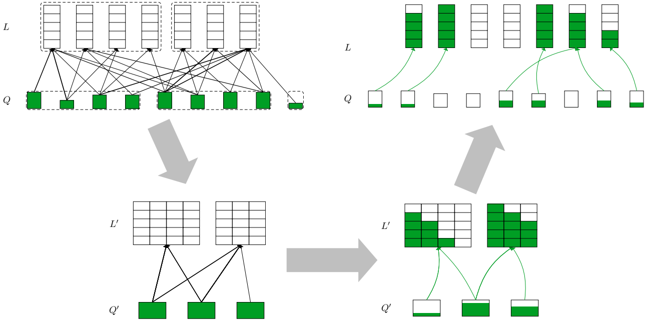
A Learning Bilevel Optimization Approach for the Demand Maximizing Battery Swapping Station Location Problem

Laurenz Tomandl Thomas Jatschka Günther Raidl
Tobias Rodemann

TU WIEN
ALGORITHMS AND COMPLEXITY GROUP

Bilevel Optimization (BLO) Approach

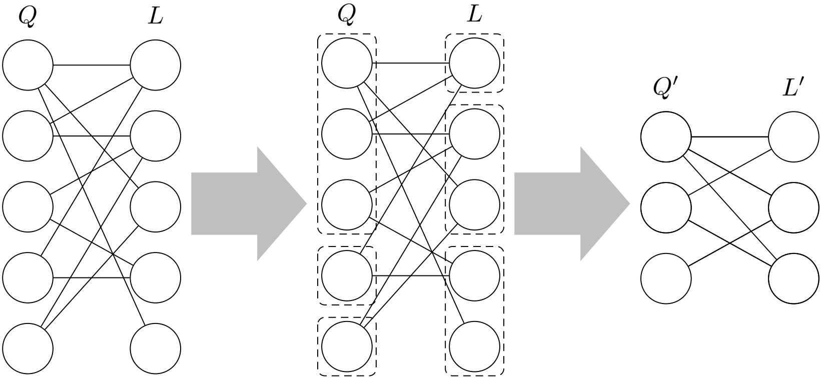
1. coarsening: reduce problem size
2. solve coarsest problem
3. projection: project solution to less coarse graphs



Bilevel Optimization (BLO) Approach

Coarsening

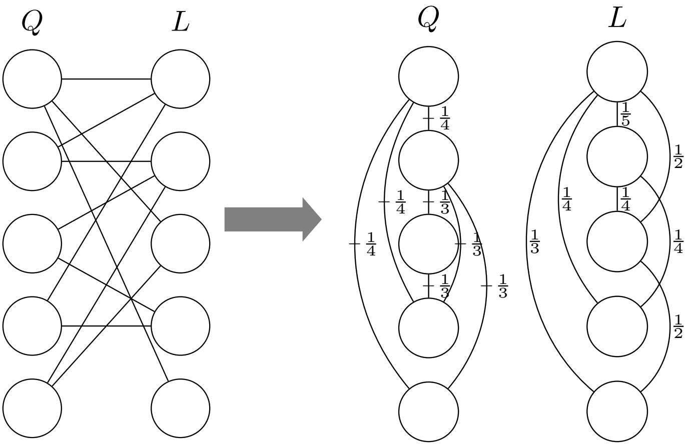
- partitioning: partition nodes of G
- contracting: aggregate nodes in each partition $\rightarrow G'$
- vertex Sets (Q, L) partitioned separately of each other



Learning Bilevel Optimization (LBLO) Approach

Similarity Calculation

- learned model: $S(v, v') = \mathrm{model}(v, v')\ldots$ machine learning similarity for L set
- heuristic approach: $S(v, v') = -\frac{|N(v) \cap N(v')|}{|N(v) \cup N(v')|}\ldots$ Jaccard similarity for Q set



Partitioning Strategies

- **k-medoids** (PAM)
  Greedy construction algorithm with 1-swap local search
- **minimum spanning tree clustering**
  Clustering by creating an mst and deleting edges of connected components until a desired size is reached
- **agglomerative clustering** with different **linkage methods**.
  clustering by partitioning the most promising nodes together
  - **minimum linkage**: use minimum weight of partitioned nodes
  - **maximum linkage**: use maximum weight of partitioned nodes
  - **mean linkage**: use mean weight of partitioned nodes
  - **dominant linkage**: use weight of the dominant node in the partitioning
- **Multi Level**: only allow pairs of nodes in a cluster. Cluster over Multiple Levels

Related Work

- Jatschka et al. (2023), EvoCOP23: Multilevel Optimization for MBSSLP
- Jatschka et al. (2020), OPTIMA20: Large Neighborhood Search for MBSSLP
- Walshaw (2002), Operations Research: Multilevel Optimization for traveling salesman problem
- Valejo et al. (2020), ACM Computing Surveys: Overview of MLO approaches

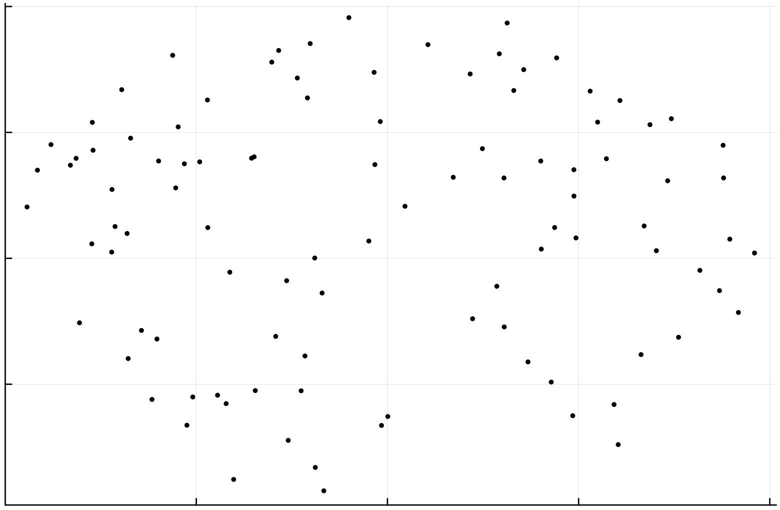
Problem Motivation

- adoption of electric vehicles steadily increasing
- major hindrance for customers: long charging times
- alternative approach: battery swapping stations
  - users replace depleted batteries with fully charged ones
  - depleted batteries are recharged and provided again later
- **Goal:** optimal setup of battery swapping station

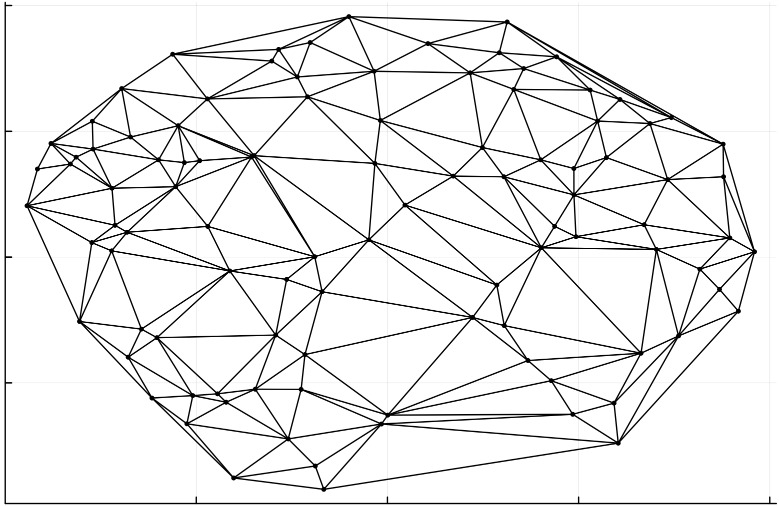
  → The Demand Maximizing Battery Swapping Station Location Problem (DMBSSLP)

© Gogoro

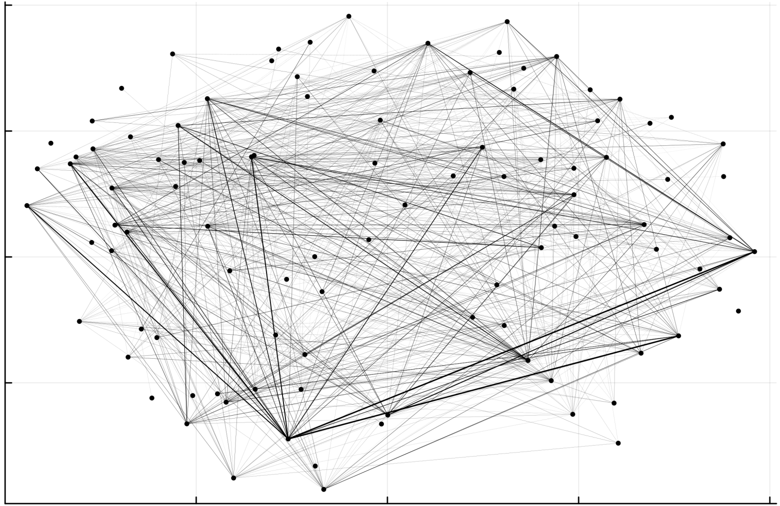
Instance and Solution Representation



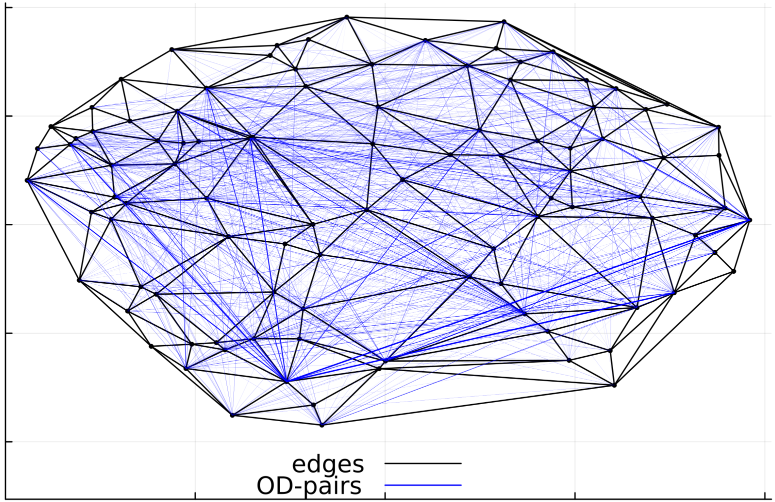
Instance and Solution Representation



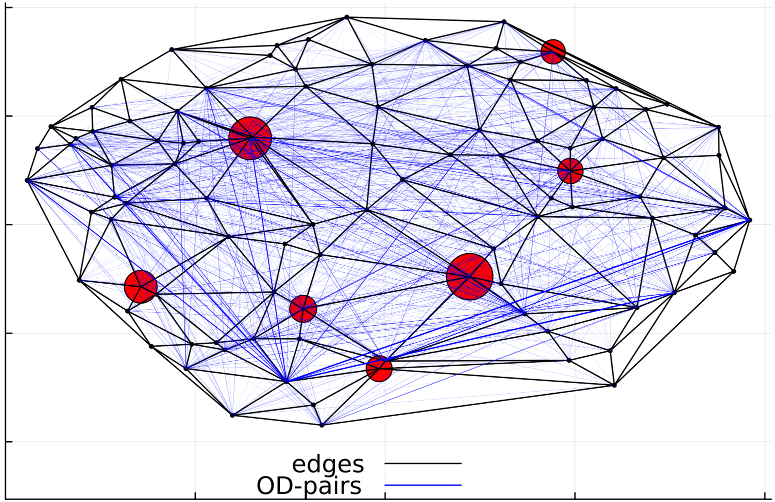
Instance and Solution Representation



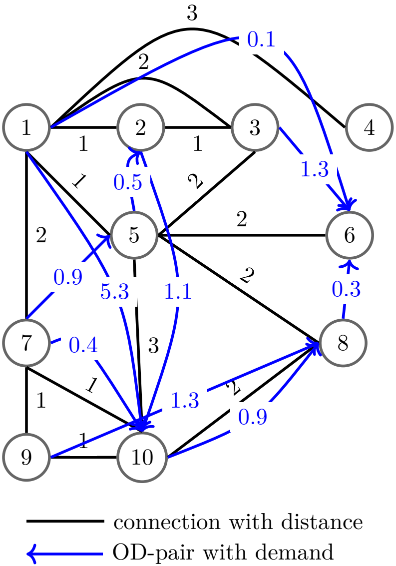
Instance and Solution Representation



Instance and Solution Representation



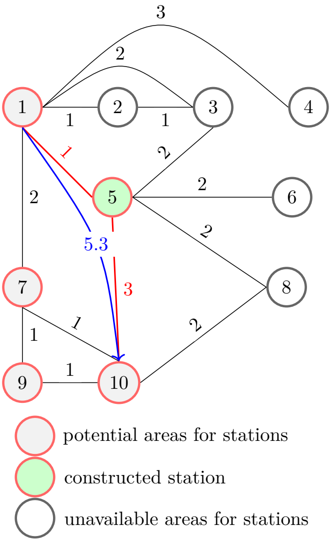
The Demand Maximizing Battery Swapping Station Location Problem (DMBSSLP)



- edge weights are distances
- nodes are possible areas for a station (#nodes = n)
- origin destination pairs (OD-pairs) constitute demand (#OD-pairs = m)
- detours impact user acceptance

Goal: decide area and capacity of a station in order to maximize satisfied demand

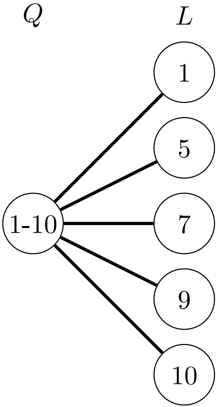
The Demand Maximizing Battery Swapping Station Location Problem (DMBSSLP)



- edge weights are distances
- nodes are possible areas for a station (#nodes = n)
- origin destination pairs (OD-pairs) constitute demand (#OD-pairs = m)
- detours impact user acceptance

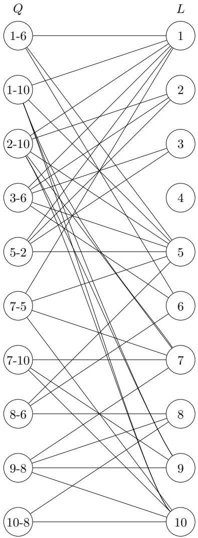
Goal: decide where to set up stations with which capacity in order to maximize satisfied demand

The Demand Maximizing Battery Swapping Station Location Problem (DMBSSLP)



- reformulation on a bipartite graph
- node set Q: OD-pairs
- node set L: areas for stations
- OD-pairs are connected to areas if a station in that area may satisfy part of the demand

The Demand Maximizing Battery Swapping Station Location Problem (DMBSSLP)



- reformulation on a bipartite graph
- node set Q: OD-pairs
- node set L: areas for stations
- OD-pairs are connected to areas if a station in that area may satisfy part of the demand

Properties of DMBSSLP

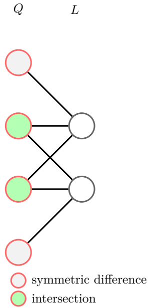
Bipartite Graph Properties

- properties of Q set
  - $\mathbf{d_q}$: demand of an OD-pair
- properties of L set
  - $\mathbf{r_l}$: number of allowed stations at area l
  - $\bar{d}_l$: The maximum demand that can be satisfied at an area l
- properties of an edge
  - e_{ql}: maximum demand that can be assigned form OD-pair q to area l
  - $\mathbf{g_{ql}}$: customer loss factor when satisfying demand of OD-pair q at area l
- global properties
  - $\mathbf{s}$: number of battery charging slots that can be built at a station
  - $\mathbf{c}$: cost of a station
  - $\mathbf{b}$: cost of a battery charging slot
  - $\mathbf{B}$: budget for the project

Model

Features

- some important features for the L neural network:
  - Δd_q: the sum of demands of OD-pairs in the symmetric difference of sets of neighboring OD-pairs
  - $\cap d_q$: the sum of demands of OD-pairs in the intersection of sets of neighboring OD-pairs
  - $\bar{d}_{l_1}, \bar{d}_{l_2}, \bar{d}_{l_3}$: the satisfiable demand at an area for the grouped nodes and the prospective merged node
  - $\frac{\sum_{q \in N(l_1)} e_{ql_1} g_{ql_1}}{\sum_{q \in N(l_1)} e_{ql_1}}, \frac{\sum_{q \in N(l_2)} e_{ql_2} g_{ql_2}}{\sum_{q \in N(l_2)} e_{ql_2}}, \frac{\sum_{q \in N(l_3)} e_{ql_3} g_{ql_3}}{\sum_{q \in N(l_3)} e_{ql_3}}$ demand weighted by the customer loss



Model

Data Collection

the trainingset contains 3000 instances of varying sizes;
1000 instances with $m = n = 1600$;
1000 instances with $m = n = 3200$;
1000 instances with $m = n = 6400$

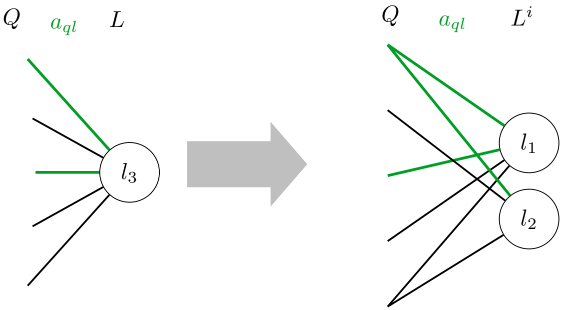
- coarsen instance by randomly merging neighboring L nodes.
- solve smaller instance using MILP
- project MILP solutions of small instance
- record features of merged nodes
- create dependent variable

Model

dependent variable

$$z_{l_1 l_2} = \left(\sum_{q \in N(l_3)} a_{ql_3}\right) - \left(\sum_{q \in N(l_3)} \tilde{a}_{ql_1} + \tilde{a}_{ql_2}\right)$$

- record difference between coarse and projected solution for individual nodes



Model
Model Parameters

Fully connected MLP
ADAM Optimizer
MSE loss with symlog transformation

$$\mathrm{symlog}(x) := \mathrm{sign}(x) \log\left(|x| + 1\right)$$

- **architecture:**
  - **width:** [40,20]; [40,40]; [80,40]; **[80,80]**; [120,80]
  - **regularization:** dropout, **no dropout**
- **learning rate:** 0.001; 0.0005; **0.0001**
- **batch size:** 64; **256**
- **number of trainings instances:** **90**; 900; 7200

Experiments

Methodology

- 9 set of 30 benchmark instances with size of $n = m = \{100, 200, 400, 800, 1600, 3200, 6400, 12800, 25600\}$
- Programming Language: Julia 1.10
- AMD EPYC 7402, 2.80GHz
- Gurobi (single threaded) 10.0 for all MI(LP) solving

Experiments

Computational Results

Average solution quality using ML-similarity or Jaccard similarity on benchmark instances

size	ML	Jaccard	ML time	Jaccard time
100	0.9274	**0.9342**	1.38s	**1.15s**
200	**0.9211**	0.9127	2.39s	**1.79s**
400	**0.9000**	0.8750	4.67s	**3.03s**
800	**0.8860**	0.8571	13.1s	**9.45s**
1600	**0.8825**	0.8614	37.5s	**16.4s**
3200	**0.8881**	0.8593	81.2s	**74.8s**
6400	**0.8817**	0.8431	331s	**268s**
12800	**0.8809**	0.8435	1240s	**1120s**
25600	**0.8714**	0.8454	**6220s**	8060s

Average solution quality using ML-similarity or Jaccard similarity on benchmark instances grouped by different partitioning algorithms

	ML	Jaccard
mst-clustering	**0.9005**	0.8749
k-medoids	**0.9019**	0.8739
multi-level	0.8641	0.8439
max-linkage	0.8964	0.871
min-linkage	**0.9005**	0.876
mean-linkage	0.8954	0.876
dominant-linkage	0.8981	0.8756

Detailed results selected algorithms solution quality

	ML			Jaccard		
	mst-clustering	k-medoids	min-linkage	mst-clustering	k-medoids	min-linkage
100	0.9321	0.9304	**0.9352**	**0.9443**	0.9425	0.9420
200	0.9272	**0.9304**	0.9261	0.9227	0.9145	**0.9227**
400	0.9045	**0.9104**	0.9046	0.8765	**0.8812**	0.8802
800	0.8893	**0.8930**	0.8895	**0.8582**	0.8582	0.8613
1600	0.8956	**0.8977**	0.8916	**0.8625**	0.8579	0.8606
3200	0.8958	0.8924	**0.8986**	**0.8680**	0.8628	0.8682
6400	0.8886	0.8891	**0.8927**	0.8450	**0.8472**	0.8468
12800	0.8838	0.8896	**0.8897**	0.8487	0.8500	**0.8510**
25600	**0.8877**	0.8841	0.8773	0.8480	**0.8512**	0.8508

Detailed results selected algorithm times

	ML			Jaccard		
	mst-clustering	k-medoids	min-linkage	mst-clustering	k-medoids	min-linkage
100	**1.30s**	1.37s	1.41s	**1.19s**	1.19s	1.19s
200	**1.98s**	2.16s	2.36s	1.55s	**1.51s**	1.88s
400	**3.47s**	3.77s	4.50s	**2.40s**	2.42s	3.23
800	**10.1s**	10.6s	12.8s	6.51s	**6.41s**	9.81s
1600	**29.2s**	29.3s	36.5s	12.5s	**12.5s**	16.4s
3200	72.2s	**64.7s**	77.0s	**64.1s**	65.5s	76.1s
6400	290s	**281s**	297s	**231s**	277s	270s
12800	**1070s**	1130s	1150s	1060s	**1030s**	1150s
25600	**4640s**	5950s	6470s	**7370s**	8640s	7960

Conclusion and Future Work

- Machine Learning supported partitioning of BLO for DMBSSLP instances improves solution quality in comparison to simple heuristic by up to 4%
- Different clustering methods don't lead to significantly different solution qualities. Faster methods like mst-clustering are therefore preferable.

- show that this principle is applicable to other more complex problems
- expand the usage of ML-similarity to both Q and L set at the same time
- apply graph neural networks, with message passing

Bibliography

T. Jatschka, F. F. Oberweger, T. Rodemann, and G. R. Raidl. Distributing battery swapping stations for electric scooters in an urban area. In N. Olenev, Y. Evtushenko, M. Khachay, and V. Malkova, editors, *Optimization and Applications, Proceedings of OPTIMA 2020 – XI International Conference Optimization and Applications*, volume 12422 of *LNCS*, pages 150–165. Springer, 2020.

T. Jatschka, T. Rodemann, and G. R. Raidl. A multilevel optimization approach for large scale battery exchange station location planning. In L. Pérez Cáceres and T. Stützle, editors, *Evolutionary Computation in Combinatorial Optimization*, volume 13987 of *LNCS*, page 50–65. Springer, 2023. URL https://www.ac.tuwien.ac.at/files/pub/jatschka_23.pdf.

A. Valejo, V. Ferreira, R. Fabbri, M. C. F. d. Oliveira, and A. d. A. Lopes. A critical survey of the multilevel method in complex networks. *ACM Computing Surveys*, 53(2), 2020. ISSN 0360-0300.

C. Walshaw. A multilevel approach to the travelling salesman problem. *Oper. Res.*, 50(5):862–877, 2002.

Properties of DMBSSLP

Solution and Objective Function

decision variables for a solution:

- $\boldsymbol{x} = (x_l)_{l \in L} \in \{0, \ldots, r_l\}$ # of opened stations in l
- $\boldsymbol{y} = (y_l)_{l \in L} \in \{0, \ldots, s \cdot x_l\}$ # of battery slots in l
- $\boldsymbol{a} = (a_{ql})_{q \in Q, l \in L}$ s.t. $0 \leq a_{ql} \leq e_{ql}$: the part of the demand of OD-pair q assigned to stations at area l

objective function (maximize fulfilled demand):

$$\max \sum_{q \in Q} \sum_{l \in N(q)} a_{ql}$$

DMBSSLP MILP formulation

$$\max \sum_{q \in Q} \sum_{l \in N(q)} a_{ql}$$

$$sx_l \geq y_l \qquad l \in L$$

$$\sum_{l \in N(q)} a_{ql} \leq d_q \qquad q \in Q$$

$$\sum_{q \in N(l)} a_{ql} \leq y_l \qquad l \in L$$

$$\sum_{l \in L} (c_l x_l + b_l y_l) \leq B$$

$$x_l \in \{0, \ldots, r_l\} \qquad l \in L$$

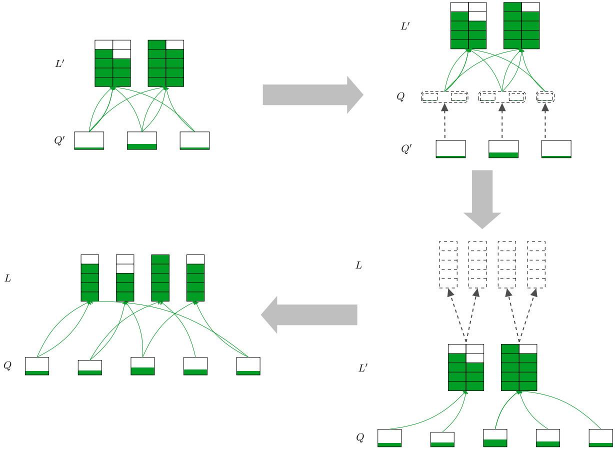
$$y_l \in \{0, \ldots, \lceil \bar{d}_l \rceil\} \qquad l \in L$$

$$0 \leq a_{ql} \leq e_{ql} \qquad q \in Q, \; l \in N(q)$$

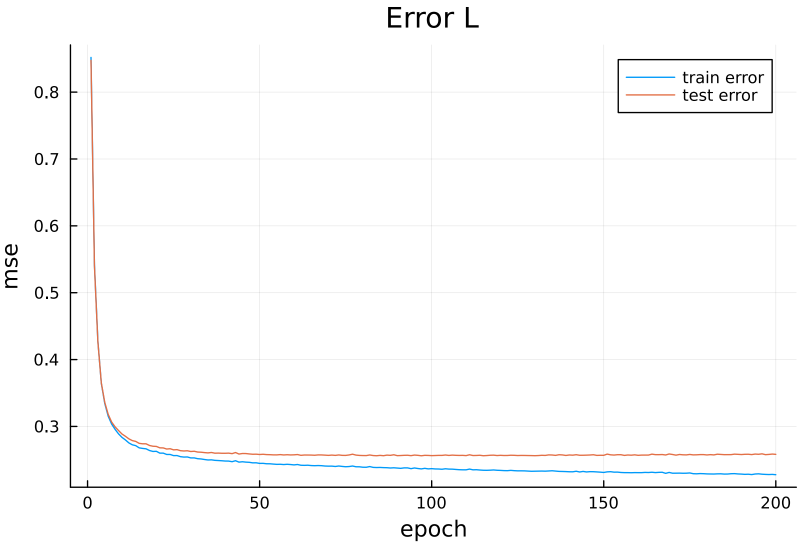
Bilevel Optimization (MLO) Approach

Projecting Solution

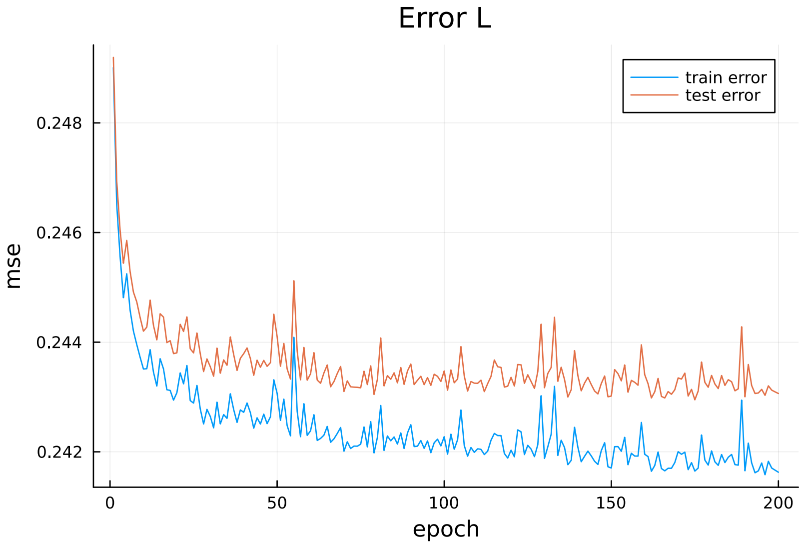
- vertex Sets (Q', L') projected independent of each other
- for each OD-pair and for each area a MI(LP) is solved
- MILP for area l are dependent of each other



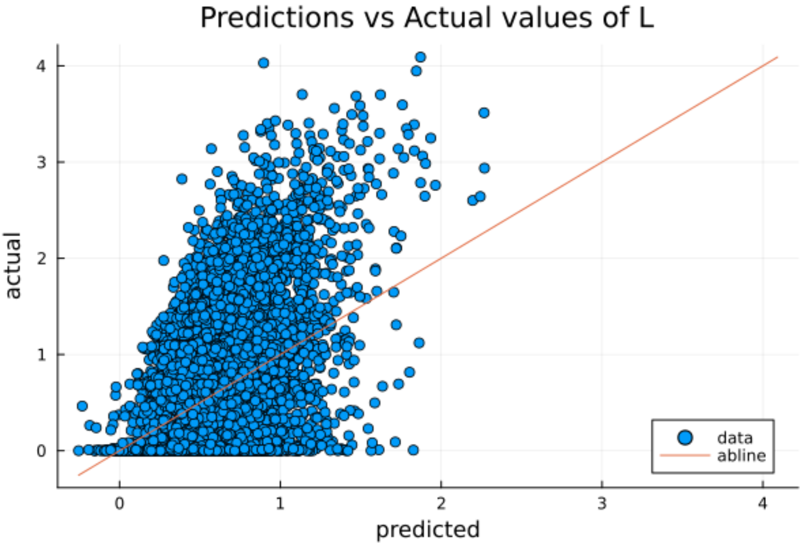
Further Results



Further Results



Further Results

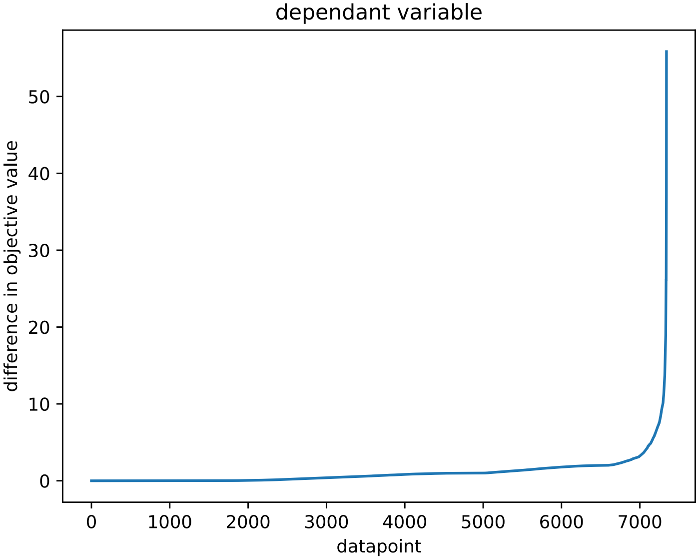


Further Results

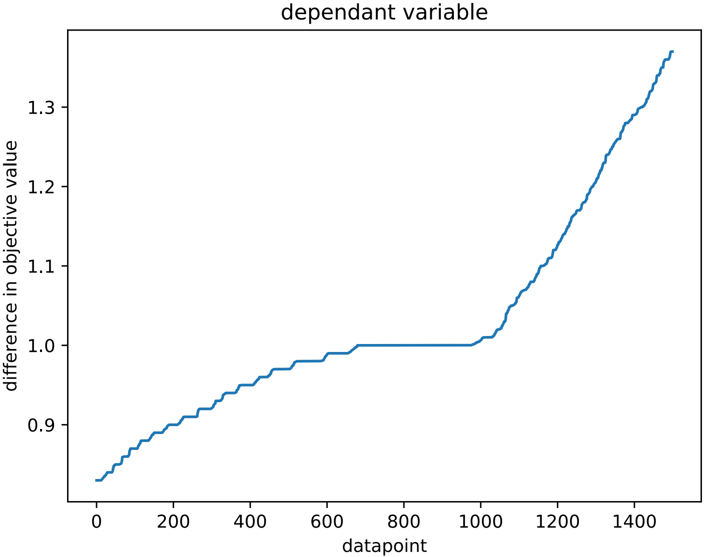
Algorithm 5.2: Partitioning

```
Input : a set of nodes U
Output: a partitioning M(U)
S ←empty max heap;
for u ∈ U do
    for v ∈ neighborhood(u)  s.t.  (u, v) ∉ S do
        S(u, v) = similarity(u, v));
    end for
end for
M ← ∅;
unpartitioned ← U;
while S is not empty do
    (u, v) ← pop pair with highest similarity from S;
    if u, v ∈ unpartitioned then
        M ← M ∪ (u, v);
        remove u, v from unpartitioned;
    end if
end while
add all nodes in unpartitioned to M partitioned just to themselves ;
return M;
```

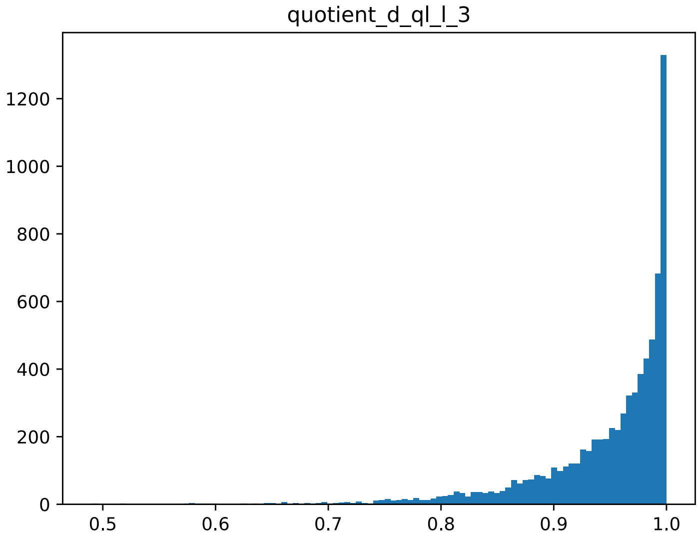
Further Results



Further Results



Further Results



Further Results

MILP L projection

$$
\begin{aligned}
\max \quad & \sum_{l' \in L_l^{i-1}} \sum_{q \in N^{i-1}(l')} a_{ql'}^{i-1} && \\
& s x_{l'}^{i-1} \geq y_{l'}^{i-1} && l' \in L_l^{i-1} \\
& \sum_{l' \in L_l^{i-1} \cap N^{i-1}(q)} a_{ql'}^{i-1} \leq \delta_q && q \in N^{i-1}(l) \\
& \sum_{q \in N^{i-1}(l')} a_{ql'}^{i-1} \leq y_{l'}^{i-1} && l' \in L_l^{i-1} \\
& \sum_{l' \in L_l^{i-1}} (c x_{l'}^{i-1} + b y_{l'}^{i-1}) \leq c \tilde{x}_l^i + b \tilde{y}_l^i && \\
& x_{l'}^{i-1} \in \{0, \ldots, r_{l'}^{i-1}\} && l' \in L_l^{i-1} \\
& y_{l'}^{i-1} \in \{0, \ldots, \lceil \bar{d}_{l'}^{i-1} \rceil\} && l' \in L_l^{i-1} \\
& 0 \leq a_{ql'}^{i-1} \leq e_{ql'}^{i-1} && l' \in L_l^{i-1},\ q \in N^{i-1}(l')
\end{aligned}
$$

Further Results

LP Q projection

$$
\begin{aligned}
\max \quad & \sum_{q' \in Q_q^{i-1}} \sum_{l \in \tilde{N}^i(q')} \tilde{a}^i_{q'l} && \\
& \sum_{l \in \tilde{N}^i(q')} \tilde{a}^i_{q'l} \le \tilde{d}^i_{q'} && q' \in Q_q^{i-1} \\
& \sum_{q' \in \tilde{N}^i(l) \cap Q_q^{i-1}} \tilde{a}^i_{q'l} \le a^i_{ql} && l \in N^i(q) \\
& 0 \le \tilde{a}^i_{q'l} \le \tilde{e}_{q'l} && q' \in Q_q^{i-1},\ l \in \tilde{N}^i(q)
\end{aligned}
$$