# An algorithmic toolkit for continuous set covering on networks

Liding Xu, Mercedes Pelegrín

OptimiX, LIX, École Polytechnique

July 7, 2022

# Table of Contents


Introduction

- Problem description
- Existing approaches

Models

- Our constribution

Experiments

Conclusion

# Set covering problem

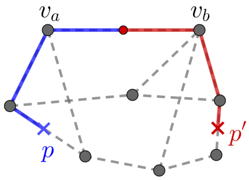
A classical NP-complete problem: Given a universal $U$ and a family $S$ of subsets of $U$, the problem asks the minimal number of subsets covering $U$.

$$\min_{s \in S} x_s \tag{1}$$

$$\sum_{e \in S} x_s \geq 1, \quad e \in U \tag{2}$$

$$x_s \in \{0, 1\}, \quad e \in S. \tag{3}$$

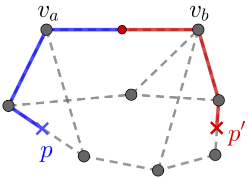
# Set covering problems on networks




(a) two facilities points

- Consider a network $N = (V, A)$,
- Assume every edge is continuous, and its continuum is the union of points.
- The continuum of $N$ is $C(N)$.

# 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 $\delta$.

# Extensions of set covering problems on networks

Here we use $S$ to denote the set of facility locations, and $U$ to denote the set of demands.

- *Discrete:* when $U = V$ and $S = V$, reduced to the classical set covering problem (ILP or approximation algorithm).
- *Semi-continuous:* when either $U = C(N)$ or $S = C(N)$, the problem is reducable to the classical set covering problem (tractable).
- *Continuous:* When both $U = C(N)$ and $S = C(N)$, the *continuous set covering on networks*.

# Continuous set covering on networks

Some applications:

- locations of ambulance bases.
- surveillance cameras.
- routing servers in a network of computers.
- cranes for construction.
- aerial military medical evacuation facilities.
- aircraft alert sites for homeland defense.
- eVTOL safety landing sites.

# Existing exact approach: discretization

- *discretization methods:* preprocsing procedures to reduce the problem to a tractable set covering problem.
- *finite dominating sets (FDS):* finite subsets of candidate locations guaranteed to contain an optimal solution.

# Existing exact approach: discretization

An example: all edges have unit length and $\delta = 2$.

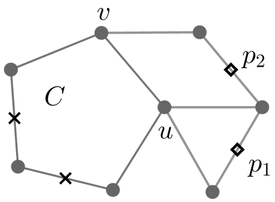



Figure: Two cycle coverage points with respect to a cycle $C$ of five nodes

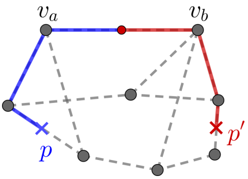
FDS: nodes and mid-points of edges.

# Existing exact approach: discretization

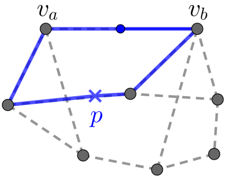
- *reduced problem:* semi-continuous, then further reduced to a discrete version.
- However, discretization methods rely on assumptions, e.g., edge lengths are natural numbers.

# Existing exact approach: MILP

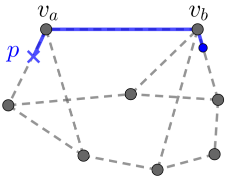
Covering conditions, an example:




(a) By two points




(b) By one point, through both ends




(c) By one point, trough one end

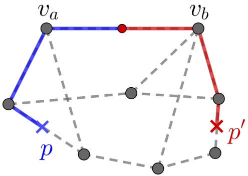
Figure: Covering of an edge $e = (v_a, v_b) \in E$

# 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 $\delta$.
- MIP solvers cannot solve this MILP for moderate networks.

# Existing exact approach: MILP

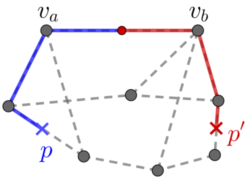
Basi ideas.




(a) By two points

- each edge can host a facility.
- an edge is covered, if the sum of available “cover range” from the left-end and the right-end is greater than the edge length.
- modeling the cover range between each pair of edges.

# Existing exact approach: MILP

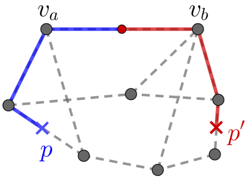



(a) By two points

Some comments:

- Each pair of edges are modeled, as in a complete graph. Networks are usually sparse!
- Some edges or nodes cannot contribute to cover, if the distance is large.
- Symmetry: if a facility is at a node, which incident edge host this node?

# New model of the covering condition



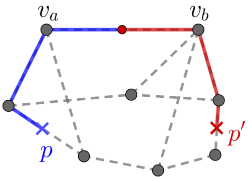

(a) By two points

- two kinds of facilities: facilities at nodes and facilities in edges.
- a point is covered by a covering path from a facility, the path length is at most $\delta$.

# Our contribution

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

# Modeling the covering condition




(a) By two points

- the residual cover: given a node, the truncated length of covering paths.
- an edge is covered if the sum of residual cover from the left end and the right end is greater than the edge length.

# MILP Model 1

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

# Preprocessing: delimitation

the reduction of the candidate space:

- `potential covers`: a set of edges and nodes in which a facility can possibly cover a given node.
- `complete covers`: a set of edges and nodes in which a facility can *always* cover a given edge.
- `partial covers`: further refinement of the potential covers.

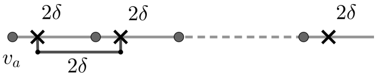
# Preprocessing: long edge modeling

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

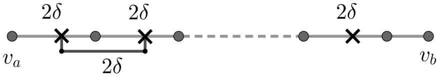
- First approach: subdivide long edges into small edges.
- Second approach: directly model the covering condition on long edge.

# 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 $e = (v_a, v_b)$

Key observation: once the location of the left-most facility is determined, other facility locations are determined.

# MILP model 2

Modify MILP model 1 for covering on long edges. We add specific variables and constraints, and other parts remain the same.

# Algorithmic tool: CFLG.jl

- Implementation is based on JuMP and written in Julia.
- Input: a network and a cover radius $\delta$.
- Output: the number of facilities and locations.
- Algorithmic options:
  - EF: Covering edges in network.
  - F0: MILP model 1 without delimitation.
  - F: MILP model 1 with delimitation.
  - SF: MILP model 1 with delimitation and some valid inequalities.
  - RF: MILP model 2.
  - SFD: semi-continuous model, SF with facilities located at nodes.

# Data

- `Kgroup:` It consists of 23 prize-collecting Steiner tree problem instances, designed to have a local structure somewhat similar to street maps.
- `City:` It consists of real data of 9 street networks for some German cities.
- `Random:` It consists of 24 random networks instances generated by the package "Networkx"

# 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 | $\sigma$(%) | $v_r$(%) | S/A/T | time | $\sigma$(%) | $v_r$(%) | 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

# More details

Code: https://github.com/lidingxu/cflg
Arxiv: Mercedes Pelegrín and Liding Xu, "Continuous Covering on Networks: Strong Mixed Integer Programming Formulations"

# Conclusion

- New preprocssing and MILP models for continuous set-covering on networks.
- An open source implementation.