O projeto de algoritmos online competitivos

Mário César San Felice

Instituto de Matemática e Estatística - Universidade de São Paulo

felice@ic.unicamp.br

23 de junho de 2015

Summary

Summary

Combinatorial Optimization Problems:

Summary

Combinatorial Optimization Problems:

- Steiner Tree, Facility Location, Connected Facility Location.

Summary

Combinatorial Optimization Problems:

- Steiner Tree, Facility Location, Connected Facility Location.

Online Computation and Competitive Analysis:

Summary

Combinatorial Optimization Problems:

- Steiner Tree, Facility Location, Connected Facility Location.

Online Computation and Competitive Analysis:

- Steiner family problems,

Summary

Combinatorial Optimization Problems:

- Steiner Tree, Facility Location, Connected Facility Location.

Online Computation and Competitive Analysis:

- Steiner family problems,
- Facility Location family problems,

Summary

Combinatorial Optimization Problems:

- Steiner Tree, Facility Location, Connected Facility Location.

Online Computation and Competitive Analysis:

- Steiner family problems,
- Facility Location family problems,
- Online Connected Facility Location problem.

Summary

Combinatorial Optimization Problems:

- Steiner Tree, Facility Location, Connected Facility Location.

Online Computation and Competitive Analysis:

- Steiner family problems,
- Facility Location family problems,
- Online Connected Facility Location problem.

Competitive Analysis of the Online Single-Source Rent-or-Buy algorithm.

Combinatorial Optimization Problems

Combinatorial Optimization Problems

Problems with an objective function to be minimized or maximized.

Combinatorial Optimization Problems

Problems with an objective function to be minimized or maximized.

Minimization problems in which we are interested:

- Steiner Tree problem,
- Facility Location problem,
- Connected Facility Location problem.

Combinatorial Optimization Problems

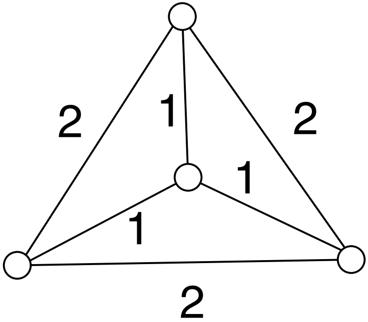
Problems with an objective function to be minimized or maximized.

Minimization problems in which we are interested:

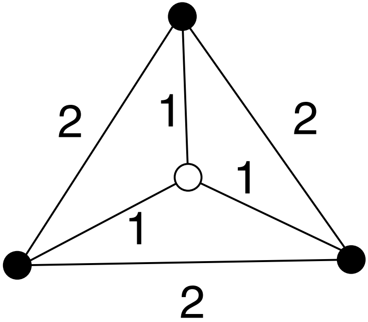
- Steiner Tree problem,
- Facility Location problem,
- Connected Facility Location problem.

These problems are NP-hard and constant factor approximation algorithms are known for them.

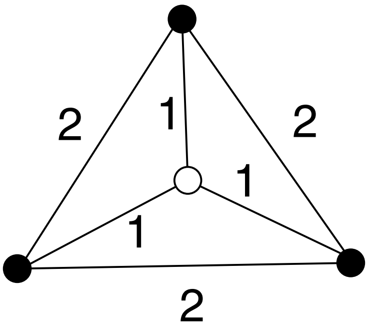
Steiner Tree Problem



Steiner Tree Problem

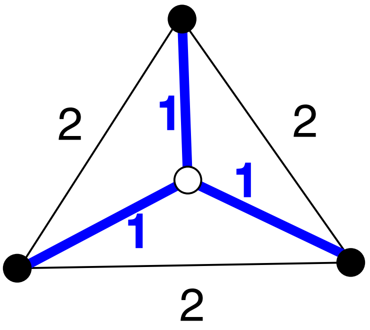


Steiner Tree Problem



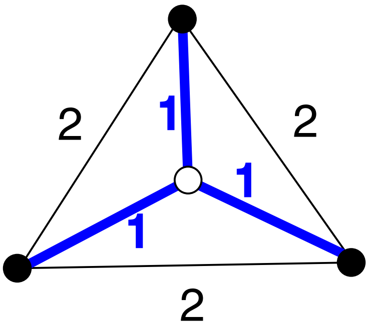
$$\min \sum_{e \in E(T)} d(e)$$

Steiner Tree Problem



$$\min \sum_{e \in E(T)} d(e)$$

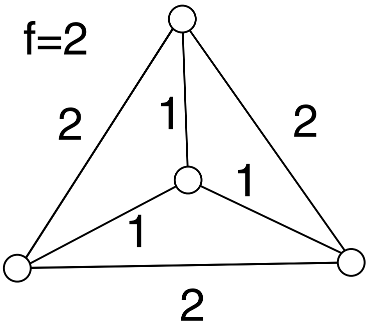
Steiner Tree Problem



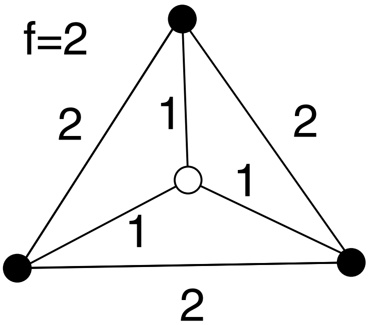
$$\min \sum_{e \in E(T)} d(e)$$

Total cost $= 3$.

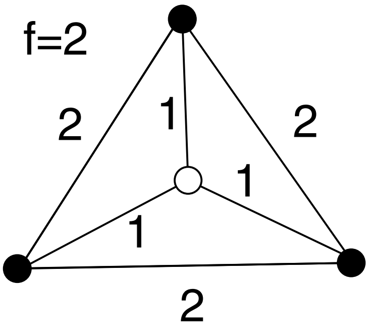
Facility Location Problem



Facility Location Problem

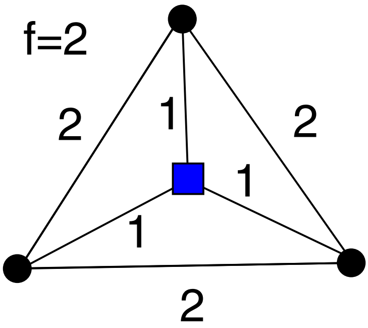


Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a)$$

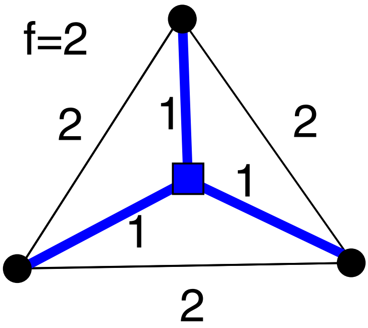
Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a)$$

Total cost $= 2$

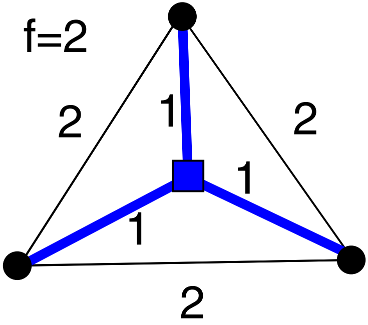
Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a)$$

Total cost $= 2 + 3$

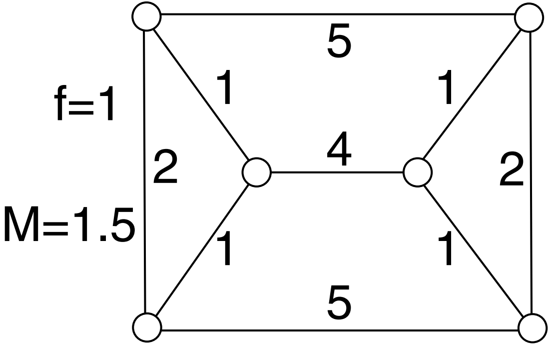
Facility Location Problem



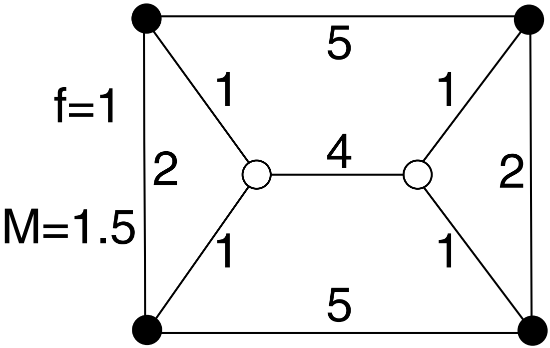
$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a)$$

Total cost $= 2 + 3 = 5.$

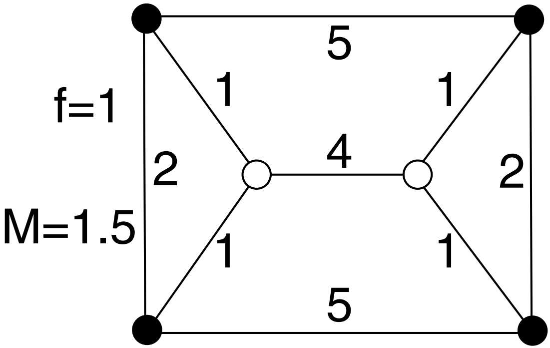
Connected Facility Location Problem



Connected Facility Location Problem

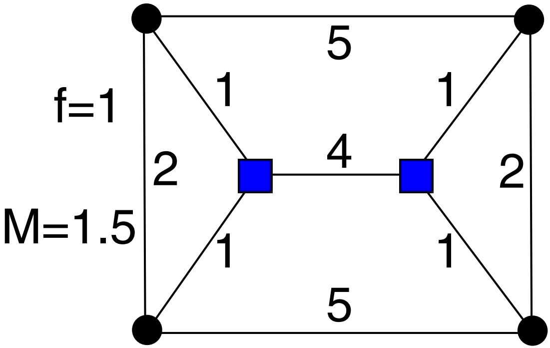


Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a) + M \sum_{e \in E(T)} d(e)$$

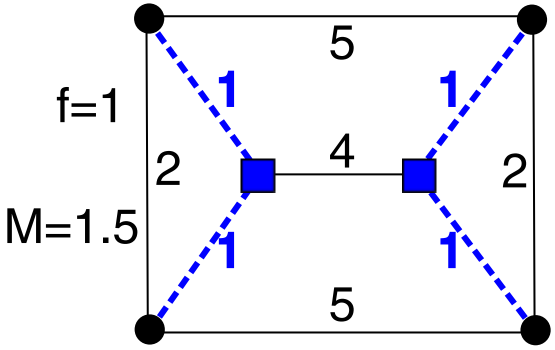
Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 2$

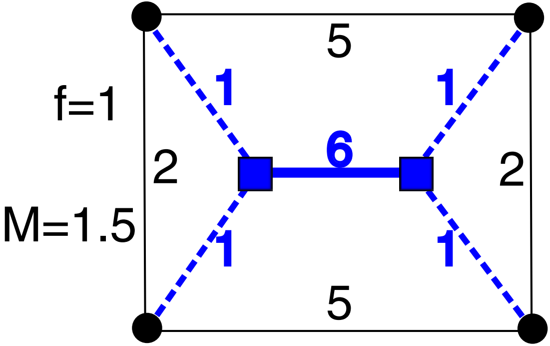
Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 2 + 4$

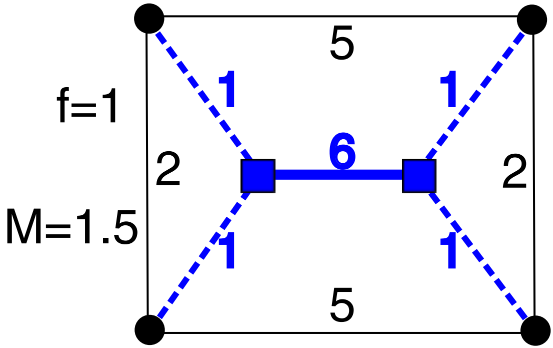
Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 2 + 4 + 6$

Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, F^a) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 2 + 4 + 6 = 12.$

Online Computation

Online Computation

Parts of the input are revealed one at a time.

Online Computation

Parts of the input are revealed one at a time.

Each part must be served before the next one arrives.

Online Computation

Parts of the input are revealed one at a time.

Each part must be served before the next one arrives.

No decision can be changed in the future.

Competitive Analysis

Competitive Analysis

Worst case technique used to analyze online algorithms.

Competitive Analysis

Worst case technique used to analyze online algorithms.

An online algorithm ALG is c-competitive if:

$$\mathrm{ALG}(I) \leq c \cdot \mathrm{OPT}(I) + \kappa,$$

for every input I and some constant κ.

Online Problems

Minimization problems in which we are interested:

Online Problems

Minimization problems in which we are interested:

- Online Steiner Tree (OST),
  Online Single-Source Rent-or-Buy (OSRoB).

Online Problems

Minimization problems in which we are interested:

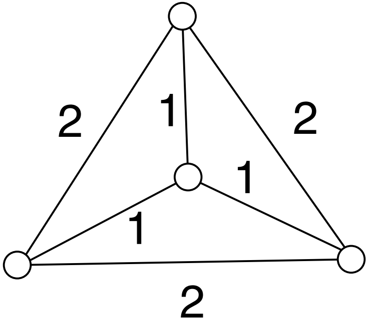
- Online Steiner Tree (OST),
  Online Single-Source Rent-or-Buy (OSRoB).
- Online Facility Location (OFL),
  Online Prize-Collecting Facility Location (OPFL).

Online Problems

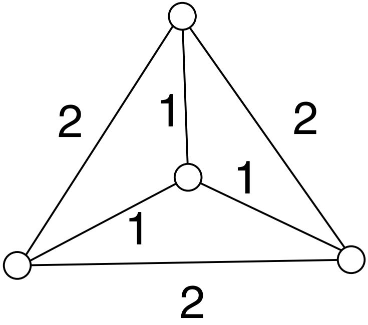
Minimization problems in which we are interested:

- Online Steiner Tree (OST),
  Online Single-Source Rent-or-Buy (OSRoB).
- Online Facility Location (OFL),
  Online Prize-Collecting Facility Location (OPFL).
- Online Connected Facility Location (OCFL).

Online Steiner Tree Problem

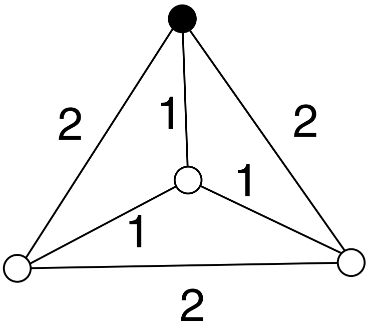


Online Steiner Tree Problem



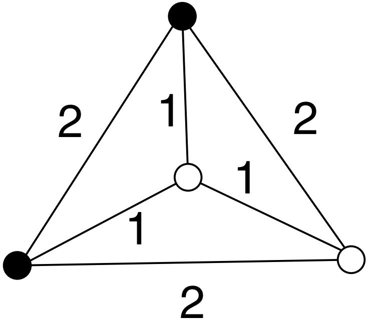
$$\min \sum_{e \in E(T)} d(e)$$

Online Steiner Tree Problem



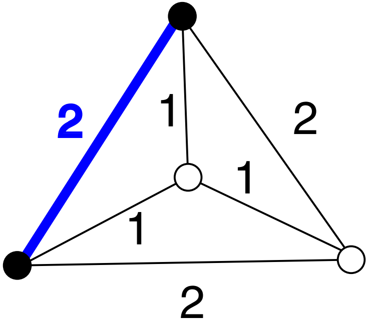
$$\min \sum_{e \in E(T)} d(e)$$

Online Steiner Tree Problem



$$\min \sum_{e \in E(T)} d(e)$$

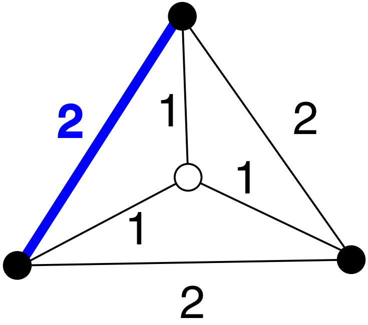
Online Steiner Tree Problem



$$\min \sum_{e \in E(T)} d(e)$$

Total cost $= 2$

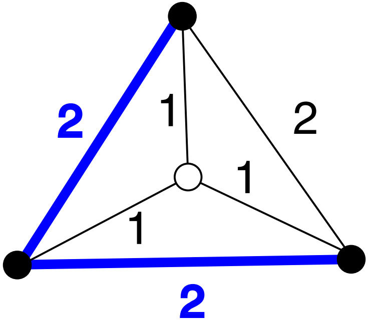
Online Steiner Tree Problem



$$\min \sum_{e \in E(T)} d(e)$$

Total cost $= 2$

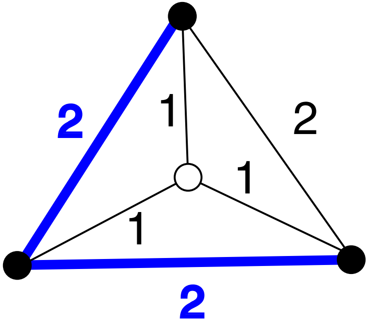
Online Steiner Tree Problem



$$\min \sum_{e \in E(T)} d(e)$$

Total cost $= 2 + 2$

Online Steiner Tree Problem



$$\min \sum_{e \in E(T)} d(e)$$

Total cost $= 2 + 2 = 4$.

Online Steiner Tree Results

Online Steiner Tree Results

There are $\mathrm{O}(\log n)$-competitive algorithms known for it.

Online Steiner Tree Results

There are $\mathrm{O}(\log n)$-competitive algorithms known for it.

We show a greedy $\lceil \log n \rceil$-competitive algorithm by Imase and Waxman [1991].

Online Steiner Tree Results

There are $\mathrm{O}(\log n)$-competitive algorithms known for it.

We show a greedy $\lceil \log n \rceil$-competitive algorithm by Imase and Waxman [1991].

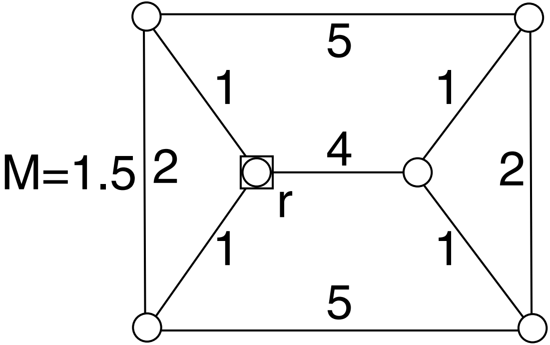
There is a lower bound of $\Omega(\log n)$ due to Imase and Waxman [1991].

Online Steiner Tree Algorithm

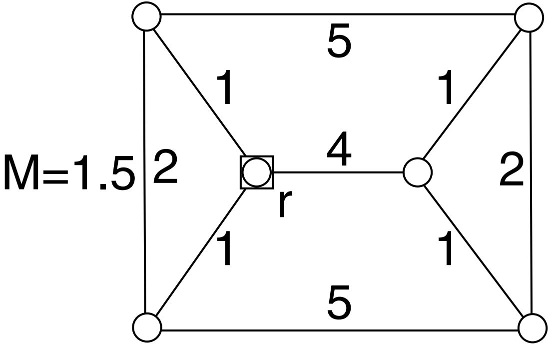
Algorithm 1: OST Algorithm.

Input: (G, d)
$T \leftarrow (\emptyset, \emptyset)$; $D \leftarrow \emptyset$;
while *a new terminal j arrives* **do**
    $T \leftarrow T \cup \{\mathrm{path}(j, V(T))\}$; /* connect */
    $D \leftarrow D \cup \{j\}$;
end
return T;

Online Single-Source Rent-or-Buy Problem

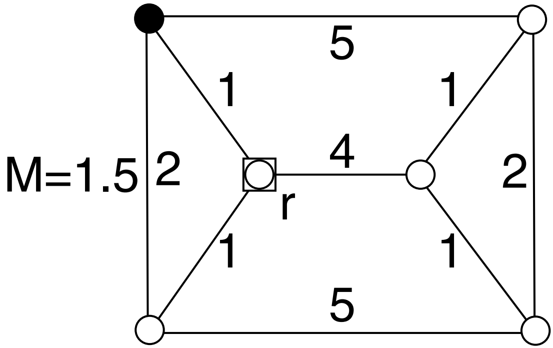


Online Single-Source Rent-or-Buy Problem



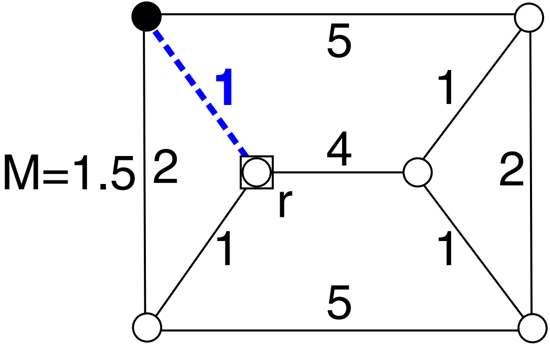
$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

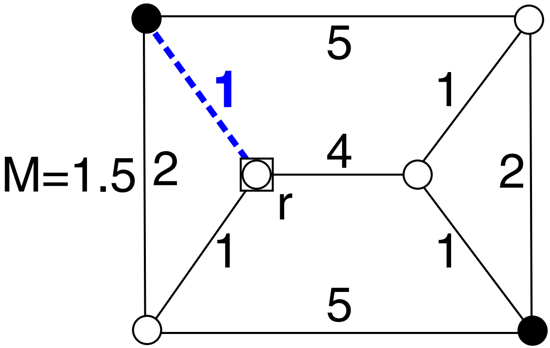
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1$

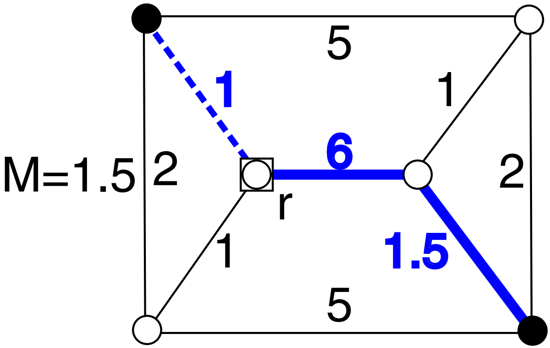
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1$

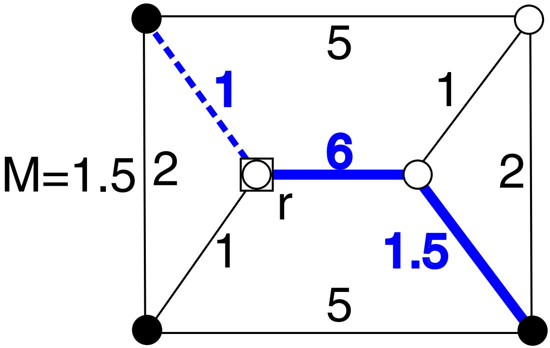
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 7.5$

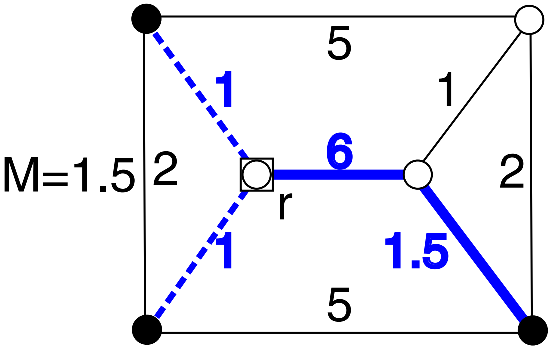
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 7.5$

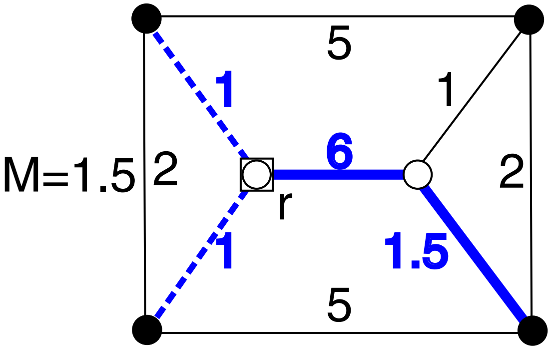
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 7.5 + 1$

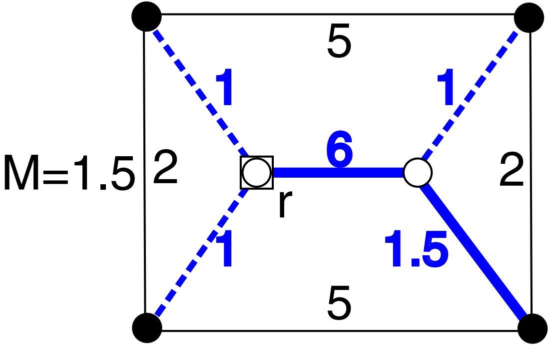
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 7.5 + 1$

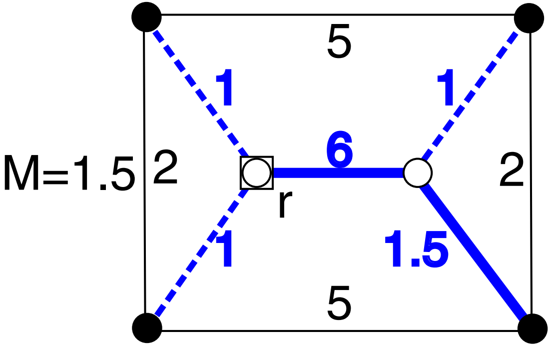
Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 7.5 + 1 + 1$

Online Single-Source Rent-or-Buy Problem



$$\min \sum_{j \in D} \sum_{e \in E(P(j))} d(e) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 7.5 + 1 + 1 = 10.5.$

Online Single-Source Rent-or-Buy Results

Online Single-Source Rent-or-Buy Results

A deterministic greedy algorithm is no better than M-competitive for this problem.

Online Single-Source Rent-or-Buy Results

A deterministic greedy algorithm is no better than M-competitive for this problem.

There is a sample-and-augment $2\lceil \log n \rceil$-competitive algorithm by Awerbuch, Azar and Bartal [2004].

Online Single-Source Rent-or-Buy Results

A deterministic greedy algorithm is no better than M-competitive for this problem.

There is a sample-and-augment $2\lceil \log n \rceil$-competitive algorithm by Awerbuch, Azar and Bartal [2004].

We show this algorithm and a simpler analysis for it.

Online Single-Source Rent-or-Buy Results

A deterministic greedy algorithm is no better than M-competitive for this problem.

There is a sample-and-augment $2\lceil \log n \rceil$-competitive algorithm by Awerbuch, Azar and Bartal [2004].

We show this algorithm and a simpler analysis for it.

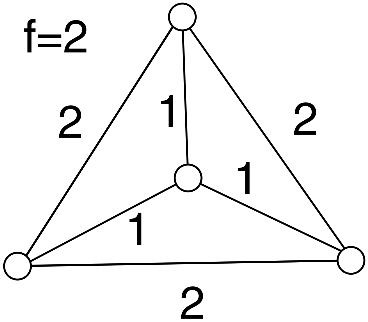
Since this problem is a generalization of the OST, the lower bound of $\Omega(\log n)$ applies to it.

Online Single-Source Rent-or-Buy Algorithm

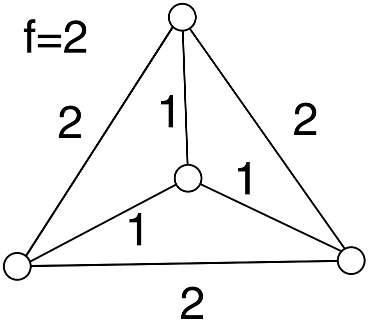
Algorithm 2: OSRoB Algorithm.

Input: (G, d, r, M)
$T \leftarrow (\{r\}, \emptyset)$; $P \leftarrow \emptyset$; $D \leftarrow \emptyset$; $D^m \leftarrow \emptyset$;
while *a new terminal j arrives* **do**
  include j in D^m with probability $\frac{1}{M}$;
  if $j \in D^m$ **then**
    $T \leftarrow T \cup \{\mathrm{path}(j, V(T))\}$; /* buy edges */
  end
  $P(j) \leftarrow \mathrm{path}(j, V(T))$; /* rent edges */
  $P \leftarrow P \cup \{P(j)\}$;
  $D \leftarrow D \cup \{j\}$;
end
return (P, T);

Online Facility Location Problem

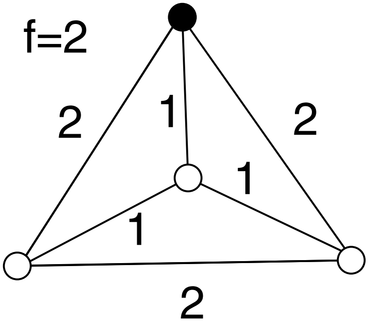


Online Facility Location Problem



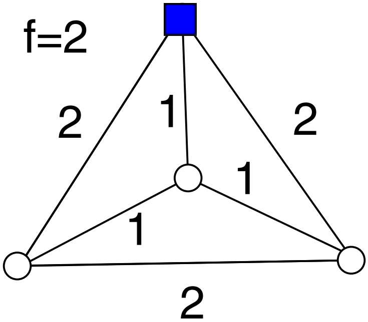
$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

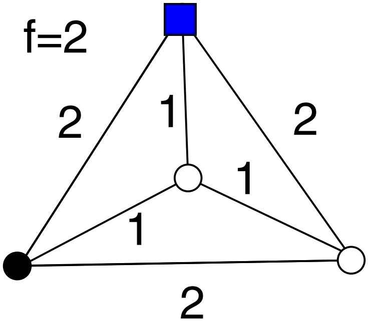
Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Total cost $= 2$

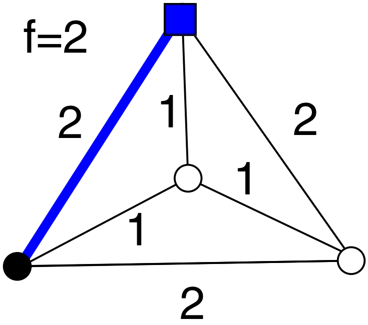
Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Total cost $= 2$

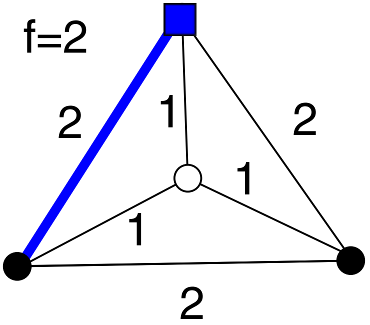
Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Total cost $= 2 + 2$

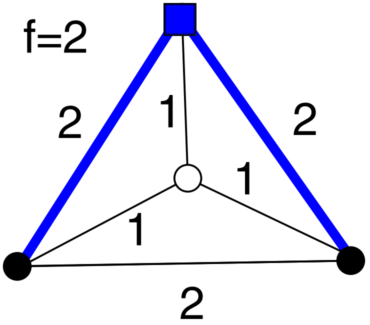
Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Total cost $= 2 + 2$

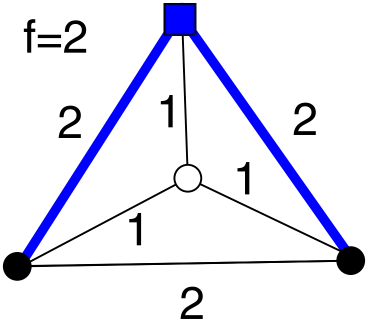
Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Total cost $= 2 + 2 + 2$

Online Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j))$$

Total cost $= 2 + 2 + 2 = 6.$

Online Facility Location Results

Online Facility Location Results

There are $\mathrm{O}(\log n)$-competitive algorithms known for it.

Online Facility Location Results

There are $\mathrm{O}(\log n)$-competitive algorithms known for it.

We show a primal-dual $(4 \log n)$-competitive algorithm by Fotakis [2007] and by Nagarajan and Williamson [2013].

Online Facility Location Results

There are $\mathrm{O}(\log n)$-competitive algorithms known for it.

We show a primal-dual $(4 \log n)$-competitive algorithm by Fotakis [2007] and by Nagarajan and Williamson [2013].

There is a lower bound of $\Omega\left(\frac{\log n}{\log \log n}\right)$ due to Fotakis [2008].

Online Facility Location LP Formulation

Online Facility Location LP Formulation

Linear programming relaxation

$$\begin{aligned} \min \quad & \textstyle\sum_{i\in F} f(i)y_i + \sum_{j\in D}\sum_{i\in F} d(j,i)x_{ji} \\ \text{s.t.} \quad & x_{ji} \le y_i && \text{for } j \in D \text{ and } i \in F, \\ & \textstyle\sum_{i\in F} x_{ji} \ge 1 && \text{for } j \in D, \\ & y_i \ge 0, x_{ji} \ge 0 && \text{for } j \in D \text{ and } i \in F, \end{aligned}$$

Online Facility Location LP Formulation

Linear programming relaxation

$$\begin{aligned} \min \quad & \sum_{i \in F} f(i) y_i + \sum_{j \in D} \sum_{i \in F} d(j,i) x_{ji} & \\ \text{s.t.} \quad & x_{ji} \le y_i & \text{for } j \in D \text{ and } i \in F, \\ & \sum_{i \in F} x_{ji} \ge 1 & \text{for } j \in D, \\ & y_i \ge 0, x_{ji} \ge 0 & \text{for } j \in D \text{ and } i \in F, \end{aligned}$$

and its dual

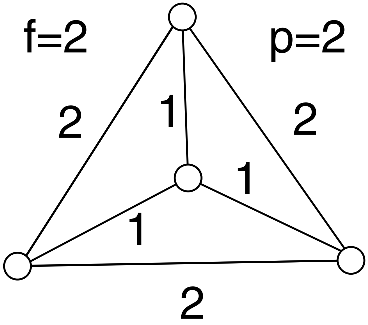
$$\begin{aligned} \max \quad & \sum_{j \in D} \alpha_j & \\ \text{s.t.} \quad & \sum_{j \in D} (\alpha_j - d(j,i))^+ \le f(i) & \text{for } i \in F, \\ & \alpha_j \ge 0 & \text{for } j \in D. \end{aligned}$$

Online Facility Location Algorithm

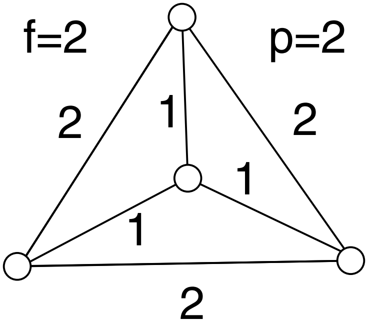
Algorithm 3: OFL Algorithm.

Input: (G, d, f, F)
$F^a \leftarrow \emptyset$; $D \leftarrow \emptyset$;
while *a new client j' arrives* **do**
  increase $\alpha_{j'}$ until one of the following happens:
  (a) $\alpha_{j'} = d(j', i)$ for some $i \in F^a$; /* connect only */
  (b) $f(i) = (\alpha_{j'} - d(j', i)) + \sum_{j \in D}(d(j, F^a) - d(j, i))^+$ for some $i \in F \setminus F^a$; /* open and connect */
  $F^a \leftarrow F^a \cup \{i\}$; $D \leftarrow D \cup \{j'\}$; $a(j') \leftarrow i$;
end
return (F^a, a);

Online Prize-Collecting Facility Location

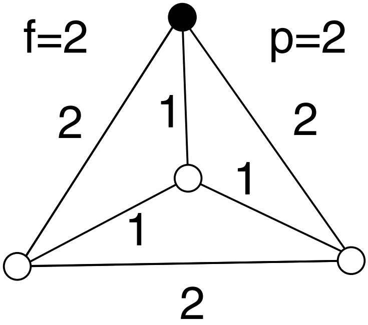


Online Prize-Collecting Facility Location



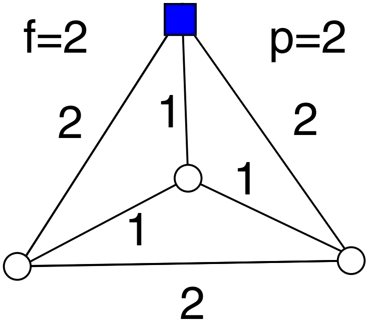
$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

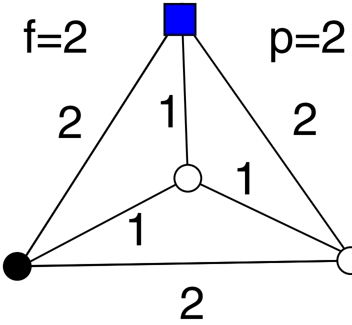
Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Total cost $= 2$

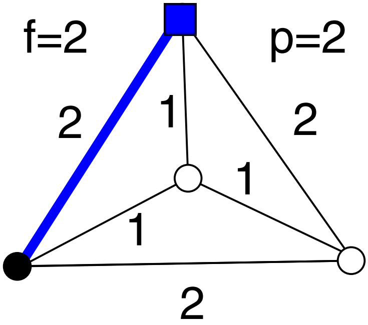
Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Total cost $= 2$

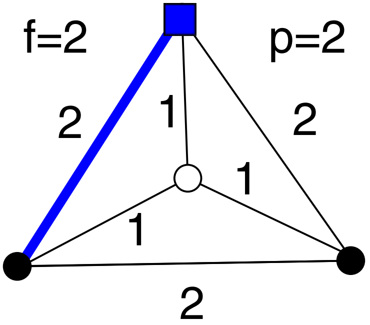
Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Total cost $= 2 + 2$

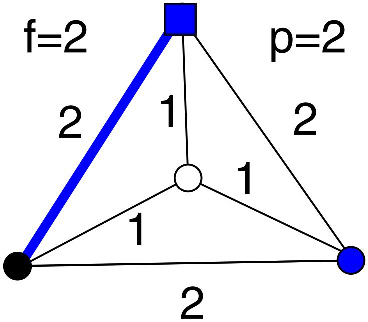
Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Total cost $= 2 + 2$

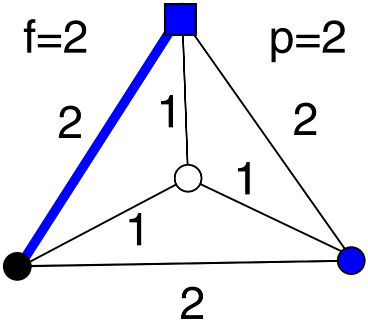
Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Total cost $= 2 + 2 + 2$

Online Prize-Collecting Facility Location



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D^c} d(j, a(j)) + \sum_{j \in D^p} p(j)$$

Total cost $= 2 + 2 + 2 = 6$.

OPFL Results

OPFL Results

We proposed the problem and showed a primal-dual $(6 \log n)$-competitive algorithm for it.

OPFL Results

We proposed the problem and showed a primal-dual $(6 \log n)$-competitive algorithm for it.

Since it is a generalization of the OFL, the lower bound of $\Omega\left(\frac{\log n}{\log \log n}\right)$ applies to it.

OPFL LP Formulation

OPFL LP Formulation

Linear programming relaxation

$$\begin{array}{lll}\min & \sum_{i \in F} f(i) y_i + \sum_{j \in D} \sum_{i \in F} d(j,i) x_{ji} + \sum_{j \in D} p(j) z_j & \\ \text{s.t.} & x_{ji} \leq y_i & \text{for } j \in D \text{ and } i \in F, \\ & \sum_{i \in F} x_{ji} + z_j \geq 1 & \text{for } j \in D, \\ & y_i \geq 0, x_{ji} \geq 0, z_j \geq 0 & \text{for } j \in D \text{ and } i \in F, \end{array}$$

OPFL LP Formulation

Linear programming relaxation

$$
\begin{array}{lll}
\min & \sum_{i \in F} f(i) y_i + \sum_{j \in D} \sum_{i \in F} d(j,i) x_{ji} + \sum_{j \in D} p(j) z_j & \\
\text{s.t.} & x_{ji} \leq y_i & \text{for } j \in D \text{ and } i \in F, \\
& \sum_{i \in F} x_{ji} + z_j \geq 1 & \text{for } j \in D, \\
& y_i \geq 0, x_{ji} \geq 0, z_j \geq 0 & \text{for } j \in D \text{ and } i \in F,
\end{array}
$$

and its dual

$$
\begin{array}{lll}
\max & \sum_{j \in D} \alpha_j & \\
\text{s.t.} & \sum_{j \in D} (\alpha_j - d(j,i))^+ \leq f(i) & \text{for } i \in F, \\
& \alpha_j \leq p(j) & \text{for } j \in D, \\
& \alpha_j \geq 0 & \text{for } j \in D.
\end{array}
$$

OPFL Algorithm

Algorithm 4: OPFL Algorithm.

Input: (G, d, f, p, F)

$D \leftarrow \emptyset$; $F^a \leftarrow \emptyset$;

while *a new client j' arrives* **do**

increase $\alpha_{j'}$ until one of the following happens:

(a) $\alpha_{j'} = d(j', i)$ for some $i \in F^a$; /* connect only */

(b) $f(i) = (\alpha_{j'} - d(j', i)) + \sum_{j \in D}(\min\{d(j, F^a), p(j)\} - d(j, i))^+$ for some $i \in F \setminus F^a$; /* open and connect */

(c) $\alpha_{j'} = p(j')$; /* pay the penalty */

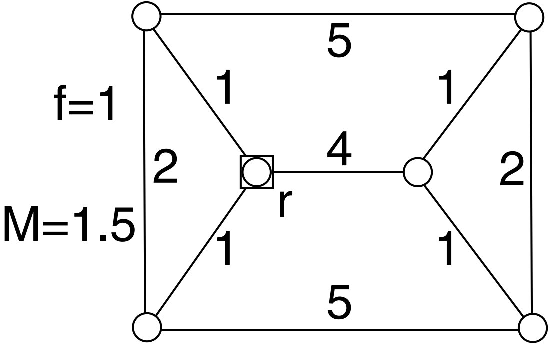
(in this case i is choose to be null, i.e., $\{i\} = \emptyset$)

$F^a \leftarrow F^a \cup \{i\}$; $D \leftarrow D \cup \{j'\}$; $a(j') \leftarrow i$;

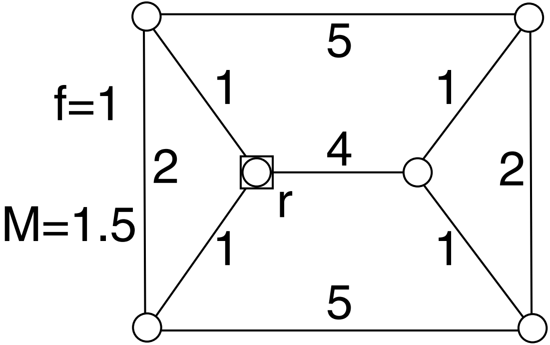
end

return (F^a, a);

Online Connected Facility Location Problem

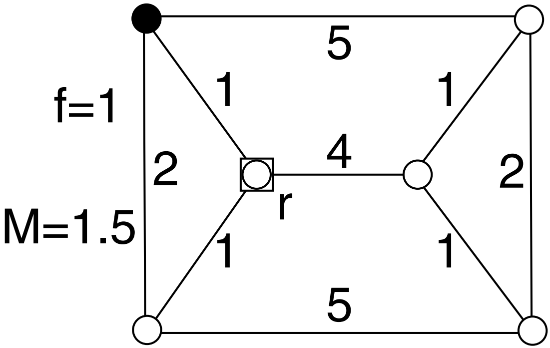


Online Connected Facility Location Problem



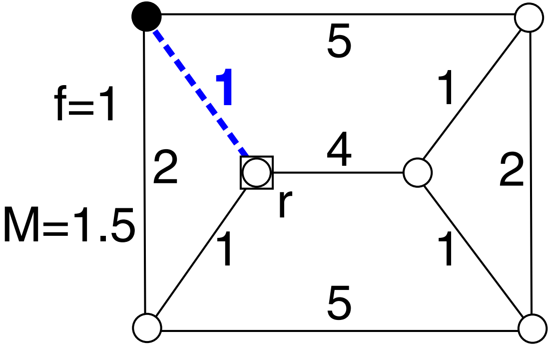
$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

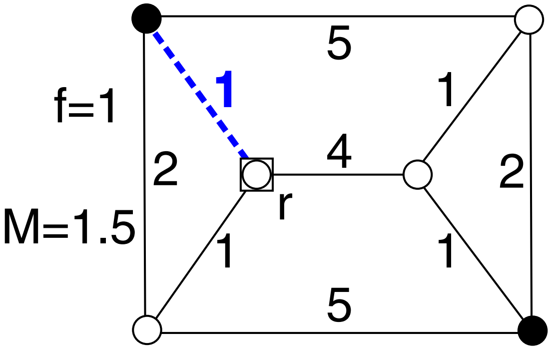
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1$

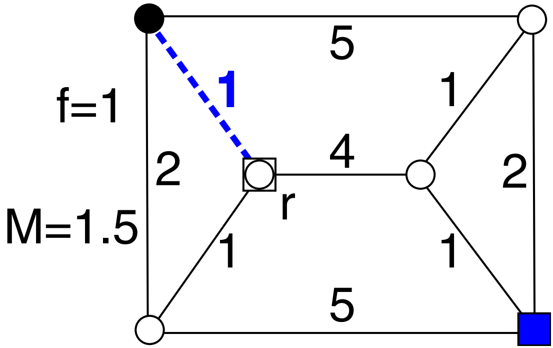
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1$

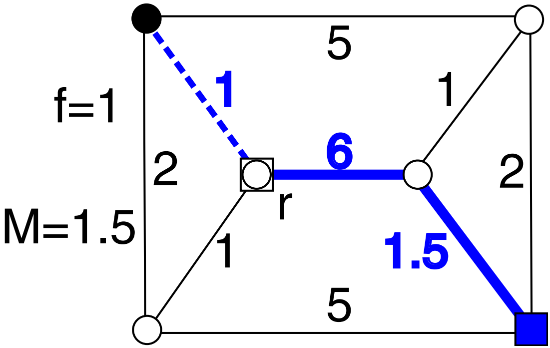
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1$

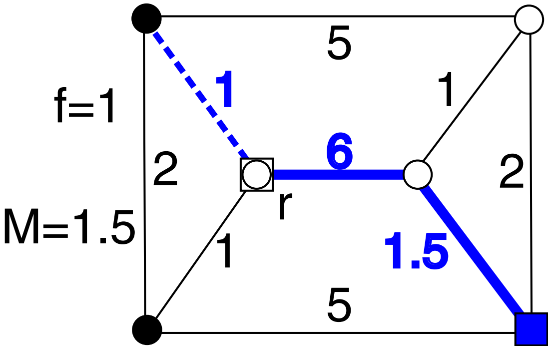
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1 + 7.5$

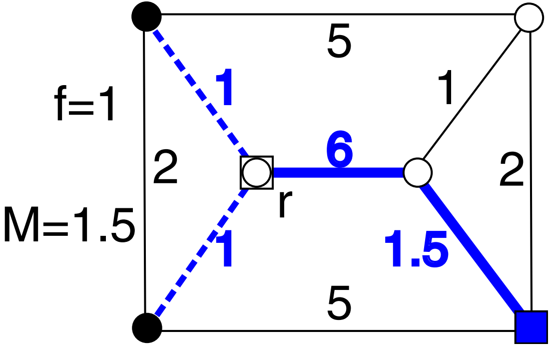
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1 + 7.5$

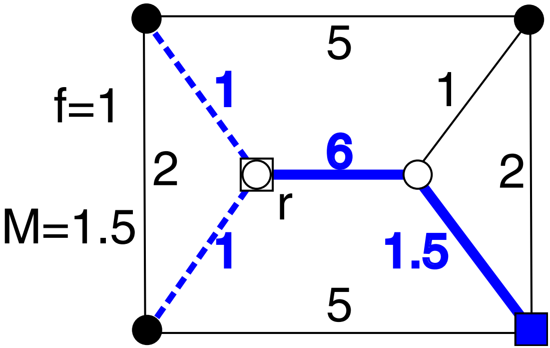
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1 + 7.5 + 1$

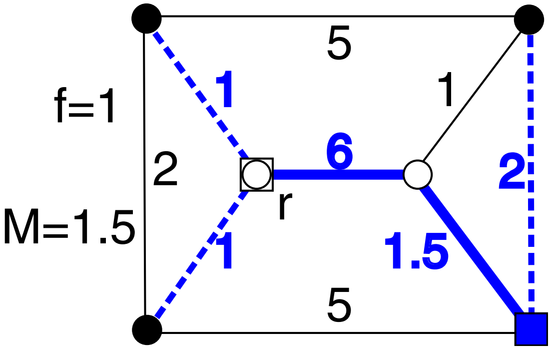
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1 + 7.5 + 1$

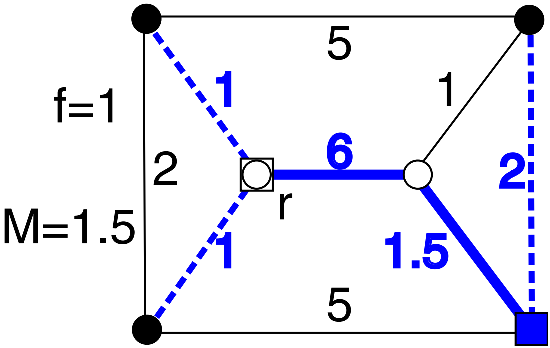
Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1 + 7.5 + 1 + 2$

Online Connected Facility Location Problem



$$\min \sum_{i \in F^a} f(i) + \sum_{j \in D} d(j, a(j)) + M \sum_{e \in E(T)} d(e)$$

Total cost $= 1 + 1 + 7.5 + 1 + 2 = 12.5.$

OCFL Results

OCFL Results

We proposed the problem and showed a sample-and-augment $18\lceil \log n \rceil$-competitive algorithm for it.

OCFL Results

We proposed the problem and showed a sample-and-augment $18\lceil \log n \rceil$-competitive algorithm for it.

We also showed that the same algorithm has competitive ratio $7\lceil \log n \rceil$ for the special case in which $M = 1$.

OCFL Results

We proposed the problem and showed a sample-and-augment $18\lceil \log n \rceil$-competitive algorithm for it.

We also showed that the same algorithm has competitive ratio $7\lceil \log n \rceil$ for the special case in which $M = 1$.

Since this problem is a generalization of the OST, the lower bound of $\Omega(\log n)$ applies to it.

OCFL Algorithm

Algorithm 5: OCFL Algorithm.

Input: (G, d, f, F, r, M)
set $f(r) \leftarrow 0$ and initialize $\mathrm{ALG}_{\mathrm{OFL}}$ with (G, d, f, F);
send r to $\mathrm{ALG}_{\mathrm{OFL}}$; $F^a \leftarrow \{r\}$; $T \leftarrow (\{r\}, \emptyset)$;
while *a new client j arrives* **do**
    send j to $\mathrm{ALG}_{\mathrm{OFL}}$; /* update virtual solution */
    include j in D^m with probability $\frac{1}{M}$;
    if $j \in D^m$ **then**
        $T \leftarrow T \cup \{\mathrm{path}(j, V(T))\}$; /* connect new facility */
        if $v(j)$ *is not opened* **then**
            $F^a \leftarrow F^a \cup \{v(j)\}$; $T \leftarrow T \cup \{(v(j), j)\}$;
        end
    end
    choose $i \in F^a$ that is closest to j; $D \leftarrow D \cup \{j\}$; $a(j) \leftarrow i$;
end
return (F^a, a, T);

Analysis of the OSRoB Algorithm

Analysis of the OSRoB Algorithm

We are going to show the following result.

Theorem

$$E[\mathrm{ALG}_{\mathrm{OSRoB}}(D_n)] \leq 2\lceil \log n \rceil \mathrm{OPT}_{\mathrm{SRoB}}(D_n) \ .$$

Analysis of the OSRoB Algorithm

Analysis of the OSRoB Algorithm

We want to compare

$$\mathrm{ALG}_{\mathrm{OSRoB}}(D_n) = \sum_{j \in D_n} \sum_{e \in E(P_n(j))} d(e) + M \sum_{e \in E(T_n)} d(e)$$
$$= R(D_n) + B(D_n) \ ,$$

Analysis of the OSRoB Algorithm

We want to compare

$$\mathrm{ALG}_{\mathrm{OSRoB}}(D_n) = \sum_{j \in D_n} \sum_{e \in E(P_n(j))} d(e) + M \sum_{e \in E(T_n)} d(e)$$
$$= R(D_n) + B(D_n) \ ,$$

with

$$\mathrm{OPT}_{\mathrm{SRoB}}(D_n) = \sum_{j \in D_n} \sum_{e \in E(P_n^*(j))} d(e) + M \sum_{e \in E(T_n^*)} d(e)$$
$$= R^*(D_n) + B^*(D_n) \ .$$

Remembering the OSRoB Algorithm

Algorithm 6: OSRoB Algorithm.

Input: (G, d, r, M)
$T \leftarrow (\{r\}, \emptyset)$; $P \leftarrow \emptyset$; $D \leftarrow \emptyset$; $D^m \leftarrow \emptyset$;
while *a new terminal j arrives* **do**
    include j in D^m with probability $\frac{1}{M}$;
    if $j \in D^m$ **then**
        $T \leftarrow T \cup \{\mathrm{path}(j, V(T))\}$; /* buy edges */
    end
    $P(j) \leftarrow \mathrm{path}(j, V(T))$; /* rent edges */
    $P \leftarrow P \cup \{P(j)\}$;
    $D \leftarrow D \cup \{j\}$;
end
return (P, T);

Analysis of the OSRoB Algorithm

Analysis of the OSRoB Algorithm

Note that

$$R(D_n) = \sum_{j \in D_n} \sum_{e \in E(P_n(j))} d(e) = \sum_{j \in D_n \setminus D_n^m} d(j, V(T_{n(j)})) = \sum_{j \in D_n} r(j) \; .$$

Analysis of the OSRoB Algorithm

Note that

$$R(D_n) = \sum_{j \in D_n} \sum_{e \in E(P_n(j))} d(e) = \sum_{j \in D_n \setminus D_n^m} d(j, V(T_{n(j)})) = \sum_{j \in D_n} r(j) \ .$$

Also, note that

$$B(D_n) = M \sum_{e \in E(T_n)} d(e) = M \sum_{j \in D_n^m} d(j, V(T_{n(j)-1})) = \sum_{j \in D_n} b(j) \ .$$

Analysis of the OSRoB Algorithm

Now we bound the expected buying cost.

Lemma

$E[B(D_n)] \leq \lceil \log n \rceil \mathrm{OPT}_{\mathrm{SRoB}}(D_n)$.

Analysis of the OSRoB Algorithm

Demonstration.

Analysis of the OSRoB Algorithm

Demonstration.

$$E[B(D_n)] \;=\; E\left[\sum_{j \in D_n^m} b(j)\right] \leq ME\left[\sum_{j \in D_n^m} d(j, V(T_{n(j)-1}))\right]$$

Analysis of the OSRoB Algorithm

Demonstration.

$$
\begin{aligned}
E[B(D_n)] &= E\left[\sum_{j \in D_n^m} b(j)\right] \leq ME\left[\sum_{j \in D_n^m} d(j, V(T_{n(j)-1}))\right] \\
&\leq ME[\mathrm{ALG}_{\mathrm{OST}}(D_n^m)] \leq M\lceil \log n \rceil E[\mathrm{OPT}_{\mathrm{ST}}(D_n^m)]
\end{aligned}
$$

Analysis of the OSRoB Algorithm

Demonstration.

$$
\begin{aligned}
E[B(D_n)] &= E\left[\sum_{j \in D_n^m} b(j)\right] \leq ME\left[\sum_{j \in D_n^m} d(j, V(T_{n(j)-1}))\right] \\
&\leq ME[\mathrm{ALG}_{\mathrm{OST}}(D_n^m)] \leq M\lceil \log n \rceil E[\mathrm{OPT}_{\mathrm{ST}}(D_n^m)] \\
&\leq M\lceil \log n \rceil \left(\frac{B^*(D_n)}{M} + E\left[\sum_{j \in D_n^m} d(j, V(T^*_{n(j)}))\right] \right)
\end{aligned}
$$

Analysis of the OSRoB Algorithm

Demonstration.

$$
\begin{aligned}
E[B(D_n)] &= E\left[\sum_{j \in D_n^m} b(j)\right] \leq ME\left[\sum_{j \in D_n^m} d(j, V(T_{n(j)-1}))\right] \\
&\leq ME[\mathrm{ALG}_{\mathrm{OST}}(D_n^m)] \leq M\lceil \log n \rceil E[\mathrm{OPT}_{\mathrm{ST}}(D_n^m)] \\
&\leq M\lceil \log n \rceil \left(\frac{B^*(D_n)}{M} + E\left[\sum_{j \in D_n^m} d(j, V(T^*_{n(j)}))\right]\right) \\
&= \lceil \log n \rceil \left(B^*(D_n) + M \sum_{j \in D_n} \frac{d(j, V(T^*_{n(j)}))}{M}\right)
\end{aligned}
$$

Analysis of the OSRoB Algorithm

Demonstration.

$$
\begin{aligned}
E[B(D_n)] &= E\left[\sum_{j \in D_n^m} b(j)\right] \leq ME\left[\sum_{j \in D_n^m} d(j, V(T_{n(j)-1}))\right] \\
&\leq ME[\mathrm{ALG}_{\mathrm{OST}}(D_n^m)] \leq M\lceil \log n \rceil E[\mathrm{OPT}_{\mathrm{ST}}(D_n^m)] \\
&\leq M\lceil \log n \rceil \left(\frac{B^*(D_n)}{M} + E\left[\sum_{j \in D_n^m} d(j, V(T^*_{n(j)}))\right] \right) \\
&= \lceil \log n \rceil \left(B^*(D_n) + M \sum_{j \in D_n} \frac{d(j, V(T^*_{n(j)}))}{M} \right) \\
&= \lceil \log n \rceil \left(B^*(D_n) + R^*(D_n) \right)
\end{aligned}
$$

Analysis of the OSRoB Algorithm

Demonstration.

$$
\begin{aligned}
E[B(D_n)] &= E\left[\sum_{j\in D_n^m} b(j)\right] \le ME\left[\sum_{j\in D_n^m} d(j, V(T_{n(j)-1}))\right] \\
&\le ME[\mathrm{ALG}_{\mathrm{OST}}(D_n^m)] \le M\lceil \log n\rceil E[\mathrm{OPT}_{\mathrm{ST}}(D_n^m)] \\
&\le M\lceil \log n\rceil \left(\frac{B^*(D_n)}{M} + E\left[\sum_{j\in D_n^m} d(j, V(T^*_{n(j)}))\right]\right) \\
&= \lceil \log n\rceil \left(B^*(D_n) + M\sum_{j\in D_n} \frac{d(j, V(T^*_{n(j)}))}{M}\right) \\
&= \lceil \log n\rceil \left(B^*(D_n) + R^*(D_n)\right) \\
&= \lceil \log n\rceil \mathrm{OPT}_{\mathrm{SRoB}}(D_n) \ .
\end{aligned}
$$

And now we bound the expected renting cost.

Lemma

$E[R(D_n)] \leq E[B(D_n)]$.

Analysis of the OSRoB Algorithm

Demonstration.

Analysis of the OSRoB Algorithm

Demonstration.

Let $E[x(j)|n(j)-1]$ be the random variable $x(j)$ conditioned to the first $n(j)-1$ random choices of the algorithm. Thus

Analysis of the OSRoB Algorithm

Demonstration.

Let $E[x(j)|n(j)-1]$ be the random variable $x(j)$ conditioned to the first $n(j)-1$ random choices of the algorithm. Thus

$$E[r(j)|n(j)-1] \;=\; \frac{M-1}{M}d(j, V(T_{n(j)}))$$

Analysis of the OSRoB Algorithm

Demonstration.

Let $E[x(j)|n(j)-1]$ be the random variable $x(j)$ conditioned to the first $n(j)-1$ random choices of the algorithm. Thus

$$\begin{aligned} E[r(j)|n(j)-1] &= \frac{M-1}{M} d(j, V(T_{n(j)})) \\ &\leq d(j, V(T_{n(j)-1})) \end{aligned}$$

Analysis of the OSRoB Algorithm

Demonstration.

Let $E[x(j)|n(j)-1]$ be the random variable $x(j)$ conditioned to the first $n(j)-1$ random choices of the algorithm. Thus

$$\begin{aligned} E[r(j)|n(j)-1] &= \frac{M-1}{M} d(j, V(T_{n(j)})) \\ &\leq d(j, V(T_{n(j)-1})) \\ &= \frac{1}{M} M d(j, V(T_{n(j)-1})) \leq E[b(j)|n(j)-1] \ . \end{aligned}$$

Analysis of the OSRoB Algorithm

Demonstration.

Let $E[x(j)|n(j)-1]$ be the random variable $x(j)$ conditioned to the first $n(j)-1$ random choices of the algorithm. Thus

$$\begin{aligned} E[r(j)|n(j)-1] &= \frac{M-1}{M} d(j, V(T_{n(j)})) \\ &\leq d(j, V(T_{n(j)-1})) \\ &= \frac{1}{M} M d(j, V(T_{n(j)-1})) \leq E[b(j)|n(j)-1] \ . \end{aligned}$$

Since this holds for any outcome of the first $n(j)-1$ random choices of the algorithm, it holds unconditionally. So

Analysis of the OSRoB Algorithm

Demonstration.

Let $E[x(j)|n(j)-1]$ be the random variable $x(j)$ conditioned to the first $n(j)-1$ random choices of the algorithm. Thus

$$\begin{aligned} E[r(j)|n(j)-1] &= \frac{M-1}{M} d(j, V(T_{n(j)})) \\ &\leq d(j, V(T_{n(j)-1})) \\ &= \frac{1}{M} M d(j, V(T_{n(j)-1})) \leq E[b(j)|n(j)-1] \ . \end{aligned}$$

Since this holds for any outcome of the first $n(j)-1$ random choices of the algorithm, it holds unconditionally. So

$$E[R(D_n)] = \sum_{j \in D_n} E[r(j)] \leq \sum_{j \in D_n} E[b(j)] = E[B(D_n)] \ .$$

Analysis of the OSRoB Algorithm

Demonstration.

Analysis of the OSRoB Algorithm

Demonstration.

Using the two previous lemmas we have that:

Analysis of the OSRoB Algorithm

Demonstration.

Using the two previous lemmas we have that:

$$E[\mathrm{ALG}_{\mathrm{OSRoB}}(D_n)] \leq E[R(D_n)] + E[B(D_n)]$$

Analysis of the OSRoB Algorithm

Demonstration.

Using the two previous lemmas we have that:

$$\begin{aligned} E[\mathrm{ALG}_{\mathrm{OSRoB}}(D_n)] &\leq E[R(D_n)] + E[B(D_n)] \\ &\leq 2\lceil \log n \rceil \mathrm{OPT}_{\mathrm{SRoB}}(D_n) \ . \end{aligned}$$

■

Acknowledgements

Thank you!

References

D. Fotakis.
A Primal-Dual Algorithm for Online Non-Uniform Facility Location.
Journal of Discrete Algorithms, Volume 5, Pages 141–148, Elsevier, 2007.

C. Nagarajan and D.P. Williamson.
Offline and Online Facility Leasing.
Discrete Optimization, Volume 10, Number 4, Pages 361–370, 2013.

D. Fotakis
On the Competitive Ratio for Online Facility Location.
Algorithmica, Volume 50, Pages 1–57, 2008.

References (cont.)

M. Imase and M.B. Waxman.
Dynamic Steiner Tree Problem.
SIAM Journal on Discrete Mathematics, Volume 4, Pages 369–384, 1991.

B. Awerbuch, Y. Azar and Y. Bartal.
On-line Generalized Steiner Problem.
Theoretical Computer Science, Volume 324, Pages 313–324, 2004.