Today

Approximation Algorithm.

Today

Approximation Algorithm.

Facility Location.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \rightarrow Z$.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \rightarrow Z$.
Find maximum weight perfect matching.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \rightarrow Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$
$$\forall v : \sum_{e=(u,v)} x_e = 1$$
$$x_e \geq 0$$

Maximum Weight Matching.

Bipartite Graph $G=(V,E)$, $w : E \rightarrow Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1$$

$$x_e \geq 0$$

Dual.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$
$$\forall v : \sum_{e=(u,v)} x_e = 1$$
$$x_e \geq 0$$

Dual.
Variable for each constraint.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \rightarrow Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.
Variable for each constraint. p_v

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \rightarrow Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad p_v$$

$$x_e \geq 0$$

Dual.
  Variable for each constraint. p_v unrestricted.
  Constraint for each variable.

Maximum Weight Matching.

Bipartite Graph $G = (V,E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side.

Maximum Weight Matching.

Bipartite Graph $G=(V,E)$, $w:E\rightarrow Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max\sum_e w_e x_e$$

$$\forall v: \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min\sum_v p_v$

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad p_v$$

$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \sum_v p_v$$
$$\forall e = (u, v) : \quad p_u + p_v \geq w_e$$

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$
$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$
$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \textstyle\sum_v p_v$$
$$\forall e = (u,v) : \quad p_u + p_v \geq w_e$$

Weak duality?

Maximum Weight Matching.

Bipartite Graph $G=(V,E)$, $w: E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v: \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.
Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \sum_v p_v$$

$$\forall e=(u,v): \quad p_u + p_v \geq w_e$$

Weak duality? Price function upper bounds matching.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \rightarrow Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.
Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \sum_v p_v$$
$$\forall e = (u, v) : \quad p_u + p_v \geq w_e$$

Weak duality? Price function upper bounds matching.
$\sum_{e \in M} w_e x_e \leq \sum_{e=(u,v) \in M} p_u + p_v \leq \sum_v p_u$.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$
$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$
$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \textstyle\sum_v p_v$$
$$\forall e = (u, v) : \quad p_u + p_v \geq w_e$$

Weak duality? Price function upper bounds matching.
$\sum_{e \in M} w_e x_e \leq \sum_{e=(u,v) \in M} p_u + p_v \leq \sum_v p_u$.

Strong Duality?

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \sum_v p_v$$
$$\forall e = (u,v) : \quad p_u + p_v \geq w_e$$

Weak duality? Price function upper bounds matching.
$\sum_{e \in M} w_e x_e \leq \sum_{e=(u,v) \in M} p_u + p_v \leq \sum_v p_u$.

Strong Duality? Same value solutions.

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.

Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \sum_v p_v$$
$$\forall e = (u, v) : \quad p_u + p_v \geq w_e$$

Weak duality? Price function upper bounds matching.
$\sum_{e \in M} w_e x_e \leq \sum_{e=(u,v) \in M} p_u + p_v \leq \sum_v p_u$.

Strong Duality? Same value solutions. Hungarian algorithm

Maximum Weight Matching.

Bipartite Graph $G = (V, E)$, $w : E \to Z$.
Find maximum weight perfect matching.
Solution: x_e indicates whether edge e is in matching.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dual.
Variable for each constraint. p_v unrestricted.
Constraint for each variable. Edge e, $p_u + p_v \geq w_e$
Objective function from right hand side. $\min \sum_v p_v$

$$\min \sum_v p_v$$

$$\forall e = (u, v) : \quad p_u + p_v \geq w_e$$

Weak duality? Price function upper bounds matching.
$\sum_{e \in M} w_e x_e \leq \sum_{e=(u,v) \in M} p_u + p_v \leq \sum_v p_u$.

Strong Duality? Same value solutions. Hungarian algorithm !!!

Integer Vertex Solution.

Any “vertex” solution is integer!

Integer Vertex Solution.

Any “vertex” solution is integer!

Linear programming feasible region: **Polytope**.

Integer Vertex Solution.

Any “vertex” solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.

Integer Vertex Solution.

Any “vertex” solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.

Integer Vertex Solution.

Any “vertex” solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.
  d “tight” constraints.

Integer Vertex Solution.

Any "vertex" solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.
  d "tight" constraints.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad p_v$$

$$x_e \geq 0$$

Dimension: m

Integer Vertex Solution.

Any "vertex" solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.
d "tight" constraints.

$$\max \sum_e w_e x_e$$

$$\forall v: \sum_{e=(u,v)} x_e = 1 \qquad p_v$$

$$x_e \geq 0$$

Dimension: m
Only $2n$ of the form $\sum_e x_e = 1$.

Integer Vertex Solution.

Any “vertex” solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.
d “tight” constraints.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad p_v$$

$$x_e \geq 0$$

Dimension: m
Only $2n$ of the form $\sum_e x_e = 1$.
Must have $m - 2n$ tight constraints of form $x_e = 0$.

Integer Vertex Solution.

Any "vertex" solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.
d "tight" constraints.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad p_v$$

$$x_e \geq 0$$

Dimension: m
Only $2n$ of the form $\sum_e x_e = 1$.
Must have $m - 2n$ tight constraints of form $x_e = 0$.
Throw away variables that are 0.

Integer Vertex Solution.

Any "vertex" solution is integer!

Linear programming feasible region: **Polytope**.
Dimension of space: number of variables.
Vertex: intersection of d linearly independent constraints.
d "tight" constraints.

$$\max \sum_e w_e x_e$$

$$\forall v : \sum_{e=(u,v)} x_e = 1 \qquad\qquad p_v$$

$$x_e \geq 0$$

Dimension: m
Only $2n$ of the form $\sum_e x_e = 1$.
Must have $m - 2n$ tight constraints of form $x_e = 0$.
Throw away variables that are 0.
Constraint matrix C with $2n$ variables. $2n$ rows.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
  or $\sum_e x_e = 1 \implies x_e = 1$.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
  or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
  or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
  And so on.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
 Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
 or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
 And so on.

Note:

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
  or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
  And so on.

Note:

  also prove the determinant is 1 or -1

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
  or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
  And so on.

Note:

  also prove the determinant is 1 or -1
    for the non-singular matrix.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
   or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
 And so on.

Note:

also prove the determinant is 1 or -1
  for the non-singular matrix.
Plus, Cramer's rule implies integrality.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
    or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
 And so on.

Note:

  also prove the determinant is 1 or -1
    for the non-singular matrix.
Plus, Cramer's rule implies integrality.

..and so on.

Constraint matrix C with $2n$ variables. $2n$ rows.
Each variable in two constraints.
Matrix C has 2 non-zeros in each row and column.
Average degree two bipartite graph.
Even cycle is linearly dependent:
  Negate equations for vertices on one side and add them.
So need another constraint of form $x_e = 0$ for each cycle.
Now, matrix has degree 1 constraint:
  or $\sum_e x_e = 1 \implies x_e = 1$.
This is an integer!!!
  And so on.

Note:

  also prove the determinant is 1 or -1
    for the non-singular matrix.
Plus, Cramer's rule implies integrality.

That's what we did.

Facility location

Set of facilities: F, opening cost f_i for facility i

Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

d_{ij} - distance between i and j.

Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

d_{ij} - distance between i and j.
(notation abuse: clients/facility confusion.)

Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

d_{ij} - distance between i and j.
(notation abuse: clients/facility confusion.)

Triangle inequality: $d_{ij} \leq d_{ik} + d_{kj}$.

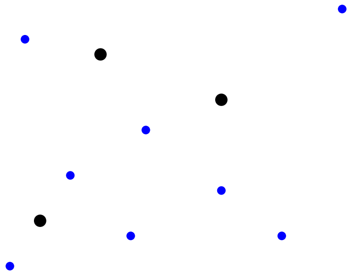
Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

d_{ij} - distance between i and j.
(notation abuse: clients/facility confusion.)

Triangle inequality: $d_{ij} \leq d_{ik} + d_{kj}$.



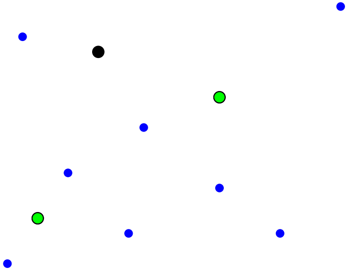
Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

d_{ij} - distance between i and j.
(notation abuse: clients/facility confusion.)

Triangle inequality: $d_{ij} \leq d_{ik} + d_{kj}$.



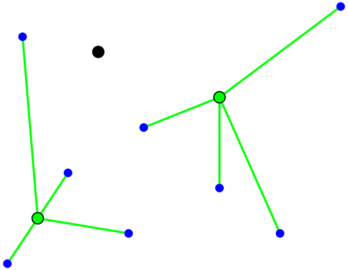
Facility location

Set of facilities: F, opening cost f_i for facility i

Set of clients: D.

d_{ij} - distance between i and j.
(notation abuse: clients/facility confusion.)

Triangle inequality: $d_{ij} \leq d_{ik} + d_{kj}$.



Facility Location

Linear program relaxation:

Facility Location

Linear program relaxation:

“Decision Variables”.

Facility Location

Linear program relaxation:

“Decision Variables”.

y_i - facility i open?

Facility Location

Linear program relaxation:

"Decision Variables".

y_i - facility i open?

x_{ij} - client j assigned to facility i.

Facility Location

Linear program relaxation:

"Decision Variables".

y_i - facility i open?

x_{ij} - client j assigned to facility i.

Facility Location

Linear program relaxation:

"Decision Variables".

y_i - facility i open?

x_{ij} - client j assigned to facility i.

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

Facility Location

Linear program relaxation:

"Decision Variables".

y_i - facility i open?

x_{ij} - client j assigned to facility i.

$$
\begin{aligned}
\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij} & \\
\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 & \\
\forall i \in F, j \in D \quad x_{ij} \leq y_i, & \\
x_{ij}, y_i \geq 0 &
\end{aligned}
$$

Facility opening cost.

Facility Location

Linear program relaxation:

"Decision Variables".
y_i - facility i open?
x_{ij} - client j assigned to facility i.

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

Facility opening cost.
Client Connnection cost.

Facility Location

Linear program relaxation:

"Decision Variables".

y_i - facility i open?

x_{ij} - client j assigned to facility i.

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

Facility opening cost.
Client Connnection cost.
Must connect each client.

Facility Location

Linear program relaxation:

"Decision Variables".

y_i - facility i open?

x_{ij} - client j assigned to facility i.

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

Facility opening cost.
Client Connnection cost.
Must connect each client.
Only connect to open facility.

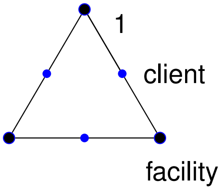
Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$



$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

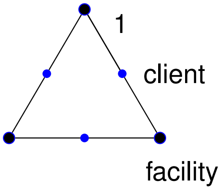
Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$



$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$

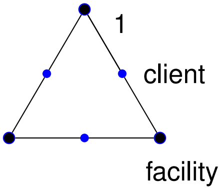
Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

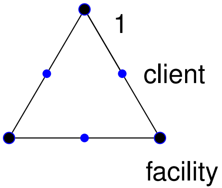


$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$ Connection Cost: 3

Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$



$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$ Connection Cost: 3
Any one Facility:
Facility Cost: 1

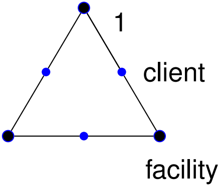
Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$



$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$ Connection Cost: 3
Any one Facility:
Facility Cost: 1 Client Cost: 3.7

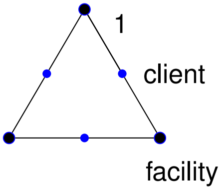
Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

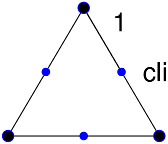


$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$ Connection Cost: 3
Any one Facility:
Facility Cost: 1 Client Cost: 3.7
Make it worse?

Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$



$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$ Connection Cost: 3
Any one Facility:
Facility Cost: 1 Client Cost: 3.7
Make it worse? Sure.

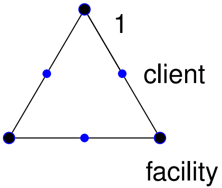
Integer Solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$



$x_{ij} = \frac{1}{2}$ edges.
$y_i = \frac{1}{2}$ edges.

Facility Cost: $\frac{3}{2}$ Connection Cost: 3
Any one Facility:
Facility Cost: 1 Client Cost: 3.7
Make it worse? Sure. Not as pretty!

Round solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

Round solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

Round independently?

Round solution?

$$
\begin{aligned}
\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij} \\
\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 \\
\forall i \in F, j \in D \quad x_{ij} \leq y_i, \\
x_{ij}, y_i \geq 0
\end{aligned}
$$

Round independently?

y_i and x_{ij} separately?

Round solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

Round independently?

y_i and x_{ij} separately? Assign to closed facility!

Round solution?

$$
\begin{aligned}
\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij} \\
\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 \\
\forall i \in F, j \in D \quad x_{ij} \leq y_i, \\
x_{ij}, y_i \geq 0
\end{aligned}
$$

Round independently?

y_i and x_{ij} separately? Assign to closed facility!

Round x_{ij} and open facilities?

Round solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

Round independently?

y_i and x_{ij} separately? Assign to closed facility!

Round x_{ij} and open facilities?
Different clients force different facilities open.

Round solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

Round independently?

y_i and x_{ij} separately? Assign to closed facility!

Round x_{ij} and open facilities?
Different clients force different facilities open.

Any ideas?

Round solution?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

Round independently?

y_i and x_{ij} separately? Assign to closed facility!

Round x_{ij} and open facilities?
Different clients force different facilities open.

Any ideas?

Use Dual!

The dual.

$\min cx, Ax \geq b$

The dual.

$\min cx, Ax \geq b \leftrightarrow$

The dual.

$\min cx, Ax \geq b \leftrightarrow \max bx, y^T A \leq c.$

The dual.

$\min cx, Ax \geq b \leftrightarrow \max bx, y^T A \leq c.$

$$\begin{aligned} \min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij} \\ \forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 \\ \forall i \in F, j \in D \quad x_{ij} \leq y_i, \end{aligned}$$

The dual.

$\min cx, Ax \geq b \leftrightarrow \max bx, y^T A \leq c.$

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 \qquad ; \ \alpha_j$$
$$\forall i \in F, j \in D \quad y_i - x_{ij} \geq 0 \qquad ; \ \beta_{ij}$$
$$x_{ij}, y_i \geq 0$$

The dual.

$\min cx, Ax \geq b \leftrightarrow \max bx, y^T A \leq c.$

$$\begin{aligned} \min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij} \\ \forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 \\ \forall i \in F, j \in D \quad x_{ij} \leq y_i, \end{aligned}$$

$$\begin{aligned} \min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij} \\ \forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1 \quad &; \ \alpha_j \\ \forall i \in F, j \in D \quad y_i - x_{ij} \geq 0 \quad &; \ \beta_{ij} \\ x_{ij}, y_i \geq 0 \end{aligned}$$

$$\begin{aligned} \max \sum_j \alpha_j \\ \forall i \quad \sum_{j \in D} \beta_{ij} \leq f_i \quad &; \ y_i \\ \forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \quad &; \ x_{ij} \\ \beta_{ij}, \alpha_j \geq 0 \end{aligned}$$

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$

$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$

$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$

$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$

$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$

$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$

$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
maximize price paid by client to connect!

Interpretation of Dual?

$$\min \sum_{i\in F} y_i f_i + \sum_{i\in F, j\in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i\in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j\in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_{j} \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \quad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.

maximize price paid by client to connect!

Objective: $\sum_j \alpha_j$ total payment.

Client j travels or pays to open facility i.

Costs client d_{ij} to get to there.

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

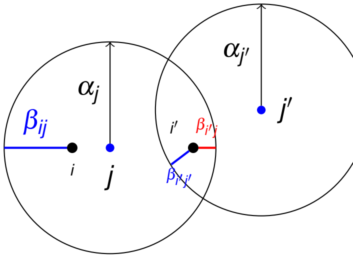
α_j charge to client.
maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.
Costs client d_{ij} to get to there.
Savings is $\alpha_j - d_{ij}$.

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \quad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
 maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.
 Costs client d_{ij} to get to there.
 Savings is $\alpha_j - d_{ij}$.
 Willing to pay $\beta_{ij} = \alpha_j - d_{ij}$.

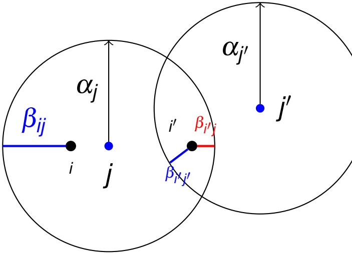


Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_{j} \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
 maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.
 Costs client d_{ij} to get to there.
 Savings is $\alpha_j - d_{ij}$.
 Willing to pay $\beta_{ij} = \alpha_j - d_{ij}$.



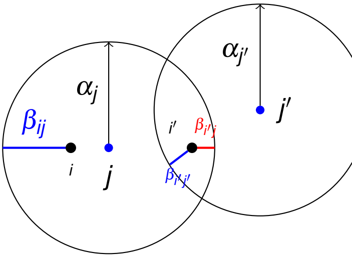
Total payment to facility i at most f_i before opening.

Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \qquad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.
Costs client d_{ij} to get to there.
Savings is $\alpha_j - d_{ij}$.
Willing to pay $\beta_{ij} = \alpha_j - d_{ij}$.
Total payment to facility i at most f_i before opening.
Complementary slackness:

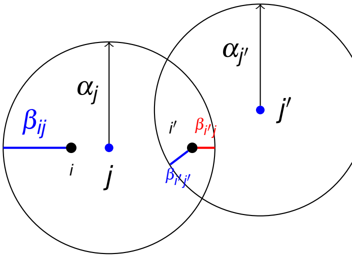


Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \quad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.
Costs client d_{ij} to get to there.
Savings is $\alpha_j - d_{ij}$.
Willing to pay $\beta_{ij} = \alpha_j - d_{ij}$.
Total payment to facility i at most f_i before opening.
Complementary slackness: $x_{ij} \geq 0$ if and only if $\alpha_j \geq d_{ij}$.

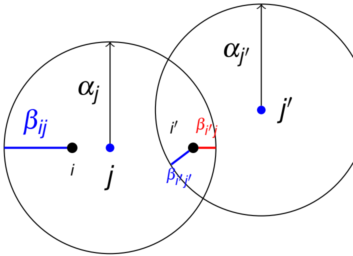


Interpretation of Dual?

$$\min \sum_{i \in F} y_i f_i + \sum_{i \in F, j \in D} x_{ij} d_{ij}$$
$$\forall j \in D \quad \sum_{i \in F} x_{ij} \geq 1$$
$$\forall i \in F, j \in D \quad x_{ij} \leq y_i,$$
$$x_{ij}, y_i \geq 0$$

$$\max \sum_j \alpha_j$$
$$\forall i \in F \quad \sum_{j \in D} \beta_{ij} \leq f_i$$
$$\forall i \in F, j \in D \quad \alpha_j - \beta_{ij} \leq d_{ij} \quad x_{ij}$$
$$\alpha_j, \beta_{ij} \geq 0$$

α_j charge to client.
maximize price paid by client to connect!
Objective: $\sum_j \alpha_j$ total payment.
Client j travels or pays to open facility i.
Costs client d_{ij} to get to there.
Savings is $\alpha_j - d_{ij}$.
Willing to pay $\beta_{ij} = \alpha_j - d_{ij}$.
Total payment to facility i at most f_i before opening.
Complementary slackness: $x_{ij} \geq 0$ if and only if $\alpha_j \geq d_{ij}$.
only assign client to "paid to" facilities.



Use Dual.

1. Find solution to primal, (x, y). and dual, (α, β).

Use Dual.

1. Find solution to primal, (x, y). and dual, (α, β).
2. For smallest (remaining) α_j,

Use Dual.

1. Find solution to primal, (x, y). and dual, (α, β).
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.

Use Dual.

1. Find solution to primal, (x, y). and dual, (α, β).
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.

Use Dual.

1. Find solution to primal, (x,y). and dual, (α,β).
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Use Dual.

1. Find solution to primal, (x, y). and dual, (α, β).
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Use Dual.

1. Find solution to primal, (x,y). and dual, (α,β).
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.
3. Removed assigned clients, goto 2.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

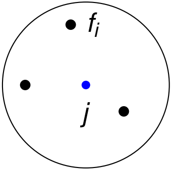
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
       Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Proof: Step 2 picks client j.

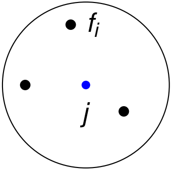


Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

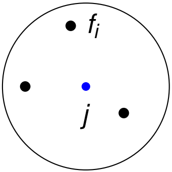
Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j



Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
      Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



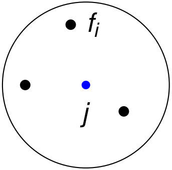
Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

$f_{\min}$

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



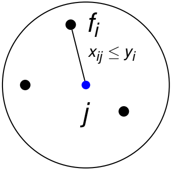
Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij}$

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



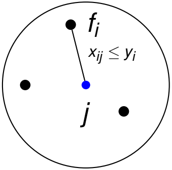
Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i$

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



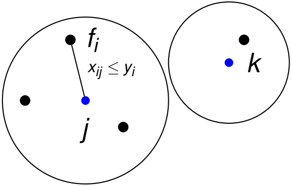
Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i \leq \sum_{i \in N_j} y_i f_i$.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

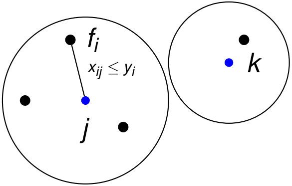
$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i \leq \sum_{i \in N_j} y_i f_i$.

For k used in Step 2.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

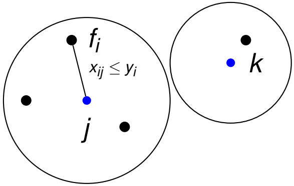
$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i \leq \sum_{i \in N_j} y_i f_i$.

For k used in Step 2.
$N_j \cap N_k = \emptyset$ for j and k in step 2.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

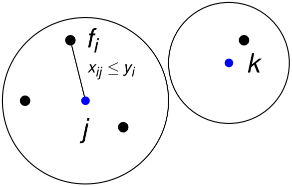
$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i \leq \sum_{i \in N_j} y_i f_i$.

For k used in Step 2.
$N_j \cap N_k = \emptyset$ for j and k in step 2.
$\rightarrow$ Any facility in ≤ 1 sum from step 2.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

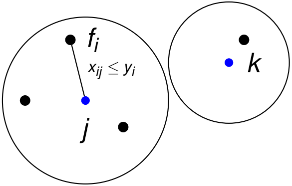
$$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i \leq \sum_{i \in N_j} y_i f_i.$$

For k used in Step 2.
$N_j \cap N_k = \emptyset$ for j and k in step 2.
$\rightarrow$ Any facility in ≤ 1 sum from step 2.
$\rightarrow$ total step 2 facility cost is $\leq \sum_i y_i f_i$.

Integral facility cost at most LP facility cost.

Claim: Total facility cost is at most $\sum_i f_i y_i$.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.



Proof: Step 2 picks client j.
$f_{\min}$ - min cost facility in N_j

$f_{\min} \leq f_{\min} \cdot \sum_{i \in N_j} x_{ij} \leq f_{\min} \sum_{i \in N_j} y_i \leq \sum_{i \in N_j} y_i f_i$.

For k used in Step 2.
$N_j \cap N_k = \emptyset$ for j and k in step 2.
$\rightarrow$ Any facility in ≤ 1 sum from step 2.
$\rightarrow$ total step 2 facility cost is $\leq \sum_i y_i f_i$.

Connection Cost.

2. For smallest (remaining) α_j,

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
      Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Connection Cost.

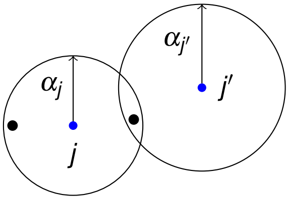
2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

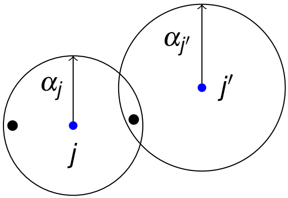


Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
       Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

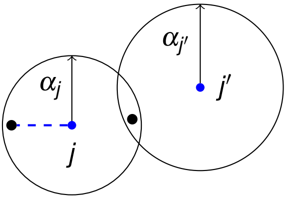
Connection Cost of j:



Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.



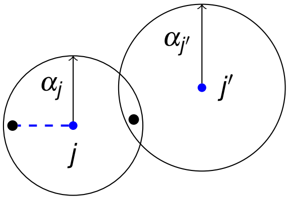
Connection Cost of j:

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

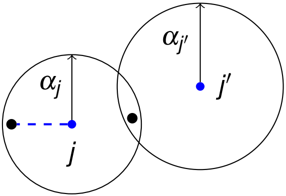
Connection Cost of j: $\leq \alpha_j$.



Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

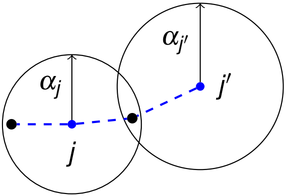


Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

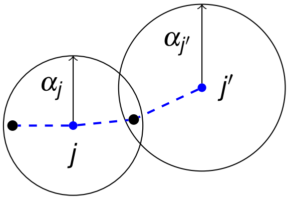


Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

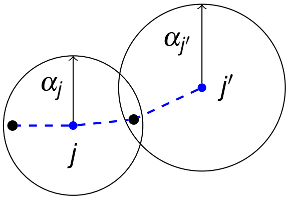


Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
$\leq \alpha_{j'} + \alpha_j + \alpha_j$

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

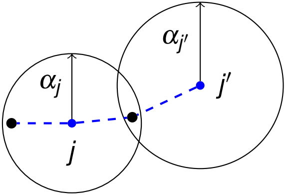


Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
$\leq \alpha_{j'} + \alpha_j + \alpha_j \leq 3\alpha_{j'}$.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.

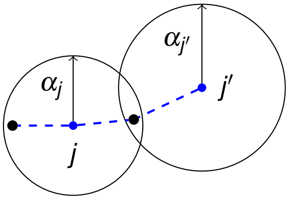


Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
$\leq \alpha_{j'} + \alpha_j + \alpha_j \leq 3\alpha_{j'}$.
since $\alpha_j \leq \alpha_{j'}$

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
      Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.



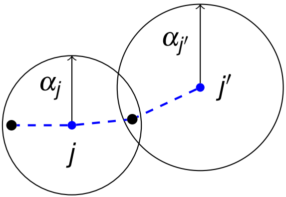
Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
  $\leq \alpha_{j'} + \alpha_j + \alpha_j \leq 3\alpha_{j'}$.
  since $\alpha_j \leq \alpha_{j'}$

Total connection cost:
  at most $3\sum_j \alpha_j \leq 3$ times Dual OPT.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.



Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
$\leq \alpha_{j'} + \alpha_j + \alpha_j \leq 3\alpha_{j'}$.
since $\alpha_j \leq \alpha_{j'}$

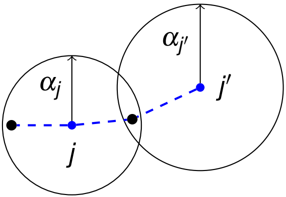
Total connection cost:
at most $3\sum_j \alpha_j \leq 3$ times Dual OPT.

Previous Slide: Facility cost:

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.



Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
$\leq \alpha_{j'} + \alpha_j + \alpha_j \leq 3\alpha_{j'}$.
since $\alpha_j \leq \alpha_{j'}$

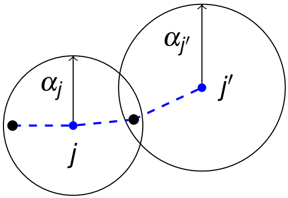
Total connection cost:
at most $3\sum_j \alpha_j \leq$ 3 times Dual OPT.

Previous Slide: Facility cost:
$\leq$ primal "facility" cost $\leq$ Primal OPT.

Connection Cost.

2. For smallest (remaining) α_j,
   (a) Let $N_j = \{i : x_{ij} > 0\}$.
   (b) Open cheapest facility i in N_j.
   Every client j' with $N_{j'} \cap N_j \neq \emptyset$ assigned to i.

Client j is directly connected. Clients j' are indirectly connected.



Connection Cost of j: $\leq \alpha_j$.
Connection Cost of j':
$\leq \alpha_{j'} + \alpha_j + \alpha_j \leq 3\alpha_{j'}$.
since $\alpha_j \leq \alpha_{j'}$

Total connection cost:
at most $3\sum_j \alpha_j \leq$ 3 times Dual OPT.

Previous Slide: Facility cost:
$\leq$ primal "facility" cost $\leq$ Primal OPT.

Total Cost: 4 OPT.

Twist on randomized rounding.

Client j:

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$,

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution!

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i$

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
 Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
 $\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
 and separate balls implies total $\leq \sum_i y_i f_i$.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j'

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

In step 2: pick in increasing order of $\alpha_j + D_j$.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

In step 2: pick in increasing order of $\alpha_j + D_j$.

$\rightarrow$ Expected cost is $\leq (2\alpha_{j'} + D_{j'})$.
Connection cost: $2\sum_j \alpha_j + \sum_j D_j$.
$2OPT(D)$ plus connection cost of primal.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

In step 2: pick in increasing order of $\alpha_j + D_j$.

$\rightarrow$ Expected cost is $\leq (2\alpha_{j'} + D_{j'})$.
Connection cost: $2\sum_j \alpha_j + \sum_j D_j$.
$2OPT(D)$ plus connection cost of primal.

Total expected cost:

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

In step 2: pick in increasing order of $\alpha_j + D_j$.

$\rightarrow$ Expected cost is $\leq (2\alpha_{j'} + D_{j'})$.
Connection cost: $2\sum_j \alpha_j + \sum_j D_j$.
$2OPT(D)$ plus connection cost of primal.

Total expected cost:
Facility cost is at most facility cost of primal.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

In step 2: pick in increasing order of $\alpha_j + D_j$.

$\rightarrow$ Expected cost is $\leq (2\alpha_{j'} + D_{j'})$.
Connection cost: $2\sum_j \alpha_j + \sum_j D_j$.
$2OPT(D)$ plus connection cost of primal.

Total expected cost:
Facility cost is at most facility cost of primal.
Connection cost at most $2OPT$ + connection cost of prmal.

Twist on randomized rounding.

Client j: $\sum_i x_{ij} = 1$, $x_{ij} \geq 0$.
Probability distribution! $\rightarrow$ Choose from distribution, x_{ij}, in step 2.

Expected opening cost:
$\sum_{i \in N_j} x_{ij} f_i \leq \sum_{i \in N_j} y_i f_i$.
and separate balls implies total $\leq \sum_i y_i f_i$.

$D_j = \sum_i x_{ij} d_{ij}$ Expected connection cost of primal for j.

Expected connection cost j' $\alpha_j + \alpha_{j'} + D_j$.

In step 2: pick in increasing order of $\alpha_j + D_j$.

$\rightarrow$ Expected cost is $\leq (2\alpha_{j'} + D_{j'})$.
Connection cost: $2\sum_j \alpha_j + \sum_j D_j$.
$2OPT(D)$ plus connection cost of primal.

Total expected cost:
Facility cost is at most facility cost of primal.
Connection cost at most $2OPT$ + connection cost of prmal.
$\rightarrow$ at most $3OPT$.

Primal dual algorithm.

1. Feasible integer solution.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Typically. (If dual is maximization.)

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Typically. (If dual is maximization.)
Begin with feasible dual.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Typically. (If dual is maximization.)
Begin with feasible dual.
Raise dual variables until tight constraint.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Typically. (If dual is maximization.)
Begin with feasible dual.
Raise dual variables until tight constraint.
Set corresponding primal variable to an integer.

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Typically. (If dual is maximization.)
  Begin with feasible dual.
  Raise dual variables until tight constraint.
  Set corresponding primal variable to an integer.

Recall Dual:

Primal dual algorithm.

1. Feasible integer solution.
2. Feasible dual solution.
3. Cost of integer solution $\leq \alpha$ times dual value.

Just did it. Used linear program. Faster?

Typically. (If dual is maximization.)
  Begin with feasible dual.
  Raise dual variables until tight constraint.
  Set corresponding primal variable to an integer.

Recall Dual:

$$
\begin{aligned}
& \max \sum_j \alpha_j \\
\forall i \in F \quad & \sum_{j \in D} \beta_{ij} \leq f_i \\
\forall i \in F, j \in D \quad & \alpha_j - \beta_{ij} \leq d_{ij} \\
& \alpha_j, \beta_{ij} \leq 0
\end{aligned}
$$

Facility location primal dual.

Phase 1:

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.
2. Raise α_j for every (unconnected) client.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.
2. Raise α_j for every (unconnected) client.
When $\alpha_j = d_{ij}$ for some i

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.
2. Raise α_j for every (unconnected) client.
When $\alpha_j = d_{ij}$ for some i
raise β_{ij} at same rate

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why?

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why?

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight *ji* clients *j* to *i*.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight *ji* clients *j* to *i*.

3. Continue until all clients connected.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:

Make “edge” between two facilities if paid by a common client.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:

Make "edge" between two facilities if paid by a common client.

Permanently open an independent set of facilities in common client graph.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:

Make "edge" between two facilities if paid by a common client.
Permanently open an independent set of facilities in common client graph.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:

Make "edge" between two facilities if paid by a common client.
Permanently open an independent set of facilities in common client graph.

For client j, connected facility i is opened.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.

2. Raise α_j for every (unconnected) client.

When $\alpha_j = d_{ij}$ for some i

raise β_{ij} at same rate Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Intution:Paying β_{ij} to open i.

Stop when $\sum_i \beta_{ij} = f_i$.

Why? Dual: $\sum_i \beta_{ij} \leq f_i$

Intution: facility paid for.

Temporarily open i.

Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:

Make "edge" between two facilities if paid by a common client.
Permanently open an independent set of facilities in common client graph.

For client j, connected facility i is opened. Good.

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.
2. Raise α_j for every (unconnected) client.
When $\alpha_j = d_{ij}$ for some i
raise β_{ij} at same rate Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.
Intution:Paying β_{ij} to open i.
Stop when $\sum_i \beta_{ij} = f_i$.
Why? Dual: $\sum_i \beta_{ij} \leq f_i$
Intution: facility paid for.
Temporarily open i.
Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:
Make "edge" between two facilities if paid by a common client.
Permanently open an independent set of facilities in common client graph.

For client j, connected facility i is opened. Good.
Connected facility not open

Facility location primal dual.

Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.
2. Raise α_j for every (unconnected) client.
When $\alpha_j = d_{ij}$ for some i
raise β_{ij} at same rate   Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.
Intution:Paying β_{ij} to open i.
Stop when $\sum_i \beta_{ij} = f_i$.
Why? Dual: $\sum_i \beta_{ij} \leq f_i$
Intution: facility paid for.
Temporarily open i.
Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:
Make "edge" between two facilities if paid by a common client.
Permanently open an independent set of facilities in common client graph.

For client j, connected facility i is opened. Good.
Connected facility not open
$\rightarrow$ exists client j' paid i and connected to open facility.

Facility location primal dual.

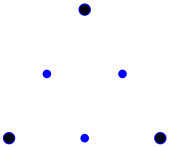
Phase 1: 1. Initially $\alpha_j, \beta_{ij} = 0$.
2. Raise α_j for every (unconnected) client.
When $\alpha_j = d_{ij}$ for some i
raise β_{ij} at same rate Why? Dual: $\alpha_j - \beta_{ij} \leq d_{ij}$.
Intution:Paying β_{ij} to open i.
Stop when $\sum_i \beta_{ij} = f_i$.
Why? Dual: $\sum_i \beta_{ij} \leq f_i$
Intution: facility paid for.
Temporarily open i.
Connect all tight ji clients j to i.

3. Continue until all clients connected.

Phase 2:
Make "edge" between two facilities if paid by a common client.
Permanently open an independent set of facilities in common client graph.

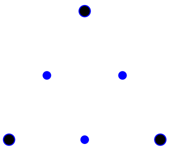
For client j, connected facility i is opened. Good.
Connected facility not open
$\rightarrow$ exists client j' paid i and connected to open facility.
Connect j to j''s open facility.

Constraints for dual.



Constraints for dual.

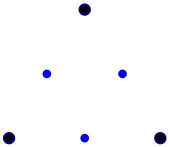
$\sum_j \beta_{ij} \leq f_i$



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

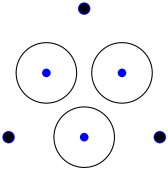


Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

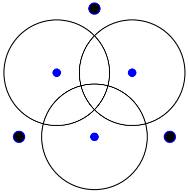


Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.



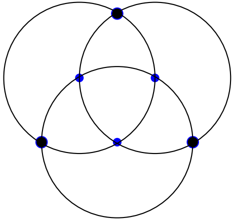
Constraints for dual.

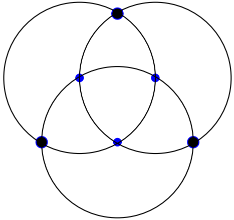
$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!





Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint:

Constraints for dual.

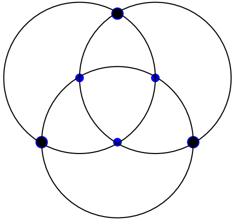
$\sum_j \beta_{ij} \leq f_i$

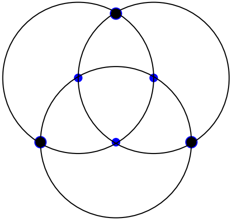
$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.





Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

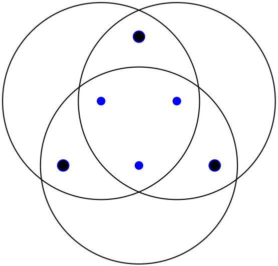
$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

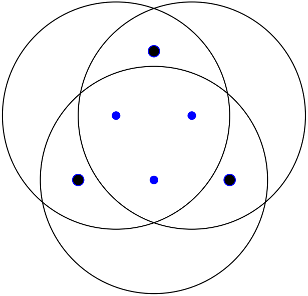
$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

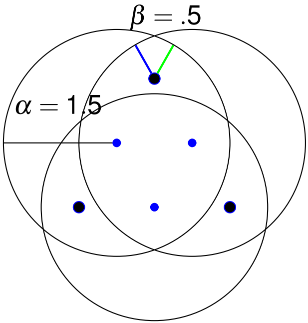
$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

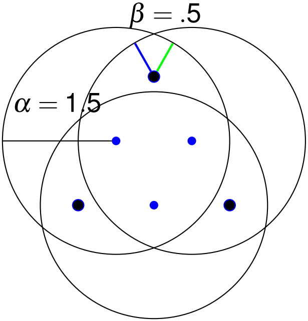
Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

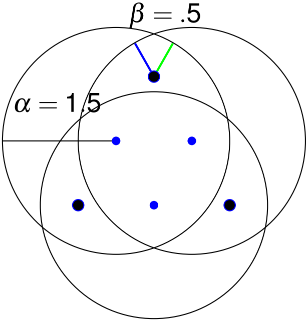
$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

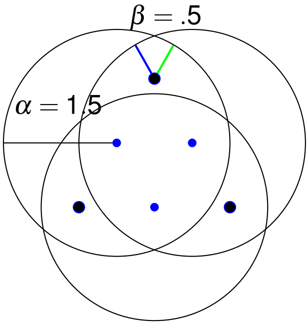
$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

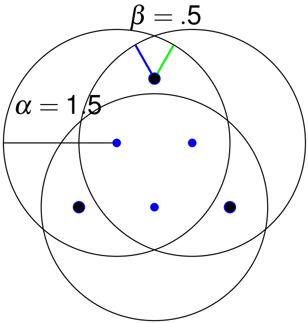
Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j$



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

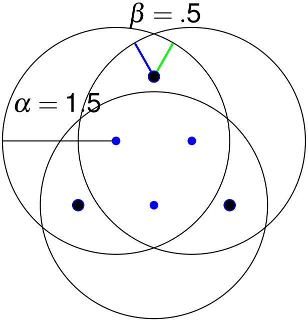
Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

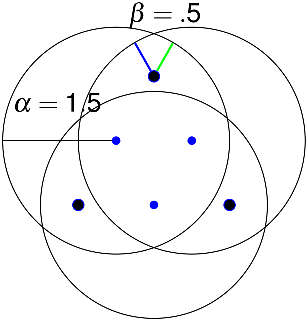
Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

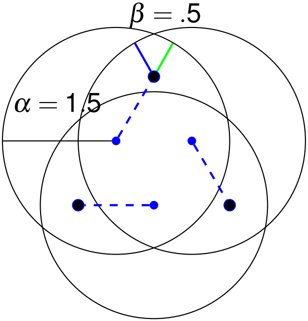
Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

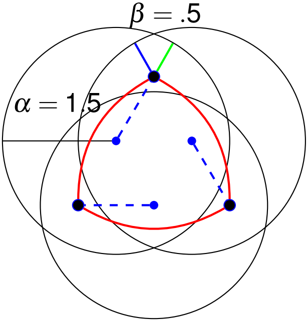
$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.

Assign Clients to “paid to” open facility.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

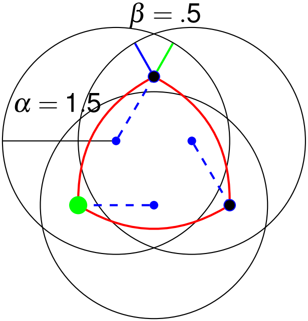
Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.

Assign Clients to “paid to” open facility.

Connect facilities with common client.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

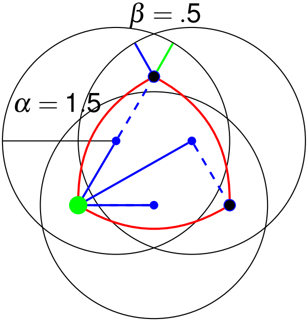
LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.

Assign Clients to “paid to” open facility.

Connect facilities with common client.

Open independent set.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

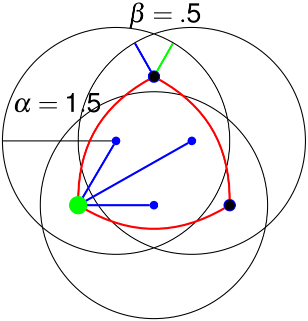
Temporarily open all facilities.

Assign Clients to “paid to” open facility.

Connect facilities with common client.

Open independent set.

Connect to “killer” client’s facility.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.

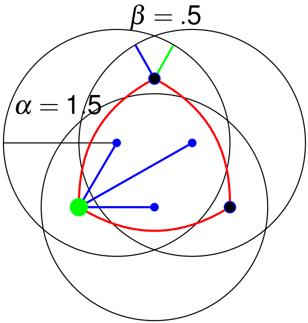
Assign Clients to “paid to” open facility.

Connect facilities with common client.

Open independent set.

Connect to “killer” client’s facility.

Cost: 1 + 3.7



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.

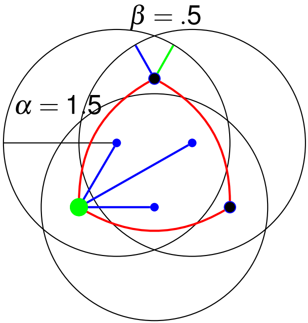
Assign Clients to “paid to” open facility.

Connect facilities with common client.

Open independent set.

Connect to “killer” client’s facility.

Cost: 1 + 3.7 = 4.7.



Constraints for dual.

$\sum_j \beta_{ij} \leq f_i$

$\alpha_i - \beta_{ij} \leq d_{ij}$.

Grow α_j.

$\alpha_j = d_{ij}$!

Tight constraint: $\alpha_j - \beta_{ij} \leq d_{ij}$.

Grow β_{ij} (and α_j).

$\sum_j \beta_{ij} = f_i$ for all facilities.

Tight: $\sum_j \beta_{ij} \leq f_i$

LP Cost: $\sum_j \alpha_j = 4.5$

Temporarily open all facilities.

Assign Clients to “paid to” open facility.

Connect facilities with common client.

Open independent set.

Connect to “killer” client’s facility.

Cost: 1 + 3.7 = 4.7.

A bit more than the LP cost.

Analysis

Claim: Client only pays one facility.

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Claim: S_i - directly connected clients to open facility i.

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.

Proof:

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.

Proof:
$f_i = \sum_{j \in S_i} \beta_{ij}$

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.

Proof:
$f_i = \sum_{j \in S_i} \beta_{ij} = \sum_{j \in S_i} \alpha_j - d_{ij}$.

Analysis

Claim: Client only pays one facility.

Independent set of facilities.

Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.

Proof:
$f_i = \sum_{j \in S_i} \beta_{ij} = \sum_{j \in S_i} \alpha_j - d_{ij}$.
Since directly connected: $\beta_{ij} = \alpha_j - d_{ij}$. $\square$

Analysis.

Claim: Client j is indirectly connected to i

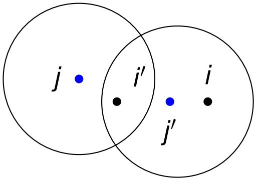
Analysis.

Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Analysis.

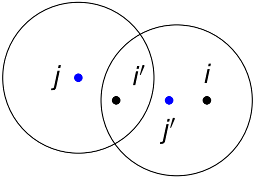
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Directly connected to (temp open) i'



Analysis.

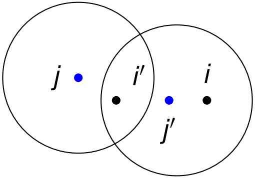
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.

Analysis.

Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

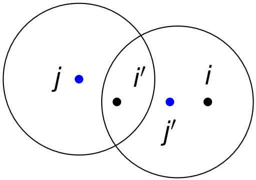


Directly connected to (temp open) i' conflicts with i.

exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.

Analysis.

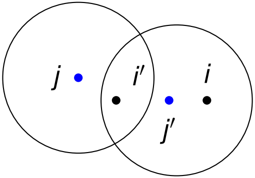
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α_j'.

Analysis.

Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.

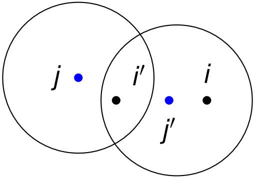
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.

When i' opens, stops both α_j and α'_j.

$\alpha_{j'}$ stopped no later

Analysis.

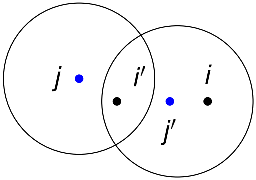
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α'_j.
$\alpha_{j'}$ stopped no later (..maybe earlier..)

Analysis.

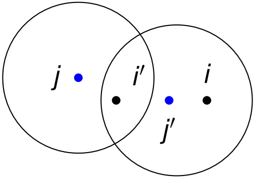
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α'_j.
$\alpha_{j'}$ stopped no later (..maybe earlier..)
$\alpha_{j'} \leq \alpha_j$.

Analysis.

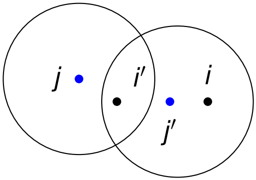
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α'_j.
$\alpha_{j'}$ stopped no later (..maybe earlier..)
$\alpha_{j'} \leq \alpha_j$.
Total distance from j to j'.

Analysis.

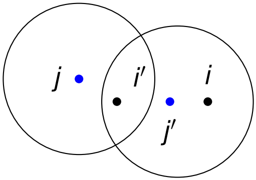
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α_j'.
$\alpha_{j'}$ stopped no later (..maybe earlier..)
$\alpha_{j'} \leq \alpha_j$.
Total distance from j to j'.
$d_{ji'} +$

Analysis.

Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.

exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.

When i' opens, stops both α_j and α_j'.

$\alpha_{j'}$ stopped no later (..maybe earlier..)

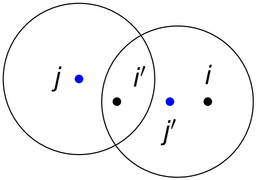
$\alpha_{j'} \leq \alpha_j$.

Total distance from j to j'.

$d_{ji'} + d_{i'j'} +$

Analysis.

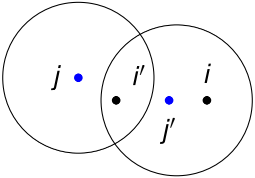
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α_j'.
$\alpha_{j'}$ stopped no later (..maybe earlier..)
$\alpha_{j'} \leq \alpha_j$.
Total distance from j to j'.
$d_{ji'} + d_{i'j'} + d_{j'i}$

Analysis.

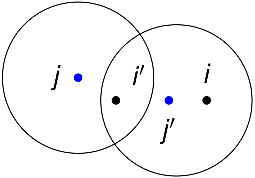
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.
exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.
When i' opens, stops both α_j and α'_j.
$\alpha_{j'}$ stopped no later (..maybe earlier..)
$\alpha_{j'} \leq \alpha_j$.
Total distance from j to j'.
$d_{ji'} + d_{i'j'} + d_{j'i} \leq 3\alpha_j$

Analysis.

Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.



Directly connected to (temp open) i' conflicts with i.

exists j' with $\alpha_{j'} \geq d_{ij'}$ and $\alpha_j \geq d_{i'j'}$.

When i' opens, stops both α_j and α_j'.

$\alpha_{j'}$ stopped no later (..maybe earlier..)

$\alpha_{j'} \leq \alpha_j$.

Total distance from j to j'.

$d_{ji'} + d_{i'j'} + d_{j'i} \leq 3\alpha_j$ □

Putting it together!

Claim: Client only pays one facility.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.
feasible dual upper bounds fractional (and integer) primal.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.
feasible dual upper bounds fractional (and integer) primal.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.
feasible dual upper bounds fractional (and integer) primal.

3 OPT.

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.
feasible dual upper bounds fractional (and integer) primal.

3 OPT.

Fast!

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.
feasible dual upper bounds fractional (and integer) primal.

3 OPT.

Fast! Cheap!

Putting it together!

Claim: Client only pays one facility.
Claim: S_i - directly connected clients to open facility i.
$f_i + \sum_{j \in S_i} d_{ij} \leq \sum_j \alpha_j$.
Claim: Client j is indirectly connected to i
$\rightarrow d_{ij} \leq 3\alpha_j$.

Total Cost:
direct clients dual (α_j) pays for facility and own connections.
plus no more than 3 times indirect client dual.
Total Cost: 3 times dual.
feasible dual upper bounds fractional (and integer) primal.

3 OPT.

Fast! Cheap! Safe!

Check: if time.

Check: if time.

Won’t see you on Tuesday.

Check: if time.

Won’t see you on Tuesday.

Guest Speaker: Tselil Schramm.

Check: if time.

Won't see you on Tuesday.

Guest Speaker: Tselil Schramm.

Semidefinite Programming and Approximation.