The Maximal Covering Location Problem under different perspectives: Upgrading and Uncertainty

Marta Baldomero Naranjo [1] Jörg Kalcsics[2] Alfredo Marín[3]
Antonio M. Rodríguez Chía [1]

[1]Universidad de Cádiz

[2]University of Edinburgh

[3]Universidad de Murcia

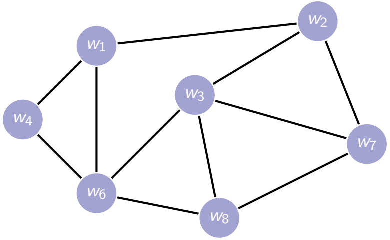
June 23-26, 2021
Advances on logistics and transportation problems on complex networks: Evaluation and conclusions

Outline

Maximal Covering Location Problem

MCLP aims to maximise the covered demand locating p facilities.

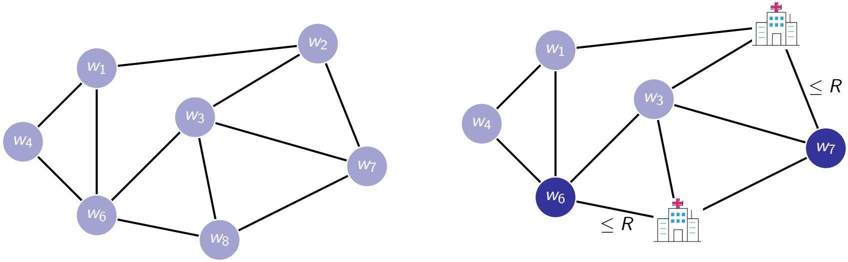
R. Church and C. ReVelle.
The maximal covering location problem.
Papers of the Regional Science Association, 3:101–118, 1974.



Maximal Covering Location Problem

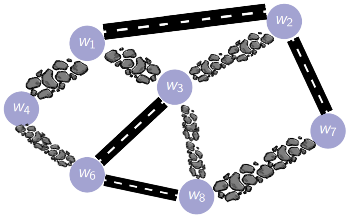
MCLP aims to maximise the covered demand locating p facilities.

R. Church and C. ReVelle.
The maximal covering location problem.
Papers of the Regional Science Association, 3:101–118, 1974.



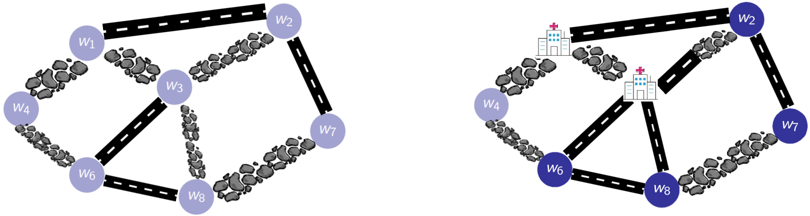
Upgrading edges MCLP

Upgrading MCLP aims to maximise the covered demand locating p facilities and reducing the length of some edges within a budget.



Upgrading edges MCLP

Upgrading MCLP aims to maximise the covered demand locating p facilities and reducing the length of some edges within a budget.



Applications

- Public government: Locate p public services at the same time that some roads are improved reducing travel times.

Applications

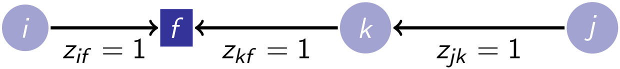
- Public government: Locate p public services at the same time that some roads are improved reducing travel times.
- In shopping centres, airports, train stations: Locate services as defibrillators, information posts while installing passenger conveyors or escalators to reduce travel times.

Formulations

- (Flow-Cov)
  - 3-index var.: if a path of length $\leq R$ from i to j traverses arc a.
  - 2-index var.: if node i is assigned to a facility at node j (assignment variables).

Formulations

- (Flow-Cov)
  - 3-index var.: if a path of length $\leq R$ from i to j traverses arc a.
  - 2-index var.: if node i is assigned to a facility at node j (assignment variables).
- (Path)
  - 2-index var.: if node j is the next node on a path of length $\leq R$ from i to a facility.



Formulations

- (Flow-Cov)
  - 3-index var.: if a path of length $\leq R$ from i to j traverses arc a.
  - 2-index var.: if node i is assigned to a facility at node j (assignment variables).
- (Path)
  - 2-index var.: if node j is the next node on a path of length $\leq R$ from i to a facility.
- (Path-Cov)
  - 2-index var.: if node j is the next node on a path of length $\leq R$ from i to a facility.
  - 2-index var.: if node i is assigned to a facility at node j (assignment variables).

Preprocessing

Formulations: (Flow-Cov) and (Path-Cov).

Preprocessing

Formulations: (Flow-Cov) and (Path-Cov).
Set the assignment variable to zero (remove) if:

- $d(i,j) > R$ even reducing the maximum amount allowed in all edges.
- $d(i,j) > R$ even reducing the maximum budget.

Preprocessing

Formulations: (Flow-Cov) and (Path-Cov).
Set the assignment variable to zero (remove) if:

- $d(i, j) > R$ even reducing the maximum amount allowed in all edges.
- $d(i, j) > R$ even reducing the maximum budget.

Several procedures to find a balance between accuracy and computational time.

Valid inequalities

- An adaptation of closest assignment constraints.

Valid inequalities

- An adaptation of closest assignment constraints.
- Strengthening several families of constraints.

Valid inequalities

- An adaptation of closest assignment constraints.
- Strengthening several families of constraints.
    - Separation procedure.

Computational Experiments

- C++ using CPLEX 20.1.0 with Concert Technology.
  - Time limit 1800 seconds.
- Intel(R) Xeon(R) W-2135 CPU 3.70 GHz 32 GB RAM.

Data

- Complete graphs
- OR-Library: pmed
  - $p \in \{1, n/10, n/20\}$,
  - $R \in \{50\% DT_{\text{MCLP}}, 60\% DT_{\text{MCLP}}, 70\% DT_{\text{MCLP}}\}$,
  - $u_e \in (0, 30\% \ell_e)$,
  - $B \in \{5\% B_{max}, 1\% B_{max}, 0.5\% B_{max}\}$.

$$B_{max} = \sum_{t=1}^{\min(n-p,m)} u_{e_{\sigma(t)}} c_{e_{\sigma(t)}},$$

where ρ is a permutation of set E such that

$$c_{e_{\rho(1)}} u_{e_{\rho(1)}} \geq c_{e_{\rho(2)}} u_{e_{\rho(2)}} \geq \ldots \geq c_{e_{\rho(m)}} u_{e_{\rho(m)}} .$$

Performance of the preprocessing phase

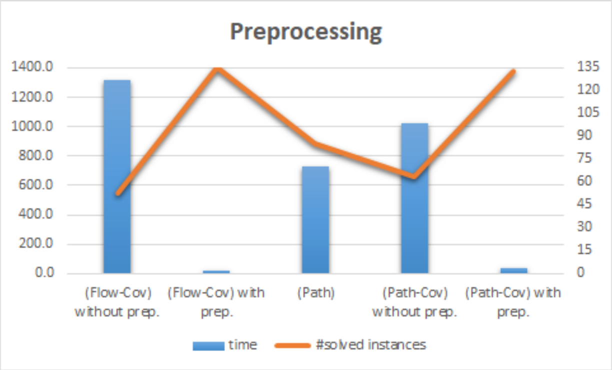


Figure: Performance on graph40

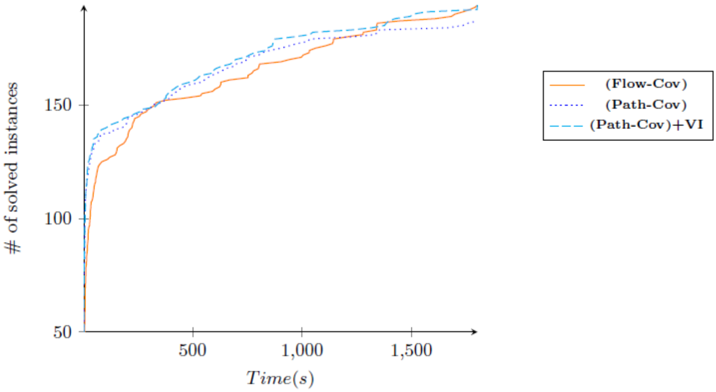


Figure: Performance profile graph of #solved instances using (Flow-Cov), (Path-Cov), (Path-Cov) + VI formulations on graph100 and graph120.

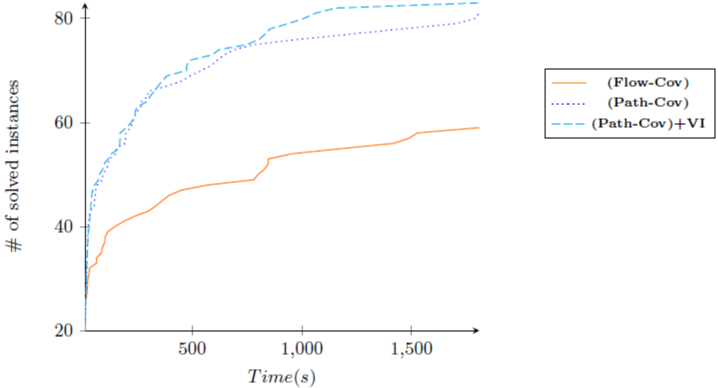


Figure: Performance profile graph of #solved instances using (Flow-Cov), (Path-Cov), (Path-Cov) + VI formulations on pmedb ($|V| = 200, |E| = 777.8$).

Conclusions and future work

- We present three MIP formulations for the upgrading maximal covering location problem.

Future work:

- Covering criteria: gradual covering, cooperative covering, etc.
- Location criteria: p-median, p-center, etc.

M. Baldomero-Naranjo, J. Kalcsics, A. Marín, and A. M. Rodríguez-Chía.
Upgrading edges in the Maximal Covering Location Problem.
Submitted

Uncertainty

A. Marín, L. I. Martínez-Merino, A. M. Rodríguez-Chía, and F. Saldanha-da-Gama.
Multi-period stochastic covering location problems: Modeling framework and solution approach.
European Journal of Operational Research, 268(2):432–449, 2018.

Introduction

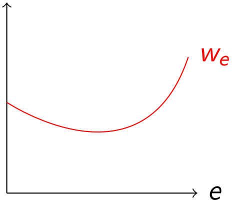
Uncertainty

- Single-facility location problem on a network.

Introduction

Uncertainty

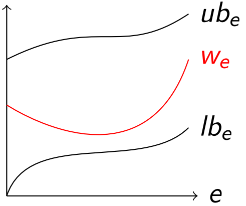
- Single-facility location problem on a network.
- The demand is distributed along the edges.



Introduction

Uncertainty

- Single-facility location problem on a network.
- The demand is distributed along the edges.
- The demand is uncertain with only a known interval estimation.



Introduction

Uncertainty

- Single-facility location problem on a network.
- The demand is distributed along the edges.
- The demand is uncertain with only a known interval estimation.
- Aim: Minimise the *worst-case* of coverage loss.

Criteria

- Coverage criterion: Maximal Covering.

Criteria

- Coverage criterion: Maximal Covering. Let $x \in G$ be a facility:
  - $z \in G$ is *covered* by x, if $d(x, z) \leq R$.
  - $C(x) := \{z \in G \mid d(x, z) \leq R\}$ is the *coverage area* of x.

Criteria

- Coverage criterion: Maximal Covering. Let $x \in G$ be a facility:
  - $z \in G$ is *covered* by x, if $d(x, z) \leq R$.
  - $C(x) := \{z \in G \mid d(x, z) \leq R\}$ is the *coverage area* of x.

  The covered demand on an edge $e \in E$ by x for a specific demand realisation w:

$$g_e(x, w) = \int_{y=(e,t)\in C_e(x)} w_e(t)\, dt. \tag{1}$$

  The total amount of covered demand on the network:

$$g(x, w) = \sum_{e \in E} g_e(x, w). \tag{2}$$

Criteria

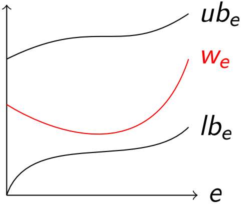
- Coverage criterion: Maximal Covering.

$$\max_{x \in G} g(x) = \sum_{e \in E} \int_{y=(e,t) \in C_e(x)} w_e(t)\, dt. \qquad (1)$$

O. Berman, J. Kalcsics, and D. Krass.
On covering location problems on networks with edge demand.
Computers & Operations Research, 74:214–227, 2016.

Criteria

- Coverage criterion: Maximal Covering.
- Uncertainty Demand: MinMax Regret. Minimise the *worst-case* of coverage loss.



$$r^* = \min_{x \in X} \max_{lb \leq w \leq ub} \left(\max_{y \in G} g(y, w) - g(x, w) \right).$$

Applications

- Locating AED, ATM, bus stops, automated parcel lockers, or bicycle parking racks in cities.
- Locating an aerosol dispenser in the air ducts of a building to disinfect the conduits.

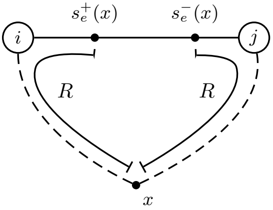
Edge coverage functions

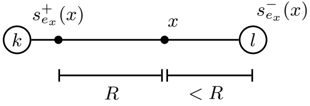
If $x \notin e = [i, j]$

$$s_e^+(x) = \min\left\{1, \max\left\{0, \frac{R - d(x,i)}{\ell_e}\right\}\right\}$$
$$s_e^-(x) = \max\left\{0, \min\left\{1, 1 - \frac{R - d(x,j)}{\ell_e}\right\}\right\}$$

If $x \in e_x = [k, l]$

$$s_{e_x}^+(x) = \max\left\{0, \frac{d(x,k) - R}{\ell_{e_x}}\right\}$$
$$s_{e_x}^-(x) = \min\left\{1, 1 - \frac{d(x,l) - R}{\ell_{e_x}}\right\}$$





Coverage area

The total coverage can now be written as

$$\begin{aligned} g(x,w) \;&= \int_{s_{e_x}^{+}(x)}^{s_{e_x}^{-}(x)} w_{e_x}(u)\,du + \sum_{e \in E^c(x)} \int_0^1 w_e(u)\,du \\ &\quad + \sum_{e \in E^p(x)} \left(\int_0^{s_e^{+}(x)} w_e(u)\,du + \int_{s_e^{-}(x)}^1 w_e(u)\,du \right) . \end{aligned}$$

Singularity points (PP)

- $PP_e \;:=\; \{i,j\} \cup NP_e \cup BP_e \cup EP_e$

  There are at most $\mathcal{O}(m)$ partition points on each $e \in E$.

- $PP \;:=\; \bigcup_{e\in E} PP_e$

  There are $\mathcal{O}(m^2)$ on the whole network.

Lemma

$s_e^+(x)$ and $s_e^-(x)$, $e \in E$, are continuous and piecewise linear functions over $x \in e_x$ with a constant number of pieces.

Edge coverage functions and singularity points

Theorem

Let $e_x \in E, x \in [z^1, z^2]$ *such that* $z^1, z^2 \in PP_{e_x}$.

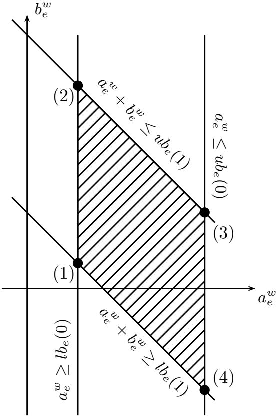
1. *The sets* $E^c(x)$, $E^u(x)$, *and* $E^p(x)$ *are identical for* $x \in [z^1, z^2]$.
2. $s_e^+(x)$ *and* $s_e^-(x)$ *have a unique linear representation for* $x \in [z^1, z^2]$.
3. $g_e(x, w)$, $e \in E$, *have a unique representation, for* $x \in [z^1, z^2]$ *and* w *a non-negative continuous demand function.*

Linear Demand realisation

Let $lb_e(t) = a_e^{lb} + b_e^{lb} \cdot t$, $ub_e(t) = a_e^{ub} + b_e^{ub} \cdot t$, and $w_e(t) = a_e^{w} + b_e^{w} \cdot t$.

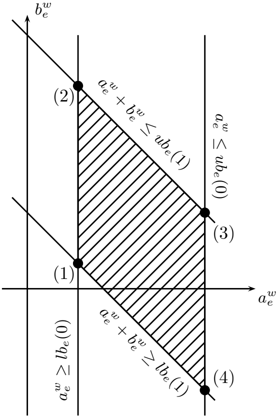
Linear Demand realisation

Let $lb_e(t) = a_e^{lb} + b_e^{lb} \cdot t$, $ub_e(t) = a_e^{ub} + b_e^{ub} \cdot t$, and $w_e(t) = a_e^w + b_e^w \cdot t$.



Linear Demand realisation

Let $lb_e(t) = a_e^{lb} + b_e^{lb} \cdot t$, $ub_e(t) = a_e^{ub} + b_e^{ub} \cdot t$, and $w_e(t) = a_e^w + b_e^w \cdot t$.



1. (a_e^{lb}, b_e^{lb}),
2. $(a_e^{lb}, a_e^{ub} + b_e^{ub} - a_e^{lb})$,
3. (a_e^{ub}, b_e^{ub}),
4. $(a_e^{ub}, a_e^{lb} + b_e^{lb} - a_e^{ub})$.

Worst-case demand realisation

Theorem

The worst-case demand realisation for a fixed x, y, and e such that $x, y \in G$ and $e \in E$ can be obtained by solving the following linear program:

$$\max_{lb_e \leq w_e \leq ub_e} a_e^w \left(c_e(y) - c_e(x)\right) + \frac{1}{2} b_e^w \left(\bar{c}_e^2(y) - \bar{c}_e^2(x)\right), \qquad (1)$$

Worst-case demand realisation

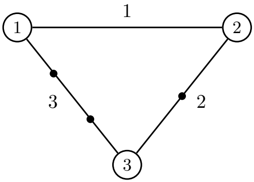
Theorem

An optimal solution of (1), (a_e^{w*}, b_e^{w*}), is given by the first column of the following table whenever the conditions of columns 2-4 are fulfilled.

(a_e^{w*}, b_e^{w*})	Conditions		
	$c_e(y) - c_e(x)$	$\bar{c}_e^2(y) - \bar{c}_e^2(x)$	$(\bar{c}_e^2(y) - \bar{c}_e^2(x)) - 2(c_e(y) - c_e(x))$
(a_e^{lb}, b_e^{lb})	≤ 0	≤ 0	≥ 0
$(a_e^{lb}, a_e^{ub} + b_e^{ub} - a_e^{lb})$	≥ 0	≥ 0	≥ 0
	≤ 0	≥ 0	$-$
(a_e^{ub}, b_e^{ub})	≥ 0	≥ 0	≤ 0
$(a_e^{ub}, a_e^{lb} + b_e^{lb} - a_e^{ub})$	≤ 0	≤ 0	≤ 0
	≥ 0	≤ 0	$-$

Algorithm

- Determine the set PP of partition points.



Let $R = 1$,

$$\begin{aligned} lb_{[1,2]}(t) &= 3 - 3t, \quad & ub_{[1,2]}(t) &= 15 + 7t, \\ lb_{[2,3]}(t) &= 3t, & ub_{[2,3]}(t) &= 7 + 3t, \\ lb_{[1,3]}(t) &= 2 + 3t, & ub_{[1,3]}(t) &= 8 + 10t. \end{aligned}$$

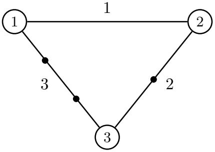
Algorithm

For $e = [i, j] \in E$:

- Sort the partition $PP_e = \{z^1, \ldots, z^{|PP_e|-1}\}$ in non-decreasing distance from node i.
- Derive the representation of the edge coverage functions over each sub-edge $[z^q, z^{q+1}]$, for $q \in \mathcal{I}_e := \{1, \ldots, |PP_e|-1\}$.

For $x_1 \in ([1,2], t_1)$ the edge coverage functions are given by

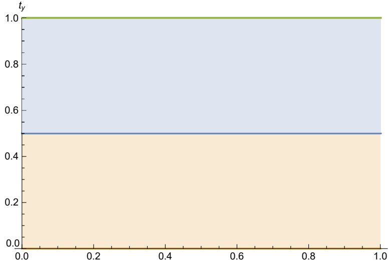
$$
\begin{aligned}
s^+_{[1,2]}(x_1) &= 0, & s^-_{[1,2]}(x_1) &= 1,\\
s^+_{[2,3]}(x_1) &= \frac{t_1}{2}, & s^-_{[2,3]}(x_1) &= 1,\\
s^+_{[1,3]}(x_1) &= \frac{1-t_1}{3}, & s^-_{[1,3]}(x_1) &= 1.
\end{aligned}
$$



Algorithm

1: **for** $e_x \in E$ **do**
2: **for** $i \in \mathcal{I}_{e_x}$ **do**
3: **for** $e_y \in E$ **do**
4: **for** $j \in \mathcal{I}_{e_y}$ **do**
5: Generate the subdivision in the rectangle $[z^i, z^{i+1}] \times [z^j, z^{j+1}]$ by the arcs defining the conditions of the worst case demand realisation for any $e \in E$.

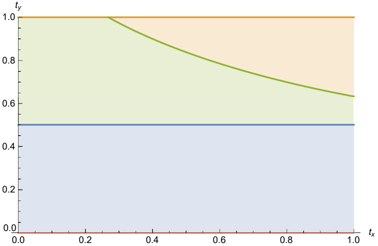
For $(x, y) \in [1, 2] \times [2, 3]$.
Cells for $r_{[1,2]}(x, y)$:



Algorithm

1: **for** $e_x \in E$ **do**
2: **for** $i \in \mathcal{I}_{e_x}$ **do**
3: **for** $e_y \in E$ **do**
4: **for** $j \in \mathcal{I}_{e_y}$ **do**
5: Generate the subdivision in the rectangle $[z^i, z^{i+1}] \times [z^j, z^{j+1}]$ by the arcs defining the conditions of the worst case demand realisation for any $e \in E$.

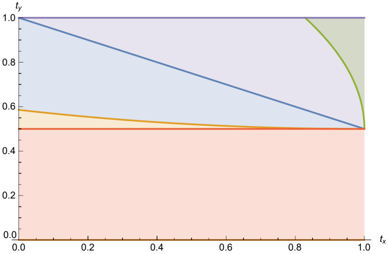
For $(x, y) \in [1, 2] \times [2, 3]$.
Cells for $r_{[2,3]}(x, y)$:



Algorithm

for $e_x \in E$ **do**
    for $i \in \mathcal{I}_{e_x}$ **do**
        for $e_y \in E$ **do**
            for $j \in \mathcal{I}_{e_y}$ **do**
                Generate the subdivision in the rectangle $[z^i, z^{i+1}] \times [z^j, z^{j+1}]$ by the arcs defining the conditions of the worst case demand realisation for any $e \in E$.

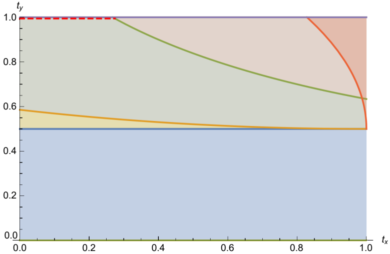
For $(x, y) \in [1, 2] \times [2, 3]$.
Cells for $r_{[1,3]}(x, y)$:



Algorithm

for $e_x \in E$ **do**
    for $i \in \mathcal{I}_{e_x}$ **do**
        for $e_y \in E$ **do**
            for $j \in \mathcal{I}_{e_y}$ **do**
                Let $\mathcal{C}^{ij}_{e_x e_y}$ be the family of arcs defined by:
                -boundaries of the cells previously obtained.
                -For any cell where $r(x,y)$ is concave, the intersection of the curve $\frac{\partial r}{\partial y}(x,y) = 0$ with that cell.

For $(x,y) \in [1,2] \times [2,3]$.
Cells for $r(x,y)$



Algorithm

```
for e_x ∈ E do
    for i ∈ I_{e_x} do
        Obtain the upper envelope, h^i_{e_x}(x), of r(x, y(x)) of the arcs
        contained in ⋃_{e_y ∈ E, j ∈ I_{e_y}} C^{ij}_{e_x e_y}.
        Find the minimum x_i^* of h^i_{e_x}(x) over [z^i, z^{i+1}].
        if h^i_{e_x}(x_i^*) < r(x^*) then
            set x^* := x_i^*, r(x^*) = h^i_{e_x}(x_i^*).
```

for $e_x \in E$ **do**

  for $i \in \mathcal{I}_{e_x}$ **do**

    Obtain the upper envelope, $h^i_{e_x}(x)$, of $r(x, y(x))$ of the arcs contained in $\bigcup_{e_y \in E, j \in \mathcal{I}_{e_y}} \mathcal{C}^{ij}_{e_x e_y}$.

    Find the minimum x_i^* of $h^i_{e_x}(x)$ over $[z^i, z^{i+1}]$.

    if $h^i_{e_x}(x_i^*) < r(x^*)$ **then**

      set $x^* := x_i^*$, $r(x^*) = h^i_{e_x}(x_i^*)$.

Algorithm

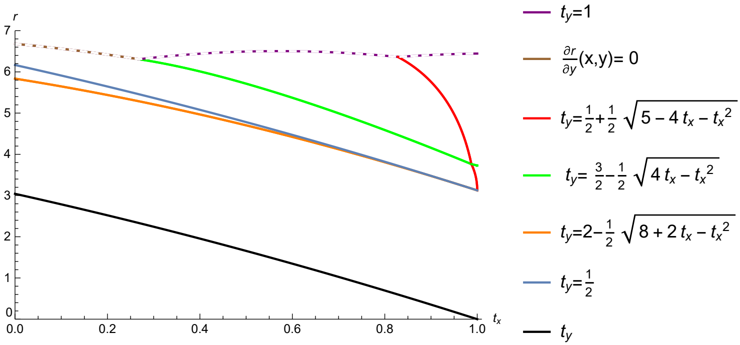


Figure: $r(x, y(x))$, $x \in [1, 2]$, and $y \in [2, 3]$.

Algorithm

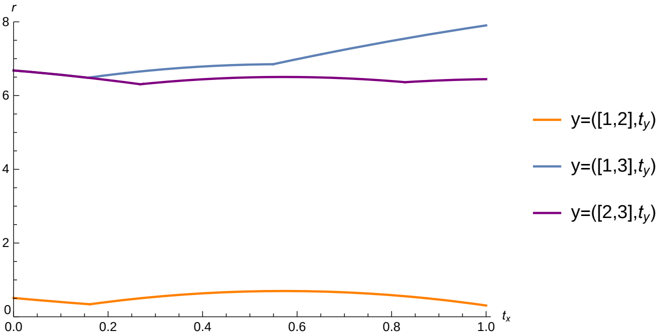


Figure: Upper envelope of $r(x, y(x))$, $x \in [1, 2]$.

The minimum value of r is 6.3055, where $x^*_{[1,3]} = ([1, 3], 0.0533)$.
This should be repeated for each $x \in [z^i, z^{i+1}]$, where $i \in \mathcal{I}_{e_x}$ and $e_x \in E$.

Algorithm

Theorem

The single facility MinMax Regret Maximal Covering Location Problem on a network with edge linear demand realisations can be solved exactly in $\mathcal{O}(m^4 \log^* m)$ *time using the previous Algorithm.*

Conclusions

- Although the majority of problems become NP-hard in the minmax regret version, we propose a polynomial time algorithm for solving the single-facility MinMax Regret MCLP on a network where the demand is
  - distributed along the edges,

Conclusions

- Although the majority of problems become NP-hard in the minmax regret version, we propose a polynomial time algorithm for solving the single-facility MinMax Regret MCLP on a network where the demand is
  - distributed along the edges,
  - constant or linear functions,

Conclusions

- Although the majority of problems become NP-hard in the minmax regret version, we propose a polynomial time algorithm for solving the single-facility MinMax Regret MCLP on a network where the demand is
  - distributed along the edges,
  - constant or linear functions,
  - uncertain with only a known interval estimation.

Conclusions

- Although the majority of problems become NP-hard in the minmax regret version, we propose a polynomial time algorithm for solving the single-facility MinMax Regret MCLP on a network where the demand is
  - distributed along the edges,
  - constant or linear functions,
  - uncertain with only a known interval estimation.

Conclusions

- Although the majority of problems become NP-hard in the minmax regret version, we propose a polynomial time algorithm for solving the single-facility MinMax Regret MCLP on a network where the demand is
  - distributed along the edges,
  - constant or linear functions,
  - uncertain with only a known interval estimation.

M. Baldomero-Naranjo, J. Kalcsics, and A. M. Rodríguez-Chía.
Minmax regret maximal covering location problems with edge demands.
Computers & Operations Research, 130:105181, 2021.

Future work

Potential avenues for future research:

- Other kind of demand realisation functions.

Future work

Potential avenues for future research:

- Other kind of demand realisation functions.
- Multi-facility location version of the problem.

Future work

Potential avenues for future research:

- Other kind of demand realisation functions.
- Multi-facility location version of the problem.
- Apply a different criterion of coverage, e.g. the gradual covering.


thank you
danke
рахмат
謝謝
ngiyabonga
teşekkür ederim
gracias
merci
obrigado
dziękuję
bedankt
спасибо
Баярлалаа
mersi
barka
welalin
tack
spas
dank je
misaotra
matondo
paldies
grazzi
mahalo
tapadh leat
хвала
asante
manana
obrigada
tenki
murakoze
faafetai lava
vinaka
спасибі
blagodaram
kiitos
dankie
nanni
nandri
dhanyavad
mauruuru
koszonom
hvala
gracie
bayarlalaa
enkosi
akun dankon ačiū
djiere dieuf
lau
mochchakkeram
дякую
mamnun
chokrane
go raibh maith agat
chnorakaloutioun
gratias ago
gracies
sulpay
taiku
sukriya
kop khun krap
sagolun
sobodi
dėkuji
mèsi
didi madloba
najis tuke
kam sah hamnida
rahmat
terima kasih
ありがとう
tanemirt
rahmet
grazie
arigatô
takk
dakujem
trugarez
dhanyavadagalu
shukriya
merce
мерси
diolch
ευχαριστώ
xiexie
감사합니다
তোমাকে ধন্যবাদ