The Price of Leasing Online
TOLA 2014

Christine Markarian
July 7, 2014

Joint work with:
Sebastian Abshoff
Peter Kling
Friedhelm Meyer auf der Heide
Outline

- Parking Permit Problem
- Infrastructure Leasing Problems
- Set Cover Leasing
- The Price of Leasing Online
Parking Permit Problem

sunny day → walk

rainy day → drive

[Meyerson - FOCS 2003]
Parking Permit Problem

- adversary gives **sunny** or **rainy** on each day
- $K$ permit *lease* types (Ex. daily, weekly, monthly, yearly)
- yearly permit is the **most expensive but cheapest per day**

Online algorithm

**Which** permits do I buy & **when** in order to provide every rainy day with a permit?

**quality of online algorithm → competitive factor $\alpha$**

- **adversary** reveals in each step part of overall input
- $\alpha = \max \frac{\text{cost of Online algorithm}}{\text{cost of Optimal Offline algorithm}}$ over all input instances

[**Meyerson - FOCS 2003**]

**Optimal Offline** knows the future in advance
Parking Permit Problem

<table>
<thead>
<tr>
<th>Lower bounds</th>
<th>Upper bounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Omega(K) ) deterministic</td>
<td>( O(K) ) deterministic</td>
</tr>
<tr>
<td>( \Omega(\log K) ) randomized</td>
<td>( O(\log K) ) randomized</td>
</tr>
</tbody>
</table>

Deterministic algorithm
For each rainy day, buy a 1-day permit, until there is some \((k \in K)\)-interval where the optimum offline solution for the sequence of days seen so far, would buy a \(k\)-day permit. In this case, also buy a \(k\)-day permit.

Randomized algorithm
Compute an \(O(\log K)\)-competitive fractional solution and then convert it into a randomized integer solution which maintains the \(O(\log K)\)-competitive factor.

[Meyerson - FOCS 2003]
Outline

- Parking Permit Problem
- Infrastructure Leasing Problems
- Set Cover Leasing
- The Price of Leasing Online
Infrastructure Leasing Problems

Leasing Company

Client 1
Client 2
Client 3

Provider 1
Provider 2
Provider 3

Long lease or short lease
..... ?
Infrastructure Leasing Problems

- Almost any online infrastructure problem can be considered with a leasing aspect.

- Anthony & Gupta generalized the Parking Permit Problem
  - Facility Leasing
  - Steiner Tree Leasing
  - Set Cover Leasing

  & gave offline algorithms to the problems…

[Anthony et al.- IPCO 2007]
Outline

- Parking Permit Problem
- Infrastructure Leasing Problems
- Set Cover Leasing
- The Price of Leasing Online
Set Cover Leasing

Online Set cover

- \( U = \{e_1, e_2, \ldots, e_n\} \)
- family \( F = \{S_1, S_2, \ldots, S_m\} \) of subsets of \( U \) and a cost associated with each subset

- an element \( e \in U \) arrives

-- choose sets from \( F \) to cover each arriving element \( e \in U \) & minimize cost of sets --

Set Cover Leasing

- \( U = \{e_1, e_2, \ldots, e_n\} \)
- family \( F = \{S_1, S_2, \ldots, S_m\} \) of subsets of \( U \). Each set in \( F \) can be leased for \( K \) different periods of time such that leasing a set \( S \) for a period \( k \):
  - incurs a cost \( c_{kS} \)
  - allows \( S \) to cover its elements for the next \( l_k \) time steps

- an element \( e \in U \) arrives

-- lease sets from \( F \) to cover each arriving element \( e \in U \) & minimize cost of sets --

generalizes Online Set Cover \((K = 1)\)
- one infinite lease -
Set Cover Leasing

- $e_1$ arrives at time $t$
- $e_1 \in \{S_3, S_5, S_8\}$
Set Cover Leasing

Ex. servers/clients in a computer network

Once a server is installed, it serves its clients **forever** without additional costs…
[Online set cover]

If servers are **leased** instead & can serve their clients only during the time they are leased…
[Set cover leasing]
Set Cover Leasing

Lower bounds
- none
- related problems
  - Online Metric Facility Location: $\Omega\left(\frac{\log n}{\log \log n}\right)$
    [ICALP 2003]
  - Online Set Cover: $\Omega\left(\frac{\log n \log m}{\log \log n + \log \log m}\right)$
    [STOC 2003]

Upper bounds
- Online Metric Facility Leasing: $O(K \log n)$
  [IPCO 2008]
- An algorithm for Online Facility Leasing: $O(l_{\text{max}} \log l_{\text{max}})$
  [SIROCCO 12]
- Randomized Online Algorithms for Set Cover Leasing Problems: $O(\log(mK) \log n)$
  [submitted to WAOA]
Set Cover Leasing

Algorithm {Set Cover Leasing}
Maintain a fraction \( f_{SkT} \) for each set \((S, k, t)\)
- set to 0 initially
- non-decreasing throughout algorithm

Maintain for each set \((S, k, t)\)
- \(2[\log(n + 1)]\) independent random variables \(X_{(SkT)(q)}\) in \([0, 1]\)
- Let \(\mu_{SkT} = \min X_{(SkT)(q)}, 1 \leq q \leq 2[\log(n + 1)]\)

\((j, t)\) arrives,

i. (fractional) If \(\sum_{(S,K,T)\in Q_j} f_{SkT} < 1\), do the following increment

\[
\text{while } \sum_{(S,K,T)\in Q_j} f_{SkT} < 1; \\
f_{SkT} = f_{SkT} \cdot \left(1 + \frac{1}{c_{ks}}\right) + \frac{1}{|Q_j| \cdot c_{ks}}
\]

ii. (integer) Lease \((S, k, T) \in Q_j\) with \(f_{SkT} > \mu_{SkT}\)

iii. If \((j, t)\) is not covered by some set in \(Q_j\)

Lease the cheapest \((S, k, T) \in Q_j\)

\[O(\log (dK) \log n) - \text{competitive}\]

(i) fractional \(\leq O(\log(dK)) \cdot \text{Opt}\)

(ii) randomized integer \(\leq O(\log n) \cdot \text{fractional}\)

(iii) step iii adds an expected cost of \(\text{Opt}/n\)
Set Cover Leasing

\( O(\log (dK) \log n) \) – competitive

(i) fractional \leq O(\log(dK)) \cdot \text{Opt}
(ii) randomized integer \leq O(\log n) \cdot \text{fractional}
(iii) step iii adds an expected cost of \( \text{Opt} / n \)

Proof: (i)

- an increment adds at most 2 to the fractional cost

\[
\sum_{(S,k,T) \in Q} \frac{c_S \cdot f_{Skt}}{c_S} \cdot \frac{1}{|Q| \cdot c_S} = \sum_{(S,k,T) \in Q} f_{Skt} < 1 + 1 \leq 2
\]

- the total number of increments in the algorithm is \( O(\log(dK)) \cdot \text{Opt} \)
  - At any time the algorithm decides to make an increment, \( \exists S_{opt} \) which is a candidate and therefore increases its fraction \( f_{S_{opt}kt} \)
  - After \( O(c_S \cdot \log|Q|) \) increments, \( f_{S_{opt}kt} > 1 \) \( \rightarrow \sum_{(S,k,T) \in Q} f_{Skt} > 1 \)
  - \( |Q| \leq d \cdot K \) [Interval Model: Same sets same leases do not coincide]

Algorithm {i-cover}

(\( j, t \) arrives)

i. (fractional) If \( \sum_{(S,k,T) \in Q_j} f_{Skt} < 1 \), do the following increment

\[
f_{Skt} = f_{Skt} \cdot \left(1 + \frac{1}{c_{ks}}\right) + \frac{1}{|Q_j| \cdot c_{ks}}
\]

ii. (integer) Lease \( (S, k, T) \in Q_j \) with \( f_{Skt} > \mu_{Skt} \)

iii. If \( (j, t) \) is not covered by some set in \( Q_j \)

Lease the cheapest \( (S, k, T) \in Q_j \)

Parking Permit Problem
Set Cover Leasing

**Proof: (ii) randomized integer** $\leq O(\log n) \cdot \text{fractional}$
- Probability to lease a set is $Pr\left( f_{Skt} > \mu_{Skt} \right)$
- $\mu_{Skt} = \min X(Skt)(q), 1 \leq q \leq 2[\log(n + 1)]$

**Proof: (iii) step iii adds an expected cost of $0pt/n$**
- [Algorithm leases the cheapest $(S, k, T) \in Q]$ $c_s \leq 0pt$
- Probability that an element is not covered [for a single $q$] is at most
  \[
  \prod_{(Skt) \in Q} (1 - f_{Skt}) \leq e^{-\sum_{(S,k,T)\in Q} f_{Skt}} \leq 1/e
  \]
- Probability that an element is not covered is at most $1/n^2$
- Additional expected cost $\leq n \cdot \frac{1}{n^2} \cdot 0pt$

$\rightarrow O(\log (dK) \log n) - \text{competitive}$
The Price of Leasing Online

Lower bounds

- **Online Set Cover**: $\Omega\left(\frac{\log n \log m}{\log \log n + \log \log m}\right) + \Omega(K)
- **Parking Permit Problem**: $\Omega(K)$

- **Online Facility Location**: $\Omega\left(\frac{\log n}{\log \log n}\right) + \Omega(K)$
- **Parking Permit Problem**: $\Omega(K)$

....

Leasing algorithms so far use techniques from non-leasing algorithms & Parking Permit Problem...

Does leasing impose an inherent difficulty?

What is the price we pay for leasing?
Thank you for your attention!

Christine Markarian
July 7, 2014