $L/d = I - A/d = I - M$.

Also:

$x^T L x = \sum_i d x_i^2 - \sum_{e=(i,j)} 2 x_i x_j$

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Alternatively: $\mu = \min_{x \perp 1} \frac{x^T (L/d) x}{x^t x}$.

$L = dI - A$ or $L/d = I - A/d = I - M$.

Also:

$$\begin{aligned} x^T L x &= \sum_i d x_i^2 - \sum_{e=(i,j)} 2 x_i x_j \\ &= \sum_{e=(i,j)} (x_i^2 + x_j^2 - 2 x_i x_j) \end{aligned}$$

Other views

Quotient Rayleigh: $\max_{x \perp 1} \frac{x^T M x}{x^t x}$.

Alternatively: $\mu = \min_{x \perp 1} \frac{x^T (L/d) x}{x^t x}$.

$L = dI - A$ or $L/d = I - A/d = I - M$.

Also:

$$\begin{aligned} x^T L x &= \sum_i d x_i^2 - \sum_{e=(i,j)} 2 x_i x_j \\ &= \sum_{e=(i,j)} (x_i^2 + x_j^2 - 2 x_i x_j) \\ &= \sum_{e=(i,j)} (x_i - x_j)^2. \end{aligned}$$

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T L x \geq 1 \sum_i (x_i - x_{i+1})^2$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T L x \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T L x \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Cauchy-Schwartz $|a||b| \geq a \cdot b$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T Lx \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Cauchy-Schwartz $|a||b| \geq a \cdot b$.

Yields: $\sum_i (x_i - x_{i+1})^2 \geq (x_1 - x_n)^2/n$

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T Lx \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Cauchy-Schwartz $|a||b| \geq a \cdot b$.

Yields: $\sum_i (x_i - x_{i+1})^2 \geq (x_1 - x_n)^2/n$

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T L x \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Cauchy-Schwartz $|a||b| \geq a \cdot b$.

Yields: $\sum_i (x_i - x_{i+1})^2 \geq (x_1 - x_n)^2/n$

Furthermore: $\sum_i x_i^2 \leq n(x_1^2 + x_n^2) \leq n(x_1 - x_n)^2$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T L x \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Cauchy-Schwartz $|a||b| \geq a \cdot b$.

Yields: $\sum_i (x_i - x_{i+1})^2 \geq (x_1 - x_n)^2/n$

Furthermore: $\sum_i x_i^2 \leq n(x_1^2 + x_n^2) \leq n(x_1 - x_n)^2$.

Warmup exercise: no larger eigenvalue on cycle.

Consider x for path.

Sort by x-value: $x_1, \ldots, x_n$.

Let $x_1^2 + x_n^2 = 1$. Shift so that $x_{n/2} = 0$.

Now, since any cut ≥ 1
$x^T L x \geq 1 \sum_i (x_i - x_{i+1})^2$.

Take $a = (x_i - x_{i+1})$ and $b = 1/\sqrt{n}$.

Cauchy-Schwartz $|a||b| \geq a \cdot b$.

Yields: $\sum_i (x_i - x_{i+1})^2 \geq (x_1 - x_n)^2/n$

Furthermore: $\sum_i x_i^2 \leq n(x_1^2 + x_n^2) \leq n(x_1 - x_n)^2$.

$\rightarrow \mu = \frac{x^T L x}{x^T x} \geq 1/n^2$.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2$

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1)x_i^2 - 2x_i \sum_{j \neq i} x_j)$.

$-2x_i \sum_{j \neq i} x_j) = -2x_i((\sum_j x_j) - x_i) = 2x_i^2 + x_i \sum_i x_j = 2x_i^2$

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

$-2 x_i \sum_{j \neq i} x_j) = -2 x_i ((\sum_j x_j) - x_i) = 2 x_i^2 + x_i \sum_i x_j = 2 x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

$-2 x_i \sum_{j \neq i} x_j) = -2 x_i ((\sum_j x_j) - x_i) = 2 x_i^2 + x_i \sum_i x_j = 2 x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij} (x_i - x_j)^2$

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

$-2 x_i \sum_{j \neq i} x_j) = -2 x_i ((\sum_j x_j) - x_i) = 2 x_i^2 + x_i \sum_i x_j = 2 x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij} (x_i - x_j)^2$

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij} (x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1)x_i^2 - 2x_i \sum_{j \neq i} x_j)$.

$-2x_i \sum_{j \neq i} x_j) = -2x_i((\sum_j x_j) - x_i) = 2x_i^2 + x_i \sum_i x_j = 2x_i^2$

$\sum_i nx_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij}(x_i - x_j)^2$

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$

Ratio of average edge “length” to pair length.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1)x_i^2 - 2x_i \sum_{j \neq i} x_j)$.

$-2x_i \sum_{j \neq i} x_j) = -2x_i((\sum_j x_j) - x_i) = 2x_i^2 + x_i \sum_i x_j = 2x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij}(x_i - x_j)^2$

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$

Ratio of average edge "length" to pair length.

Conductance $\phi(G) = n \frac{E(S, \overline{S})}{|S||V-S|}$.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

$-2 x_i \sum_{j \neq i} x_j) = -2 x_i ((\sum_j x_j) - x_i) = 2 x_i^2 + x_i \sum_i x_j = 2 x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij} (x_i - x_j)^2$

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij} (x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$

Ratio of average edge “length” to pair length.

Conductance $\phi(G) = n \frac{E(S, \overline{S})}{|S||V - S|}$.

Idea: pick random threshold.

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

$-2 x_i \sum_{j \neq i} x_j) = -2 x_i ((\sum_j x_j) - x_i) = 2 x_i^2 + x_i \sum_i x_j = 2 x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij} (x_i - x_j)^2$

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij} (x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$

Ratio of average edge "length" to pair length.

Conductance $\phi(G) = n \frac{E(S, \overline{S})}{|S||V-S|}$.

Idea: pick random threshold.
  Average number of edges cut compared to pairs is μ?

Interpretation of quadratic form

Consider $\mu = \min_{x \perp 1} \frac{x^T L x}{d x^T x}$.

$\sum_i x_i = 0$

Claim: $n \sum_i x_i^2 = \frac{1}{n} \sum_{i \leq j} (x_i - x_j)^2$

$\sum_i (2(n-1) x_i^2 - 2 x_i \sum_{j \neq i} x_j)$.

$-2 x_i \sum_{j \neq i} x_j) = -2 x_i ((\sum_j x_j) - x_i) = 2 x_i^2 + x_i \sum_i x_j = 2 x_i^2$

$\sum_i n x_i^2 - (\sum_i x_i)(\sum_j x_j)$.

Claim: $x^L x = \sum_{e=(i,j)} M_{ij} (x_i - x_j)^2$

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij} (x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$

Ratio of average edge “length” to pair length.

Conductance $\phi(G) = n \frac{E(S, \overline{S})}{|S||V - S|}$.

Idea: pick random threshold.
  Average number of edges cut compared to pairs is μ?

Random threshold.

$$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$$

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.
True if “length” corresponded to $|x_i - x_j|$.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.
True if "length" corresponded to $|x_i - x_j|$.
Off when length is $(x_i - x_j)^2$.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.
True if "length" corresponded to $|x_i - x_j|$.
Off when length is $(x_i - x_j)^2$.

Off by average length of edges.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.
True if "length" corresponded to $|x_i - x_j|$.

Off when length is $(x_i - x_j)^2$.

Off by average length of edges.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.
True if "length" corresponded to $|x_i - x_j|$.
Off when length is $(x_i - x_j)^2$.

Off by average length of edges.
For path:
edges have $|x_i - x_j|$ being $1/n$ of that for pairs.

Random threshold.

$\mu = \min_{x \perp 1} \frac{\sum_e M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$

Conductance $\phi(G) = n\frac{E(S,\overline{S})}{|S||V-S|}$.

Idea: pick random threshold.

Average number of edges cut compared to pairs is μ.
True if "length" corresponded to $|x_i - x_j|$.
Off when length is $(x_i - x_j)^2$.

Off by average length of edges.
For path:
edges have $|x_i - x_j|$ being $1/n$ of that for pairs.
For expanders:
edges go a constant fraction of what random pairs do.

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Take the 2^{nd} eigenvector $x = \text{argmin}_{x \in \mathbb{R}^V - \text{Span}\{\mathbf{1}\}} \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}$

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Take the 2^{nd} eigenvector $x = \text{argmin}_{x \in \mathbb{R}^V - \text{Span}\{\mathbf{1}\}} \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}$

Consider x as an embedding of the vertices to the real line.

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Take the 2^{nd} eigenvector $x = \text{argmin}_{x \in \mathbb{R}^V - \text{Span}\{\mathbf{1}\}} \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}$

Consider x as an embedding of the vertices to the real line.

Round x to get a $x \in \{0,1\}^V$

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Take the 2^{nd} eigenvector $x = \text{argmin}_{x \in \mathbb{R}^V - \text{Span}\{\mathbf{1}\}} \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}$

Consider x as an embedding of the vertices to the real line.

Round x to get a $x \in \{0,1\}^V$

Rounding: Take a threshold t,

$$\begin{cases} x_i \geq t & \rightarrow x_i = 1 \\ x_i < t & \rightarrow x_i = 0 \end{cases}$$

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Take the 2^{nd} eigenvector $x = \text{argmin}_{x \in \mathbb{R}^V - \text{Span}\{\mathbf{1}\}} \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}$

Consider x as an embedding of the vertices to the real line.

Round x to get a $x \in \{0,1\}^V$

Rounding: Take a threshold t,

$$\begin{cases} x_i \geq t & \to x_i = 1 \\ x_i < t & \to x_i = 0 \end{cases}$$

What will be a good t?

Cheeger Hard Part.

Now let's get to the hard part of Cheeger $h(G) \leq \sqrt{2(1-\lambda_2)}$.

Idea: We have $1-\lambda_2$ as a continuous relaxation of $\phi(G)$

Take the 2^{nd} eigenvector $x = \text{argmin}_{x \in \mathbb{R}^V - \text{Span}\{\mathbf{1}\}} \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}$

Consider x as an embedding of the vertices to the real line.

Round x to get a $x \in \{0,1\}^V$

Rounding: Take a threshold t,

$$\begin{cases} x_i \geq t & \rightarrow x_i = 1 \\ x_i < t & \rightarrow x_i = 0 \end{cases}$$

What will be a good t?

We don't know. Try all possible thresholds ($n-1$ possibilities), and hope there is a t leading to a good cut!

Sweep Cut Algorithm

Input: $G = (V, E)$, $x \in \mathbb{R}^V, x \perp \mathbf{1}$

Sweep Cut Algorithm

Input: $G = (V, E)$, $x \in \mathbb{R}^V, x \perp \mathbf{1}$

Sort the vertices in non-decreasing order with respect to x
WLOG $V = \{1, \ldots, n\}$ $\quad x_1 \leq x_2 \leq \ldots \leq x_n$

Sweep Cut Algorithm

Input: $G = (V, E)$, $x \in \mathbb{R}^V, x \perp \mathbf{1}$

Sort the vertices in non-decreasing order with respect to x
WLOG $V = \{1, \ldots, n\}$ $\qquad x_1 \leq x_2 \leq \ldots \leq x_n$

Let $S_i = \{1, \ldots, i\}$ $\qquad i = 1, \ldots, n-1$

Sweep Cut Algorithm

Input: $G = (V, E)$, $x \in \mathbb{R}^V, x \perp \mathbf{1}$

Sort the vertices in non-decreasing order with respect to x
WLOG $V = \{1, \ldots, n\}$ $\qquad x_1 \leq x_2 \leq \ldots \leq x_n$

Let $S_i = \{1, \ldots, i\}$ $\qquad i = 1, \ldots, n-1$

Return $S = \text{argmin}_{S_i} h(S_i)$

Sweep Cut Algorithm

Input: $G = (V, E)$, $x \in \mathbb{R}^V, x \perp \mathbf{1}$

Sort the vertices in non-decreasing order with respect to x
WLOG $V = \{1, \ldots, n\}$ $\quad x_1 \leq x_2 \leq \ldots \leq x_n$

Let $S_i = \{1, \ldots, i\}$ $\quad i = 1, \ldots, n-1$

Return $S = \text{argmin}_{S_i} h(S_i)$

Main Lemma: $G = (V, E)$, d-regular

$$x \in \mathbb{R}^V, x \perp \mathbf{1}, \mu = \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i - x_j)^2}$$

Sweep Cut Algorithm

Input: $G = (V, E)$, $x \in \mathbb{R}^V, x \perp \mathbf{1}$

Sort the vertices in non-decreasing order with respect to x
WLOG $V = \{1, \ldots, n\}$ $\quad x_1 \leq x_2 \leq \ldots \leq x_n$

Let $S_i = \{1, \ldots, i\}$ $\quad i = 1, \ldots, n-1$

Return $S = \mathrm{argmin}_{S_i} h(S_i)$

Main Lemma: $G = (V, E)$, d-regular

$$x \in \mathbb{R}^V, x \perp \mathbf{1}, \mu = \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{d}{n} \sum_{i,j} (x_i - x_j)^2}$$

If S is the ouput of the sweep cut algorithm, then $h(S) \leq \sqrt{2\mu}$

Sweep Cut Algorithm

Input: $G=(V,E)$, $x\in\mathbb{R}^V, x\perp \mathbf{1}$

Sort the vertices in non-decreasing order with respect to x
WLOG $V=\{1,\ldots,n\}$ $\quad x_1\le x_2\le\ldots\le x_n$

Let $S_i=\{1,\ldots,i\}$ $\quad i=1,\ldots,n-1$

Return $S=\text{argmin}_{S_i} h(S_i)$

Main Lemma: $G=(V,E)$, d-regular

$$x\in\mathbb{R}^V, x\perp \mathbf{1}, \mu=\frac{\sum_{i,j}M_{ij}(x_i-x_j)^2}{\frac{d}{n}\sum_{i,j}(x_i-x_j)^2}$$

If S is the ouput of the sweep cut algorithm, then $h(S)\le\sqrt{2\mu}$

Note: Applying the Main Lemma with the 2^{nd} eigenvector v_2, we have $\mu=1-\lambda_2$, and $h(G)\le h(S)\le\sqrt{2(1-\lambda_2)}$. Done!

Proof of Main Lemma

WLOG $V = \{1, \ldots, n\}$ $\qquad x_1 \leq x_2 \leq \ldots \leq x_n$

Proof of Main Lemma

WLOG $V = \{1, \ldots, n\}$ $\qquad x_1 \leq x_2 \leq \ldots \leq x_n$

Want to show

$$\exists i \text{ s.t. } h(S_i) = \frac{\frac{1}{d}|E(S_i, V - S_i)|}{\min(|S_i|, |V - S_i|)} \leq \sqrt{2\mu}$$

Proof of Main Lemma

WLOG $V = \{1, \ldots, n\}$ $\quad x_1 \leq x_2 \leq \ldots \leq x_n$

Want to show

$$\exists i \text{ s.t. } h(S_i) = \frac{\frac{1}{d}|E(S_i, V - S_i)|}{\min(|S_i|, |V - S_i|)} \leq \sqrt{2\mu}$$

Probabilistic Argument: Construct a distribution D over $\{S_1, \ldots, S_{n-1}\}$ such that

$$\frac{\mathbb{E}_{S \sim D}[\frac{1}{d}|E(S, V - S)|]}{\mathbb{E}_{S \sim D}[\min(|S|, |V - S|)]} \leq \sqrt{2\mu}$$

Proof of Main Lemma

WLOG $V = \{1, \ldots, n\}$ $\quad x_1 \leq x_2 \leq \ldots \leq x_n$

Want to show

$$\exists i \text{ s.t. } h(S_i) = \frac{\frac{1}{d}|E(S_i, V - S_i)|}{\min(|S_i|, |V - S_i|)} \leq \sqrt{2\mu}$$

Probabilistic Argument: Construct a distribution D over $\{S_1, \ldots, S_{n-1}\}$ such that

$$\frac{\mathbb{E}_{S \sim D}[\frac{1}{d}|E(S, V - S)|]}{\mathbb{E}_{S \sim D}[\min(|S|, |V - S|)]} \leq \sqrt{2\mu}$$

$$\rightarrow \mathbb{E}_{S \sim D}[\tfrac{1}{d}|E(S, V - S)| - \sqrt{2\mu} \min(|S|, |V - S|)] \leq 0$$

Proof of Main Lemma

WLOG $V = \{1, \ldots, n\}$ $\quad x_1 \leq x_2 \leq \ldots \leq x_n$

Want to show

$$\exists i \text{ s.t. } h(S_i) = \frac{\frac{1}{d}|E(S_i, V - S_i)|}{\min(|S_i|, |V - S_i|)} \leq \sqrt{2\mu}$$

Probabilistic Argument: Construct a distribution D over $\{S_1, \ldots, S_{n-1}\}$ such that

$$\frac{\mathbb{E}_{S \sim D}[\frac{1}{d}|E(S, V - S)|]}{\mathbb{E}_{S \sim D}[\min(|S|, |V - S|)]} \leq \sqrt{2\mu}$$

$$\rightarrow \mathbb{E}_{S \sim D}[\tfrac{1}{d}|E(S, V - S)| - \sqrt{2\mu}\min(|S|, |V - S|)] \leq 0$$

$$\exists S \qquad \tfrac{1}{d}|E(S, V - S)| - \sqrt{2\mu}\min(|S|, |V - S|) \leq 0$$

The distribution D

WLOG, shift and scale so that $x_{\lfloor \frac{n}{2} \rfloor} = 0$, and $x_1^2 + x_n^2 = 1$

The distribution D

WLOG, shift and scale so that $x_{\lfloor \frac{n}{2} \rfloor} = 0$, and $x_1^2 + x_n^2 = 1$

Take t from the range $[x_1, x_n]$ with density function $f(t) = 2|t|$.

The distribution D

WLOG, shift and scale so that $x_{\lfloor \frac{n}{2} \rfloor} = 0$, and $x_1^2 + x_n^2 = 1$

Take t from the range $[x_1, x_n]$ with density function $f(t) = 2|t|$.

Check: $\int_{x_1}^{x_n} f(t)\mathrm{d}t = \int_{x_1}^{0} -2t\mathrm{d}t + \int_{0}^{x_n} 2t\mathrm{d}t = x_1^2 + x_n^2 = 1$

The distribution D

WLOG, shift and scale so that $x_{\lfloor \frac{n}{2} \rfloor} = 0$, and $x_1^2 + x_n^2 = 1$

Take t from the range $[x_1, x_n]$ with density function $f(t) = 2|t|$.

Check: $\int_{x_1}^{x_n} f(t)\mathrm{d}t = \int_{x_1}^{0} -2t\mathrm{d}t + \int_{0}^{x_n} 2t\mathrm{d}t = x_1^2 + x_n^2 = 1$

$S = \{i : x_i \leq t\}$

The distribution D

WLOG, shift and scale so that $x_{\lfloor \frac{n}{2} \rfloor} = 0$, and $x_1^2 + x_n^2 = 1$

Take t from the range $[x_1, x_n]$ with density function $f(t) = 2|t|$.

Check: $\int_{x_1}^{x_n} f(t)\mathrm{d}t = \int_{x_1}^{0} -2t\mathrm{d}t + \int_{0}^{x_n} 2t\mathrm{d}t = x_1^2 + x_n^2 = 1$

$S = \{i : x_i \leq t\}$

Let D be distribution over $S_1, \ldots, S_{n-1}$ from the above process.

Denominator.

Goal: $\frac{\mathbb{E}_{S \sim D}[\frac{1}{d}|E(S, V-S)|]}{\mathbb{E}_{S \sim D}[min(|S|, |V-S|)]} \leq \sqrt{2\mu}$

Denominator.

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Denominator:

Denominator.

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Denominator:

Let $T_i =$ indicator for "i is in the smaller set of $S, V-S$"

Denominator.

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Denominator:

Let $T_i =$ indicator for "i is in the smaller set of $S, V-S$"

Can check

$$\mathbb{E}_{S\sim D}[T_i] = Pr[T_i = 1] = x_i^2$$

Denominator.

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Denominator:

Let $T_i =$ indicator for "i is in the smaller set of $S, V-S$"

Can check

$$\mathbb{E}_{S\sim D}[T_i] = Pr[T_i = 1] = x_i^2$$

$$\begin{aligned}
\mathbb{E}_{S\sim D}[min(|S|,|V-S|)] &= \mathbb{E}_{S\sim D}[\sum_i T_i] \\
&= \sum_i \mathbb{E}_{S\sim D}[T_i] \\
&= \sum_i x_i^2
\end{aligned}$$

Numerator

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

Numerator

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

Let $T_{i,j} =$ indicator for i, j is cut by $(S, V-S)$

Numerator

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

Let $T_{i,j} =$ indicator for i,j is cut by $(S, V-S)$

$$\begin{cases} x_i, x_j \text{ same sign:} & Pr[T_{i,j} = 1] = |x_i^2 - x_j^2| \\ x_i, x_j \text{ different sign:} & Pr[T_{i,j} = 1] = x_i^2 + x_j^2 \end{cases}$$

Numerator

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

Let $T_{i,j} =$ indicator for i,j is cut by $(S, V-S)$

$$\begin{cases} x_i, x_j \text{ same sign:} & Pr[T_{i,j}=1] = |x_i^2 - x_j^2| \\ x_i, x_j \text{ different sign:} & Pr[T_{i,j}=1] = x_i^2 + x_j^2 \end{cases}$$

A common upper bound: $\mathbb{E}[T_{i,j}] = Pr[T_{i,j}=1] \leq |x_i - x_j|(|x_i| + |x_j|)$

Numerator

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

Let $T_{i,j} =$ indicator for i, j is cut by $(S, V - S)$

$$\begin{cases} x_i, x_j \text{ same sign:} & Pr[T_{i,j} = 1] = |x_i^2 - x_j^2| \\ x_i, x_j \text{ different sign:} & Pr[T_{i,j} = 1] = x_i^2 + x_j^2 \end{cases}$$

A common upper bound: $\mathbb{E}[T_{i,j}] = Pr[T_{i,j} = 1] \leq |x_i - x_j|(|x_i| + |x_j|)$

$$\begin{aligned} \mathbb{E}_{S\sim D}[\frac{1}{d}|E(S, V - S)|] &= \frac{1}{2}\sum_{i,j} M_{ij}\mathbb{E}[T_{i,j}] \\ &\leq \frac{1}{2}\sum_{i,j} M_{ij}|x_i - x_j|(|x_i| + |x_j|) \end{aligned}$$

Cauchy-Schwarz Inequality

$|a \cdot b| \leq \|a\|\|b\|$, as $a \cdot b = \|a\|\|b\| cos(a, b)$

Cauchy-Schwarz Inequality

$|a \cdot b| \leq \|a\|\|b\|$, as $a \cdot b = \|a\|\|b\| cos(a, b)$

Applying with $a, b \in \mathbb{R}^{n^2}$ with $a_{ij} = \sqrt{M_{ij}}|x_i - x_j|, b_{ij} = \sqrt{M_{ij}}|x_i| + |x_j|$

Cauchy-Schwarz Inequality

$|a \cdot b| \leq \|a\|\|b\|$, as $a \cdot b = \|a\|\|b\| cos(a, b)$

Applying with $a, b \in \mathbb{R}^{n^2}$ with $a_{ij} = \sqrt{M_{ij}}|x_i - x_j|, b_{ij} = \sqrt{M_{ij}}|x_i| + |x_j|$

Numerator:

$$
\begin{aligned}
\mathbb{E}_{S \sim D}[\frac{1}{d}|E(S, V-S)|] &= \frac{1}{2}\sum_{i,j} M_{ij}\mathbb{E}[T_{i,j}] \\
&\leq \frac{1}{2}\sum_{i,j} M_{ij}|x_i - x_j|(|x_i| + |x_j|) \\
&= \frac{1}{2} a \cdot b \\
&\leq \frac{1}{2}\|a\|\|b\|
\end{aligned}
$$

Simplify numerator.

Recall $\mu = \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}, a_{ij} = \sqrt{M_{ij}}|x_i - x_j|, b_{ij} = \sqrt{M_{ij}}(|x_i| + |x_j|)$

Simplify numerator.

Recall $\mu = \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}, a_{ij} = \sqrt{M_{ij}}|x_i - x_j|, b_{ij} = \sqrt{M_{ij}}(|x_i| + |x_j|)$

$$\begin{aligned}
\|a\|^2 &= \sum_{i,j} M_{ij}(x_i - x_j)^2 = \frac{\mu}{n}\sum_{i,j}(x_i - x_j)^2 \\
&= \frac{\mu}{n}\sum_{i,j}(x_i^2 + x_j^2) - \sum_{i,j} 2x_i x_j \\
&= \frac{\mu}{n}\sum_{i,j}(x_i^2 + x_j^2) - 2(\sum_i x_i)^2 \\
&\leq \frac{\mu}{n}\sum_{i,j}(x_i^2 + x_j^2) = 2\mu\sum_i x_i^2
\end{aligned}$$

Simplify numerator.

Recall $\mu = \frac{\sum_{i,j} M_{ij}(x_i - x_j)^2}{\frac{1}{n}\sum_{i,j}(x_i - x_j)^2}, a_{ij} = \sqrt{M_{ij}}|x_i - x_j|, b_{ij} = \sqrt{M_{ij}}(|x_i| + |x_j|)$

$$\begin{aligned}
\|a\|^2 &= \sum_{i,j} M_{ij}(x_i - x_j)^2 = \frac{\mu}{n}\sum_{i,j}(x_i - x_j)^2 \\
&= \frac{\mu}{n}\sum_{i,j}(x_i^2 + x_j^2) - \sum_{i,j} 2x_i x_j \\
&= \frac{\mu}{n}\sum_{i,j}(x_i^2 + x_j^2) - 2(\sum_i x_i)^2 \\
&\leq \frac{\mu}{n}\sum_{i,j}(x_i^2 + x_j^2) = 2\mu \sum_i x_i^2
\end{aligned}$$

$$\begin{aligned}
\|b\|^2 &= \sum_{i,j} M_{ij}(|x_i| + |x_j|)^2 \\
&\leq \sum_{i,j} M_{ij}(2x_i^2 + 2x_j^2) \\
&= 4\sum_i x_i^2
\end{aligned}$$

Put together.

Goal: $\frac{\mathbb{E}_{S \sim D}[\frac{1}{d}|E(S, V-S)|]}{\mathbb{E}_{S \sim D}[min(|S|, |V-S|)]} \leq \sqrt{2\mu}$

Put together.

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

$$\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|] = \leq \frac{1}{2}\|a\|\|b\|$$

$$\leq \frac{1}{2}\sqrt{2\mu\sum_i x_i^2}\sqrt{4\sum_i x_i^2} \quad = \sqrt{2\mu}\sum_i x_i^2$$

Recall **Denominator:**

$$\mathbb{E}_{S\sim D}[min(|S|,|V-S|)] = \sum_i x_i^2$$

We get

$$\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$$

Put together.

Goal: $\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$

Numerator:

$$\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|] = \leq \frac{1}{2}\|a\|\|b\|$$
$$\leq \frac{1}{2}\sqrt{2\mu\sum_i x_i^2}\sqrt{4\sum_i x_i^2} \qquad = \sqrt{2\mu}\sum_i x_i^2$$

Recall **Denominator:**

$$\mathbb{E}_{S\sim D}[min(|S|,|V-S|)] = \sum_i x_i^2$$

We get

$$\frac{\mathbb{E}_{S\sim D}[\frac{1}{d}|E(S,V-S)|]}{\mathbb{E}_{S\sim D}[min(|S|,|V-S|)]} \leq \sqrt{2\mu}$$

Thus $\exists S_i$ such that $h(S_i) \leq \sqrt{2\mu}$, which gives $h(G) \leq \sqrt{2(1-\lambda)}$ $\square$

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S|$

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S| \implies$ flow of value $h(G)|S|$.
Max flow-min cut theorem.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S| \implies$ flow of value $h(G)|S|$.
Max flow-min cut theorem.

From path argument: $\implies \mu \geq \frac{1}{\ell^2} = h(G)^2$.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S| \implies$ flow of value $h(G)|S|$.
Max flow-min cut theorem.

From path argument: $\implies \mu \geq \frac{1}{\ell^2} = h(G)^2$.

Run argument over sets of size 2^i and one gets the upper bound.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S| \implies$ flow of value $h(G)|S|$.
  Max flow-min cut theorem.

From path argument: $\implies \mu \geq \frac{1}{\ell^2} = h(G)^2$.

Run argument over sets of size 2^i and one gets the upper bound.

Why no $\log n$ factor?

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S| \implies$ flow of value $h(G)|S|$.
  Max flow-min cut theorem.

From path argument: $\implies \mu \geq \frac{1}{\ell^2} = h(G)^2$.

Run argument over sets of size 2^i and one gets the upper bound.

Why no $\log n$ factor?   The mass splits, and **every level** has $\mu_i \geq h(G)^2$.

Kind of a proof.

$G = (V, E)$, $h = h(G)$.

Claim:
From $S \subset V$ of vertices, $|E(N_{1/h}(S))| \geq 2|E(S)|$.

Claim': there are $\Omega(h|S|)$ paths of length $\ell = 1/h$ in $N_{1/h}(S)$.

Cut size is $\geq h(G)|S| \implies$ flow of value $h(G)|S|$.
Max flow-min cut theorem.

From path argument: $\implies \mu \geq \frac{1}{\ell^2} = h(G)^2$.

Run argument over sets of size 2^i and one gets the upper bound.

Why no $\log n$ factor? The mass splits, and **every level** has $\mu_i \geq h(G)^2$.

Cheeger proof magically does this!