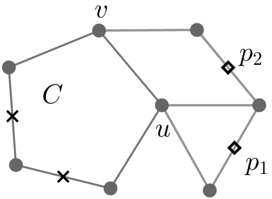
MILP models for continuous set covering on networks

Liding Xu, Mercedes Pelegrín

OptimiX, LIX, École Polytechnique

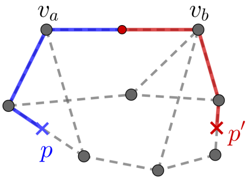
March 1, 2023

Set covering problems on networks



- Consider a network $N = (V, A)$,
- Every edge in A is a continuum.
- $C(N)$ is the union of edges A and nodes N.
- We aim to cover $C(N)$ rather than V.

Set covering problems on networks



(a) two facilities points p and p'

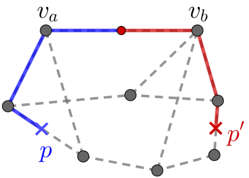
- Let $d(p_1, p_2)$ measure the shortest path distance between two points p_1 and p_2 in $C(N)$.
- Each point $p \in C(N)$ can cover the points in $C(N)$ with distance at most δ.

Extensions of set covering problems on networks

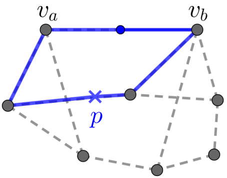
Here we use F to denote the set of facility locations, and D to denote the set of demands.

- *Discrete:* $F = D = V$, a classical set covering problem.
- *Semi-continuous:* either $F = C(N)$ or $D = C(N)$, reduced to the classical set covering problem (tractable).
- *Continuous:* both $F = D = C(N)$, the *continuous set covering on networks*.

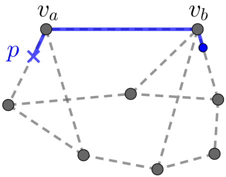
How to continuously cover an edge



(a) By two points



(b) By one point, through both ends



(c) By one point, trough one end

Figure: Covering of an edge $e = (v_a, v_b) \in E$

Existing exact approach: MILP

- The only existing MILP formulation is by Fröhlich et al., "Covering edges in network".
- Basic assumption: edge length is at most δ.
- The existing MILP assumes that every pair of edges can cover each other, so the size is large.
- MIP solvers cannot solve this MILP for moderate networks.

Our contribution

In practice, many edges are far away, and edge lengths can greater than δ.

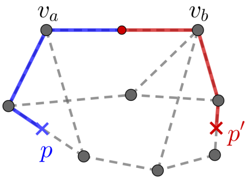
- Various preprocessing techniques: delimitation and modelling long edges.
- Two main new MILP models and some strengthening technique.
- An open-source implementation.

Modeling the facilities

We want to label/index a set of potential facilities.

- Two types of facility index: $N \cup A$.
- 0-1 facility variables $y \in \{0, 1\}$: 1 for installed, 0 for not installed.
- Node facilities: y_v associated node $v \in N$.
- Edge facilities: y_e associated edge $e \in A$, with a free location variable $q_e \in [0, \ell_e]$.
- These variables determine $P \subset C(N)$, locations of installed facilities.

Modeling the covering condition



(a) By two points

- The residual cover r_{v_a}, r_{v_b}: the truncated length of covering paths.

$$r_v := \max_{p \in P}(d(p, v) - \delta, 0).$$

- The edge $e = (v_a, v_b)$ is covered if $r_{v_a} + r_{v_b} \geq \ell_e$ (the length of e).

Modeling the covering condition

- $$\max_{p \in P}(d(p, v_a) - \delta, 0) + \max_{p \in P}(d(p, v_b) - \delta, 0) \geq \ell_e$$

  is a concave constraint.
- $$\max_{p \in P}(d(p, v_a) - \delta, 0), \max_{p \in P}(d(p, v_b) - \delta, 0)$$

  are piece-wise linear functions.
- big-M technique (with extra binary variables) models $\max, \operatorname{argmax}$ of piece-wise linear functions.
- Inner representation: $d(p, v_a), d(p, v_b)$ are linear constraints on y, q.

Preprocessing: delimitation

A facility in an edge can only cover its neighbors locally (distance at most δ)!

- delimitation: a concrete characterization (covers) of δ-neighbors for edges and facilities.
- `potential covers`, `complete covers`, and `partial covers`.
- Simple covering condition:

$$\max_{p \in P(e)} (d(p, v_a) - \delta, 0) + \max_{p \in P(e)} (d(p, v_b) - \delta, 0) \geq \ell_e,$$

  with $P(e) \subseteq P$.
- Less binary variables, smaller big-M.

MILP Model 1

$$
\begin{aligned}
\min \sum_{f\in\mathcal{F}} y_f & & (6a)\\
\text{s.t. } w_e \geq y_f & \quad e\in E, f\in\mathcal{F}_c(e) & (6b)\\
w_e \leq \sum_{f\in\mathcal{F}_c(e)} y_f & \quad e\in E & (6c)\\
x_v \geq 1 - \sum_{e\in E(v)} (1-w_e) & \quad v\in V & (6d)\\
x_v \leq w_e & \quad v\in V, e\in E(v) & (6e)\\
y_{v'_{i'}} + y_{e'} \leq 1 & \quad e'\in E, i'\in\{a,b\} & (6f)\\
q_{e'} \leq l_{e'} y_{e'} & \quad e'\in E & (6g)\\
l_e(1-w_e) \leq r_{v_a} + r_{v_b} & \quad e\in E & (6h)\\
x_v + \sum_{v'\in\mathcal{V}_p(v)} z_{vv'} + \sum_{(e',i')\in\mathcal{EI}_p(v)} z_{ve'i'} = 1 & \quad v\in V & (6i)\\
z_{vv'} \leq y_{v'} & \quad v\in V, v'\in\mathcal{V}_p(v) & (6j)\\
z_{ve'i'} \leq y_{e'} & \quad v\in V, (e',i')\in\mathcal{EI}_p(v) & (6k)\\
r_v \leq M_v(1-x_v) & \quad v\in V & (6l)\\
r_v \leq M_{vv'}(1-z_{vv'}) + \delta - d(v,v') & \quad v\in V, v'\in\mathcal{V}_p(v) & (6m)\\
r_v \leq M_{ve'i'}(1-z_{ve'i'}) + \delta - \tau_{ve'i'}(q_{e'}) & \quad v\in V, (e',i')\in\mathcal{EI}_p(v) & (6n)\\
y_f, w_e \in \{0,1\} & \quad f\in\mathcal{F}, e\in E & (6o)\\
x_v, z_{vv'}, z_{ve'i'} \in \{0,1\} & \quad v\in V, v'\in\mathcal{V}_p(v), (e',i')\in\mathcal{EI}_p(v) & (6p)\\
q_{e'}, r_v \geq 0 & \quad e'\in E, v\in V. & (6q)
\end{aligned}
$$

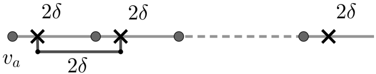
Preprocessing: long edge modeling

The previous modeling assumes short edges: $l_e \leq \delta$. Large edges?

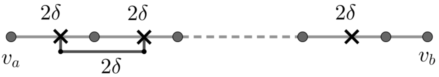
- First approach: subdivide long edges into small edges.
- Second approach: directly model the covering condition on long edge.

Preprocessing: long edge modeling

(a) A facility is located at v_a ($q_e = 0$)



(b) A facility is located at the tail ($0 < q_e \leq \hat{l}_e$)

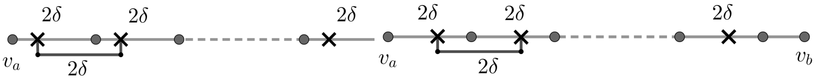


(c) No facility is located at the tail ($\hat{l}_e < q_e \leq 2\delta$)

Figure: Covering a long edge $e = (v_a, v_b)$

#facilities in Figures 4a and 4b = #facilities in Figure 4c +1. An indicator variable can model tansition of two states.

MILP model 2



- The location of the left facility determines the other facility locations (2δ).
- Scalability (invariant to edge length): need only one indicator variable for state transition and a location variable for the left facility.
- Modification of MILP model 1: add specific variables and constraints for long edges, and other parts for small edges remain the same.

Algorithmic tool: CFLG.jl

- Implementation is based on JuMP and written in Julia.
- Input: a network and a cover radius δ.
- Output: the number of facilities and locations.
- Algorithmic options:
  - EF: Covering edges in network by Fröhlich.
  - F0/F: MILP model 1 without/with delimitation.
  - SF: MILP model 1 with delimitation and some valid inequalities.
  - RF: MILP model 2.

Performance metric

- the relative dual gap is defined as:

$$\sigma := \frac{\overline{v} - \underline{v}}{\overline{v}},$$

  where $\overline{v}$ is an upper-bound and $\underline{v}$ is a lower-bound.

- the relative primal bound

$$v_r := \frac{\overline{v}}{n_{sd}},$$

- t: the total running time in CPU seconds.
- S/A/T: the number of solved instances/ the number of affected instances/ the number of total instances in the benchmark.

Experimental results I

Benchmark	Radius	EF				F0			
		time	$\sigma(\%)$	$v_r(\%)$	S/A/T	time	$\sigma(\%)$	$v_r(\%)$	S/A/T
city	Small	1800.0	100.0%	100.0%	0/0/9	1801.7	56.8%	83.3%	0/3/9
	Large	1800.0	100.0%	100.0%	0/0/9	1800.9	42.3%	36.2%	0/6/9
Kgroup_A	Small	1800.0	100.0%	100.0%	0/0/11	1802.6	25.1%	85.0%	0/11/11
	Large	1800.0	100.0%	100.0%	0/0/11	139.2	14.7%	19.2%	7/11/11
Kgroup_B	Small	1800.0	100.0%	100.0%	0/0/12	1800.4	92.6%	98.8%	0/1/12
	Large	1800.0	100.0%	100.0%	0/0/12	1800.1	93.2%	86.6%	0/1/12
random_A	Small	1800.0	100.0%	100.0%	0/0/12	16.8	15.9%	54.8%	9/12/12
	Large	1800.0	100.0%	100.0%	0/0/12	0.2	25.5%	19.5%	12/12/12
random_B	Small	1800.0	100.0%	100.0%	0/0/12	1317.6	36.4%	63.3%	1/12/12
	Large	1800.0	100.0%	100.0%	0/0/12	154.4	26.0%	10.0%	11/12/12
all	Small	1800.0	100.0%	100.0%	0/0/56	625.8	37.4%	74.8%	10/39/56
	Large	1800.0	100.0%	100.0%	0/0/56	132.5	33.1%	25.9%	30/42/56

Table: Results for continuous models

Experimental results II

Benchmark	Radius	F				SF			
		time	σ(%)	v_r(%)	S/A/T	time	σ(%)	v_r(%)	S/A/T
city	Small	1802.9	29.5%	62.2%	0/9/9	1801.3	30.1%	66.9%	0/9/9
	Large	1801.2	28.4%	21.7%	0/9/9	1800.9	29.1%	21.7%	0/9/9
Kgroup_A	Small	1803.0	33.1%	82.2%	0/11/11	1801.3	32.0%	80.6%	0/11/11
	Large	238.0	18.9%	19.1%	8/11/11	300.8	19.0%	19.1%	8/11/11
Kgroup_B	Small	1800.6	80.8%	240.5%	0/12/12	1801.4	79.7%	191.9%	0/12/12
	Large	1800.4	85.1%	80.5%	0/12/12	1800.7	85.9%	77.3%	0/12/12
random_A	Small	20.2	16.5%	54.3%	9/12/12	16.1	17.1%	54.9%	9/12/12
	Large	0.3	25.5%	19.5%	12/12/12	0.2	10.4%	17.9%	12/12/12
random_B	Small	1574.2	38.8%	64.9%	1/12/12	1501.2	40.0%	67.5%	1/12/12
	Large	220.5	19.9%	10.3%	9/12/12	175.7	18.8%	10.0%	11/12/12
all	Small	675.0	35.2%	86.2%	10/56/56	637.6	35.5%	83.6%	10/56/56
	Large	163.0	30.2%	23.6%	29/56/56	160.9	24.9%	22.8%	31/56/56

Table: Results for continuous models

Experimental results III

Benchmark	Radius	RF			
		time	$\sigma(\%)$	$v_r(\%)$	S/A/T
city	Small	1804.4	16.2%	54.1%	0/9/9
	Large	1801.5	25.8%	21.3%	0/9/9
Kgroup_A	Small	1622.6	21.5%	77.5%	1/11/11
	Large	158.9	19.2%	19.3%	8/11/11
Kgroup_B	Small	1800.9	59.1%	154.2%	0/12/12
	Large	1800.6	75.5%	63.3%	0/12/12
random_A	Small	15.9	8.1%	54.3%	9/12/12
	Large	0.3	26.6%	19.8%	12/12/12
random_B	Small	1304.3	38.5%	63.8%	1/12/12
	Large	190.2	19.8%	11.2%	9/12/12
all	Small	604.9	23.7%	75.4%	11/56/56
	Large	146.6	29.2%	22.8%	29/56/56

Table: Results for continuous models

Conclusion

- New preprocssing and MILP models for continuous set-covering on networks.
- An open source implementation: https://github.com/lidingxu/cflg .
- Pelegrín, Mercedes, and Liding Xu. "Continuous covering on networks: Improved mixed integer programming formulations." Omega (2023): 102835.