The Multiway Cut Problem

Renato Werneck
(rwerneck@princeton.edu)

Main sources:

Gruia Calinescu, Howard Karloff and Yuval Rabani, "An Improved Approximation Algorithm for Multiway Cut", *Journal of Computer and System Sciences*, 60(3):564–574, 2000.

Vijay V. Vazirani, *Approximation Algorithms*, Springer, 2001.

Outline

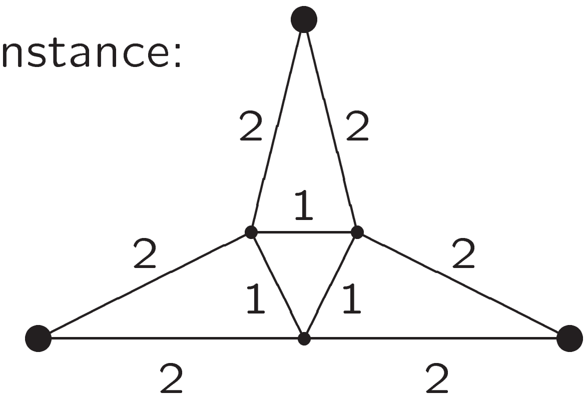
- Statement of the problem
- 2-approximation algorithm
- Basic LP relaxation
- Improved LP relaxation
- 1.5-approximation algorithm

The Multiway Cut Problem

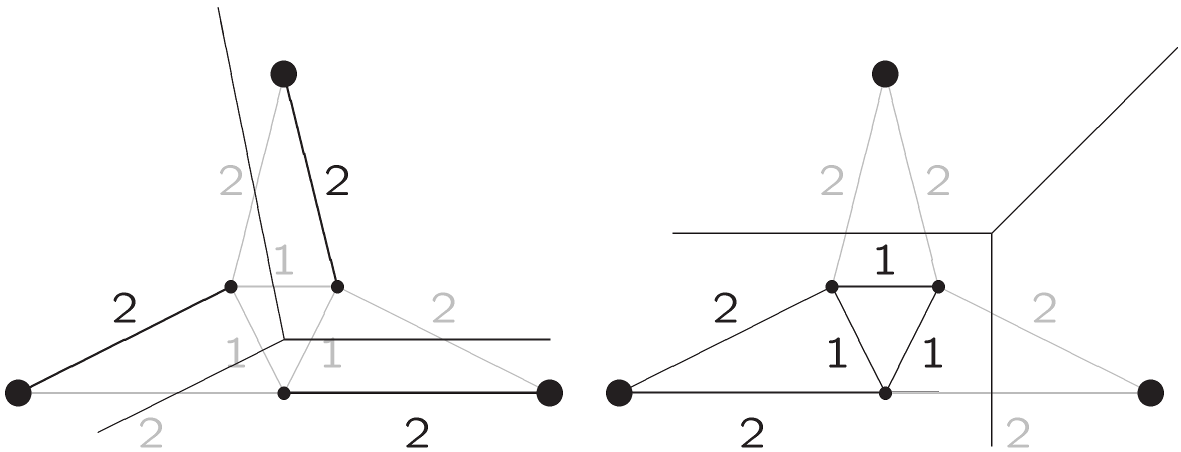
- Input:
  - an undirected graph $G = (V, E)$
  - a nonnegative cost function $c(u, v)$
  - a set $T \subseteq V$ of *terminals*
  - convention:
    * $V = \{1, 2, \ldots, n\}$
    * $T = \{1, 2, \ldots, k\}$
- Output:
  - a minimum cost set of edges $C \subseteq E$ such that each terminal is in a different component in $(V, E \setminus C)$
- This is *not* the "Multicut Problem":
  - given a graph and k *pairs* of vertices, find a minimum cost set of edges whose removal separates each pair;
  - best known approximation algorithm: $O(\log k)$.

Example

- Instance:



- Feasible solutions:



- This can be interpreted as a partition problem:
  - partition the vertices into k sets, each containing one terminal;
  - solution cost: sum of the cost of all edges between sets

A Simple Algorithm

- for every terminal t, find the minimum cut C_t (with cost $c(C_t)$) separating t from all other terminals;

- discard the cut u for which $c(C_u)$ is maximum;

- return the union of the $k-1$ remaning cuts.

A Simple Algorithm - Analysis

- Let A be the set of edges in the optimal solution.
- Remove all edges in A from the graph:
  - every terminal t will be in a different connected component;
  - let $A_t \subseteq A$ be the set of edges leaving the component where t is.
- Every edge appears in exactly two sets, so:

$$\sum_{i=1}^{t} c(A_t) = 2 \cdot c(A) = 2 \cdot OPT \tag{1}$$

- C_t is the minimum cut separating t from all other vertices. By definition:

$$c(C_t) \leq c(A_t) \tag{2}$$

- Using the fact that the heaviest cut is discarded,

$$\begin{aligned} c(C) &\leq \left(1 - \frac{1}{k}\right) \sum_{t=1}^{k} c(C_t) \\ &\leq \left(1 - \frac{1}{k}\right) \sum_{t=1}^{k} c(A_t) \\ &\leq \left(2 - \frac{2}{k}\right) OPT \end{aligned}$$

Semimetric

- A pair (V, d) is a semimetric if
  - V is a set ("points");
  - d is a function $d : V \times V \to \mathcal{R}$ ("distance") such that:
    * $d(u, v) \geq 0$, $\forall u, v \in V$;
    * $d(u, v) = d(v, u)$, $\forall u, v \in V$;
    * $d(u, w) \leq d(u, v) + d(v, w)$, $\forall u, v, w \in V$.

Semimetric-based IP Formulation

- Formulation:

$$
\begin{array}{lll}
\text{min} & \sum_{(u,v)\in E} c(u,v)d(u,v) & \\
\text{s.t.} & & \\
& (V,d) & \text{is a semimetric} \\
& d(t_1,t_2) = 1 & \forall t_1, t_2 \in T, t_1 \neq t_2 \\
& d(u,v) \in \{0,1\} & \forall u, v \in V
\end{array}
$$

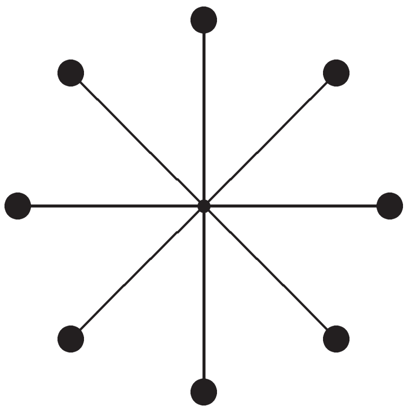
- $d(\cdot,\cdot)$ are the variables
- Intuition:
  - $d(u,v)$:
    - 0: u and v in the same component;
    - 1: u and v in different components.
  - distance between any two terminals must be 1.

Linear Relaxation

- Linear relaxation (LP1):

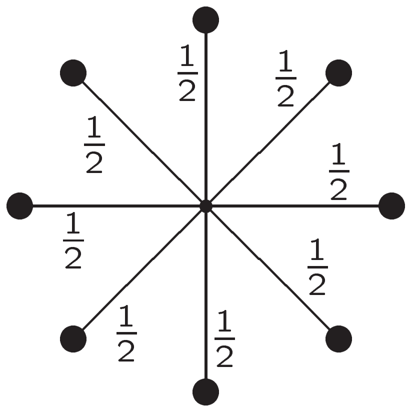
$$
\begin{array}{lrl}
\text{min} & \sum_{(u,v)\in E} c(u,v)d(u,v) & \\
\text{s.t.} & & \\
& (V,d) & \text{is a semimetric} \\
& d(t_1,t_2) = 1 & \forall t_1, t_2 \in T, t_1 \neq t_2 \\
& 0 \leq d(u,v) \leq 1 & \forall u, v \in V
\end{array}
$$

- Integrality ratio: $2\left(1 - \frac{1}{k}\right)$



k terminals,
one non-terminal
unit costs

optimal solution: $k-1$



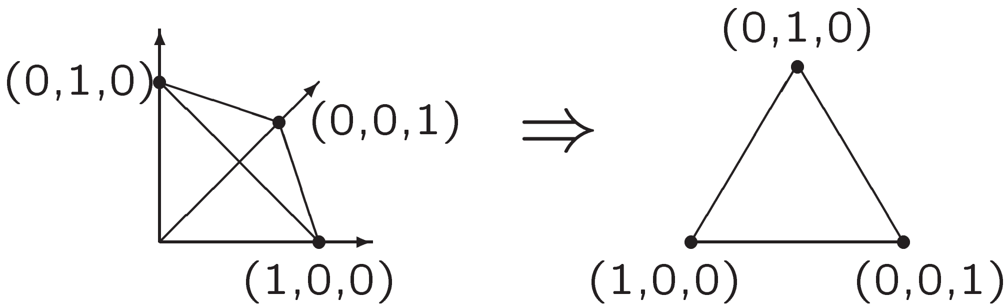
fractional solution: $k/2$

Simplex

- Δ_k denotes the $(k-1)$-*simplex*, defined by:

$$\{x \in \mathcal{R}^k | (x \geq 0) \text{ and } (\sum_{i=1}^{k} x_i = 1)\}$$

- Characteristics:
  - convex polytope in $\mathcal{R}^k$
  - $(k-1)$-dimensional
  - Example (Δ_3):



Linear Relaxation (SLP) — Intuition

- Every vertex v is associated with a k-dimensional vector x_v, with each component between 0 and 1. For $k = 11$, we would have:

$x_v =$

x_v^1	x_v^2	x_v^3	x_v^4	x_v^5	x_v^6	x_v^7	x_v^8	x_v^9	x_v^{10}	x_v^{11}

- Terminals:

$x_3 =$

0	0	1	0	0	0	0	0	0	0	0

$x_8 =$

0	0	0	0	0	0	0	1	0	0	0

- Non-terminals (integer case):
  - $x_v^t = 1 \Rightarrow v$ and t in the same component;
  - $x_v^t = 0 \Rightarrow v$ and t in different components;
  - Exactly one coordinate is 1, all others are 0.

- Non-terminals (linear relaxation):
  - $0 \leq x_v^t \leq 1$
  - $\sum_t x_v^t = 1$

$x_v =$

0	0.5	0	0	0	0	0	0.2	0	0	0.3

Linear Relaxation (SLP) — Intuition

- The *distance* between two vertices is defined as:

$$d(u,v) = \frac{1}{2}\sum_{t=1}^{k} |x_u^t - x_v^t|$$

  - $d(u,v) = 0 \Rightarrow u$ and v in the same component;
  - $d(u,v) = 1 \Rightarrow u$ and v in different components;
  - $0 < d(u,v) < 1 \Rightarrow$?

- Examples:

$x_2 =$	·	1	·	·	·	·	·	·	·	·	·
$x_7 =$	·	·	·	·	·	·	1	·	·	·	·
$x_{17} =$	·	0.3	·	·	·	·	·	0.4	·	·	0.3
$x_{39} =$	·	0.5	·	·	·	·	·	0.5	·	·	·
$x_{43} =$	·	1	·	·	·	·	·	·	·	·	·

$d(2,7) = 1$ $\qquad$ $d(2,17) = 0.7$

$d(2,43) = 0$ $\qquad$ $d(17,39) = 0.3$

Linear Relaxation (SLP)

- Formulation:

$$\begin{array}{ll} \text{min} & \sum_{(u,v)\in E} c(u,v)d(u,v) \\ \text{s.t.} & \\ & \begin{array}{rl} d(u,v) = \frac{1}{2}\sum_{t=1}^{k} |x_u^t - x_v^t| & \forall (u,v) \in E \\ x_u \in \Delta_k & 1 \le u \le n \\ x_t = e_t & 1 \le t \le k \end{array} \end{array}$$

- e_t is a unit k-dimensional vector with 1 at coordinate t and 0 elsewhere.

- Nodes are placed at the $(k-1)$-simplex:
  - terminals: vertices
  - non-terminals: anywhere (maybe inside)
  - integral solution $\Leftrightarrow$ all nodes at vertices

Is SLP Linear?

- $x_t = e_t$: clearly linear

- $x_u \in \Delta_k$ actually means:

$$\begin{aligned} x_u^i &\geq 0, \quad \text{for } i = 1, 2, \ldots, k \\ \sum_{i=1}^{k} x_u^i &= 1 \end{aligned}$$

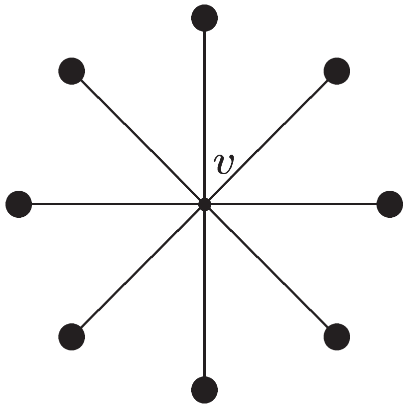
- $d(u,v) = \frac{1}{2}\sum_{t=1}^{k} |x_u^t - x_v^t|$ can be written as:

$$\begin{aligned} x_{uv}^t &\geq x_u^t - x_v^t \\ x_{uv}^t &\geq x_v^t - x_u^t \\ d(u,v) &= \frac{1}{2}\sum_{t=1}^{k} x_{uv}, \end{aligned}$$

where x_{uv}^t (with $1 \leq t \leq k$) are new variables.

SLP relaxation

- SLP is stronger than LP1
- Example:



- For any terminal t, $d(v,t)$ is:

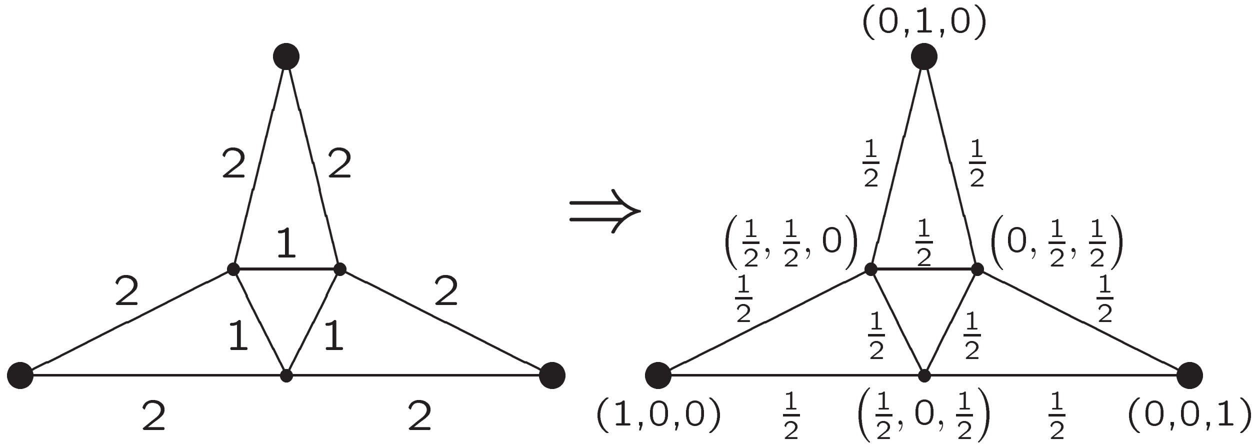
$$
\begin{aligned}
d(v,t) &= \frac{1}{2}\left[(1 - x_v^t) + \sum_{i \neq t} x_v^i\right] \\
d(v,t) &= \frac{1}{2}\left[(1 - 2x_v^t) + \sum_{i=1}^{k} x_v^i\right] \\
d(v,t) &= \frac{1}{2}[(1 - 2x_v^t) + 1] \\
d(v,t) &= 1 - x_v^t
\end{aligned}
$$

- Solution value:

$$\sum_{t=1}^{k} c(v,t)d(v,t) = \sum_{t=1}^{k}(1)(1 - x_v^t) = k - \sum_{t=1}^{k} x_v^t = k - 1$$

Integrality Ratio

- Example (integrality ratio $= \frac{16}{15}$):



edge costs
integer solution: 8

x values and distances
fractional solution: 7.5

- Karger et al. (STOC'99)
  - worst case for $k = 3$: 12/11
  - best known upper bound (for any k): ~1.3438

Solution Normalization

Lemma: Every solution to SLP can be transformed into a solution in which, for any u and v, x_u and x_v differ in at most two positions.

- For every edge (u, v) that violates this:
  - create a new vertex w between u and v;
  - ensure that the integer solution is preserved:
    * set $c(u, w) \leftarrow c(u, v)$ and $c(w, v) \leftarrow c(u, v)$
  - ensure that the fractional solution is preserved:
    * Let i be the coordinate in which the difference is minimum (but non-zero).
    * Assume $x_u^i < x_v^i$, and let $\alpha = x_v^i - x_u^i$.
    * Let j be such that $x_u^j \geq x_v^j + \alpha$ (it exists).
    * Define x_w:
      · $x_w^i \leftarrow x_v^i - \alpha$ (same as x_u^i);
      · $x_w^j \leftarrow x_v^j + \alpha$;
      · $x_w^t \leftarrow x_v^t, \forall t \notin \{i, j\}$.
    * The fractional solution is preserved:
      · $x_w \in \Delta_k$ (because $\sum_{t \in T} x_w^t = 1$);
      · $d(u, v) = d(u, w) + d(w, v)$.

Normalization

$$x_u = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|}\hline a & b & c & d & e & f & g & h & i & j & k \\ \hline\end{array}$$

$$x_v = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|}\hline a & b & c & d & e & f^* & g^* & h^* & i^* & j^* & k^* \\ \hline\end{array}$$

x_u and x_v differ in 6 coordinates

$$\begin{array}{r|c|c|c|c|c|c|c|c|c|c|c|}
\cline{2-12}
x_u = & a & b & c & d & e & f & g & h & i & j & k \\
\cline{2-12}
\multicolumn{1}{r}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\downarrow} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\
\cline{2-12}
x_w = & a & b & c & d & e & f^* & g^* & h^* & i & j^{**} & k^* \\
\cline{2-12}
\multicolumn{1}{r}{} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow\text{-}\alpha} & \multicolumn{1}{c}{\uparrow\text{+}\alpha} & \multicolumn{1}{c}{\uparrow} \\
\cline{2-12}
x_v = & a & b & c & d & e & f^* & g^* & h^* & i^* & j^* & k^* \\
\cline{2-12}
\end{array}$$

x_u and x_w differ in (at most) 5 coordinates

x_w and x_v differ in 2 coordinates

Some Definitions

- x = optimal, normalized solution of SLP, with value OPT_f.

- $E_t = \{(u,v) \in E \mid x_u^t \neq x_v^t\}$
  - edges between vertices that differ in coord. t
  - each edge (u,v) appears in two of these sets

- $W_t = \sum_{e \in E_t} c(e)d(e)$
  - cost of all edges in E_t, weighted by the distances
  - "contribution" of E_t to the solution value
  - each edge with $d(e) > 0$ is counted twice

- $B(t,\rho) = \{v \in V \mid x_v^t \geq \rho\}$, for $\rho \in (0,1)$
  - set of all vertices that are "close" to terminal t
  - v is "close" to $i \Leftrightarrow x_v^t \geq \rho$

The Algorithm

1. Compute the optimal, normalized solution x of SLP;

2. Relabel the terminals so that W_k is the largest among all W_i;

3. Pick a random σ in $\{(1, 2, \ldots, k-1, k), (k-1, k-2, \ldots, 1, k)\}$;

4. Pick a random $\rho \in (0, 1)$;

5. For $i = 1$ to $k-1$ do $V_{\sigma(i)} \leftarrow B(i, \rho) - \bigcup_{j<i} V_{\sigma(j)}$;

6. $V_k \leftarrow V - \bigcup_{i<k} V_i$;

7. Return $C = \{(u, v) \in E \mid u \in V_i, v \in V_j, i \neq j\}$.

Analysis (1/4)

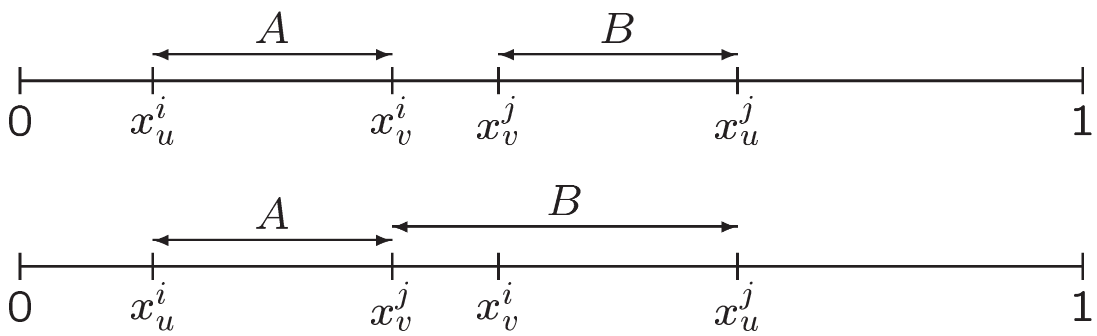
Lemma: If $e \in E - E_k$, Prob$[e \in C] \leq 1.5 \cdot d(e)$.

- Let $e = (u, v)$ and i and j be the coordinates in which x_u and x_v differ (if $x_u = x_v$, $d(u, v) = 0$ and we are done).

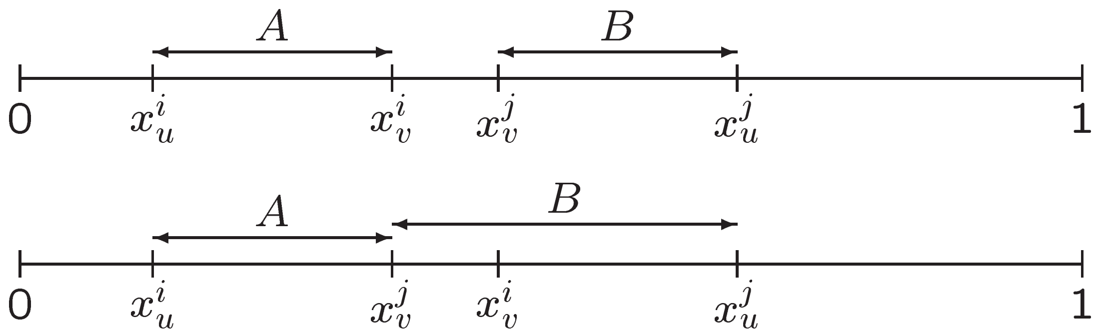
- Assume $x_u^i = \min\{x_u^i, x_u^j, x_v^i, x_v^j\}$. Facts:

  1. $x_u^i + x_u^j = x_v^i + x_v^j$ (x_u and x_v are in Δ_k)
  2. $x_u^j = \max\{x_u^i, x_u^j, x_v^i, x_v^j\}$ (from 1)
  3. $x_u^j - x_v^j = x_v^i - x_u^i$ (from 1)
  4. $x_u^j - x_v^j = x_v^i - x_u^i = d(u, v)$ (from def. of d)
  5. either $x_v^i \leq x_v^j$ or $x_v^i \geq x_v^j$

- There are two possible configurations:



Analysis (2/4)



- $(u, v) \in C$ in cut $\iff$ u and v in different sets
  - Case 1: $\rho \notin (A \cup B)$
    - $*$ u and v in the same set (S_k, S_i, or S_j)
  - Case 2: $\rho \in (A \cup B)$
    - $*$ $P_1 = \text{Prob}[\rho \in (A \cup B)] = |A \cup B| = |A| + |B|$
    - $*$ Special case: $\rho \in A$ and $\sigma(j) < \sigma(i)$
      - $\cdot$ both u and v will end up in S_j
      - $\cdot$ $P_2 = \text{Prob}[(\rho \in A) \wedge (\sigma(j) < \sigma(i))] = |A| \cdot \frac{1}{2}$
  - Probability that u and v end up in different sets:

$$\begin{aligned}
\text{Prob}[(u, v) \in C] &= P_1 - P_2 \\
&= (|A| + |B|) - |A|/2 \\
&= |A|/2 + |B| \\
&\leq d(u, v)/2 + d(u, v) \\
&= 1.5 \cdot d(u, v)
\end{aligned}$$

Analysis (3/4)

Lemma: If $e \in E_k$, Prob$[e \in C] \leq d(e)$.

- Let $e = (u, v)$ and i and k be the coordinates in which x_u and x_v differ.
  - $\sigma(i) < \sigma(k)$, by definition;
  - u and v will end up in different sets if ρ falls between x_u^i and x_v^i:

$$\text{Prob}[(u, v) \in C] = |x_u^i - x_v^i| = d(e)$$

Analysis (4/4)

$$
\begin{aligned}
E[c(C)] &= \sum_{e \in E} c(e)\text{Prob}[e \in C] \\
&= \sum_{e \in E - E_k} c(e)\text{Prob}[e \in C] + \sum_{e \in E_k} c(e)\text{Prob}[e \in C] \\
&\leq \sum_{e \in E - E_k} c(e)(1.5 \cdot d(e)) + \sum_{e \in E_k} c(e)d(e) \\
&= 1.5 \sum_{e \in E - E_k} c(e)d(e) + \sum_{e \in E_k} c(e)d(e) \\
&= 1.5 \sum_{e \in E - E_k} c(e)d(e) + (1.5 - 0.5) \sum_{e \in E_k} c(e)d(e) \\
&= 1.5 \sum_{e \in E} c(e)d(e) - 0.5 \sum_{e \in E_k} c(e)d(e) \\
&= 1.5 \cdot \text{OPT}_f - 0.5 \cdot W_k \qquad (3)
\end{aligned}
$$

Recall that

$$\sum_{i=1}^{k} W_i = \sum_{i=1}^{k} \sum_{e \in E_i} c(e)d(e) = 2 \sum_{e \in E} c(e)d(e) = 2 \cdot \text{OPT}_f,$$

since every edge is counted twice. This implies that

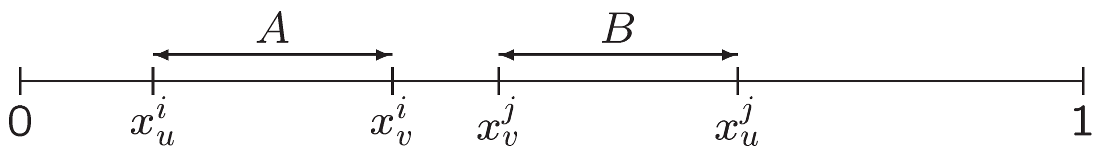
$$W_k \geq (2 \cdot \text{OPT}_f)/k,$$

since W_k is the largest W_i. From (1):

$$
\begin{aligned}
E[c(C)] &\leq 1.5 \cdot \text{OPT}_f - 0.5(2 \cdot \text{OPT}_f/k) \\
&= (1.5 - 1/k) \cdot \text{OPT}_f
\end{aligned}
$$

Derandomization

- Potential problem: ρ is real-valued;
- Thought experiment:
  - Fix σ;
  - Let C_ρ be the solution associated to a given ρ;
  - Let $C_{\rho'}$ be the solution associated to $\rho' = \rho + \epsilon$;
  - When can C_ρ and $C_{\rho'}$ be different?
    - only when there is a vertex u and a terminal t such that $x_u^t \geq \rho$ and $x_u^t < \rho'$.
    - "interesting" things only happen when ρ assumes certain values.



- It is enough to consider:
  - $O(kn')$ values of ρ (where n' is the number of vertices *after* normalization)
  - two values of σ (for each value of ρ)
- There exists a deterministic $(1.5-1/k)$-approximation algorithm for the Multiway Cut Problem.