Large-scale Urban Facility Location Selection with Knowledge-informed Reinforcement Learning

Hongyuan Su*, Yu Zheng*
{suhy24,y-zheng19}@mails.tsinghua.edu.cn
Department of Electronic Engineering, BNRist,
Tsinghua University
Beijing, China

Jingtao Ding
dingjt15@tsinghua.org.cn
Department of Electronic Engineering, BNRist,
Tsinghua University
Beijing, China

Depeng Jin
jindp@tsinghua.edu.cn
Department of Electronic Engineering, BNRist,
Tsinghua University
Beijing, China

Yong Li
liyong07@tsinghua.edu.cn
Department of Electronic Engineering, BNRist,
Tsinghua University
Beijing, China

ABSTRACT

The facility location problem (FLP) is a classical combinatorial optimization challenge aimed at strategically laying out facilities to maximize their accessibility. In this paper, we propose a reinforcement learning method tailored to solve large-scale urban FLP, capable of producing near-optimal solutions at superfast inference speed. We distill the essential swap operation from local search, and simulate it by intelligently selecting edges on a graph of urban regions, guided by a knowledge-informed graph neural network, thus sidestepping the need for heavy computation of local search. Extensive experiments on four US cities with different geospatial conditions demonstrate that our approach can achieve comparable performance to commercial solvers with less than 5% accessibility loss, while displaying up to 1000 times speedup. We deploy our model as an online geospatial application at https://huggingface.co/spaces/tsinghua-fib-lab/MFLP.

CCS CONCEPTS

• **Computing methodologies** → **Reinforcement learning**; **Planning and scheduling**.

KEYWORDS

Facility location problem, reinforcement learning, large-scale optimization

*Both authors contributed equally to this research.

1 INTRODUCTION

In real cities, facility layout tends to deviate from residential demands for corresponding services, leading to costly travel [12, 13]. Therefore, optimizing accessibility by strategically locating urban facilities is crucial for creating more sustainable and inclusive cities. In fact, facility location problem (FLP) is a classic combinatorial optimization (CO) problem [2, 8], which is notoriously challenging due to the NP-hardness inherent in selecting p urban regions to place facilities from N candidate regions [5]. As both N and p are typically large in urban contexts, designing a reliable algorithm that delivers satisfactory solutions within reasonable timeframes is difficult. Furthermore, unlike the standard FLP setup, urban facilities exhibit multiple types, each with distinct budgets and residential needs, further complicating the task.

FLP is traditionally addressed using heuristics, metaheuristics [3, 4, 7] and commercial mixed integer programming (MIP) solvers [1]. Despite the near-optimal solutions they offer, the heavy computation of local search inherent in these approaches renders them suitable only for small-scale problems, while they tend to be unacceptably slow [11]. On the other hand, machine learning models are proposed to tackle FLP, *i.e.,* ML4CO [9, 10, 14], which replace intensive local search with learnable neural networks. Though ML4CO significantly accelerates solution generation through fast model inference, it has drastically degenerated accessibility performance for the overlooking rich prior knowledge inherent in FLP.

In this paper, we present a deep reinforcement learning (DRL) method to solve the large-scale urban facility location problem, which is able to achieve comparable accessibility performance to commercial solvers while generating solutions with up to 1000 times speedup. Specifically, we construct an urban graph with geospatial information, based on which we distill the essential *swap* operation in local search, replacing inefficient exploration with rapid edge selections which are informed by a GNN tailored to the FLP. Additionally, we integrate tabu search [7] and node benefits [6] into the graph via dynamic wiring to avoid redundant and low-quality solutions. Finally, to address practical urban FLP with multiple types of facilities, we design a *divide-and-conquer* algorithm to solve the problem with two stages, which first accomplishes single facility solutions independently, then conducts adjustments, both using the unified *SWAP* operation and GNN model.
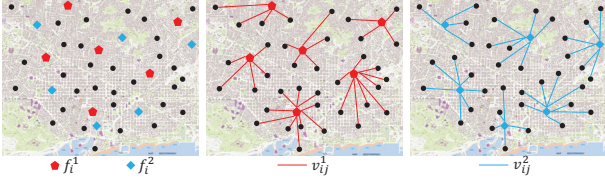
Figure 1: An example of MFLP. Red pentagons and blue diamonds are the placed two types of facilities. Red and blue edges indicate that residents of each region visit their closest facilities to access basic services. Best viewed in color.

To summarize, the contributions of this paper are as follows,

- To the best of our knowledge, we are the first to investigate practical large-scale urban FLP with multiple types of facilities.
- We develop a knowledge-informed RL method that incorporates rich prior knowledge of FLP into the design of Markov Decision Process (MDP) and GNN, as well as a divide-and-conquer algorithm that solves the problem with the unified RL model, bypassing the need for heavy computation of local search.
- We conduct experiments using real-world large-scale urban geospatial and human mobility data, demonstrating that our approach can achieve comparable performance to commercial solvers with less than 5% performance loss while generating solutions up to 1000 times faster.

2 PROBLEM FORMULATION

Given the total number of facilities P, candidate locations for facilities $\mathcal{N} = \{N_1, N_2, \cdots, N_n\}$, and corresponding demands $\mathcal{H} = \{h_1, h_2, \cdots, h_n\}$, the FLP aims to maximize the accessibility of facilities. Different from the FLP, which focuses on a single type of facility, we investigate Multi-Facility Location Problem (MFLP), which involves $\mathcal{K}$ different types of facilities which are distinguished by different superscripts, and the type k facility has to be placed in P^k different regions among $\mathcal{N}$, with the corresponding residential demands denoted as $\mathcal{H}^k = \{h_1^k, \ldots, h_n^k\}$. The MFLP aims to optimize accessibility to all types of facilities, which can be measured by the access cost (AC) as follows:

$$AC = \sum_{k \in \mathcal{K}} \sum_{i=1}^{n} \sum_{j=1}^{n} h_i^k d_{ij} Y_{ij}^k, \qquad (1)$$

where d_{ij} denotes the distance between location N_i and N_j, and Y_{ij} indicates whether demand h_i at N_i is satisfied by a facility at N_j.

For the purpose of modeling the interactions between different facilities aforementioned, we introduce incompatibility to MFLP, which is the key to differentiating MFLP from multiple independent FLPs. Incompatibility requires only one specific type of facility can be placed at each region, and can be characterized as,

$$X_i^p X_i^q \equiv 0, \ \forall p, q \in \mathcal{K}, p \neq q, N_i \in \mathcal{N}, \qquad (2)$$

where X_i^p is an indicator variable for whether a facility of type p is placed at N_i.

Figure 1 provides an illustrative example of MFLP for two different types of urban facilities. In particular, when $|K| = 1$, the MFLP will degenerate into the standard FLP problem.

3 METHOD

We introduce a knowledge-informed DRL approach to solve the large-scale FLP. Inspired by local search methods, the SWAP operator is developed to fully leverage the searched solution space. Based on the SWAP operator, we define a swap graph for the large-scale FLP and formulate the FLP as a sequential edge selection problem with a domain knowledge-driven dynamic wiring to identify high-quality SWAP operations. Additionally, a divide-and-conquer framework is further proposed to address the real-world MFLP, taking into account constraints between different types of facilities with a two-stage process.

3.1 The Universal SWAP operator

Cities of large scale contain thousands of candidate regions and require hundreds of facilities, making it time-consuming to generate solutions from scratch due to its repeated computations. To make full use of the already obtained solutions, we distill the universal SWAP operator based on the idea of local search [7], which generates new solutions by fine-tuning and improving the current solution. Specifically, each SWAP operator takes two locations, *insert* and *remove*, and moves the facility at *remove* to *insert*. Through successive SWAP operators, the initial solution can be adjusted towards better solutions iteratively until reaching the optimal.

3.2 Knowledge-Informed RL

As introduced previously, heuristic methods are accurate but slow, while ML4CO approaches accelerate solution generation with significant accuracy deterioration. We propose a knowledge-informed RL method that uses the best of both worlds, generating accurate solutions at a superfast speed, as illustrated in Figure 2.

MDP on Graph. We regard all the candidate locations as nodes and add edges between nodes with and without facility to construct a swap graph, which can be formulated as follows,

$$\mathcal{G}_t = \{\mathcal{N}, \mathcal{E}_t\}, e_{ij,t} = \mathbb{I}\{X_{i,t} < X_{j,t}\}. \qquad (3)$$

The SWAP operator is thus equivalent to edge selection on the swap graph, as illustrated in Figure 2 (b). Moreover, since the solution of MFLP is generated through a sequence of SWAP operators, solving MFLP can be framed as an MDP with the following elements:

- **State** summarizes the information about the topology of urban regions and the placement of facilities.
- **Action** denotes choosing an edge on the swap graph and exchanging the corresponding facility placement.
- **Reward** is the reduction of access cost defined by Equation (1).

Dynamic Wiring by FLP Knowledge. Not all edges on the swap graph benefit the reduction of the access cost, particularly when the solution is close to the optimal or falls into a local optimum. Here, we introduce a dynamic wiring technique informed by FLP knowledge, to initially assess the exploration value of each action. Specifically, as shown in Figure 2, our dynamic wiring involves filtering two categories of edges that contribute less to the optimal solution. *Tabu Edges* are inspired by Tabu Search [7], and recently visited locations by SWAP operators will be recorded as Tabu nodes, and edges containing Tabu Nodes are considered as Tabu Edges. *Negative Benefit Edges* indicate actions with significant negative returns. The change of total access cost after taking action can be
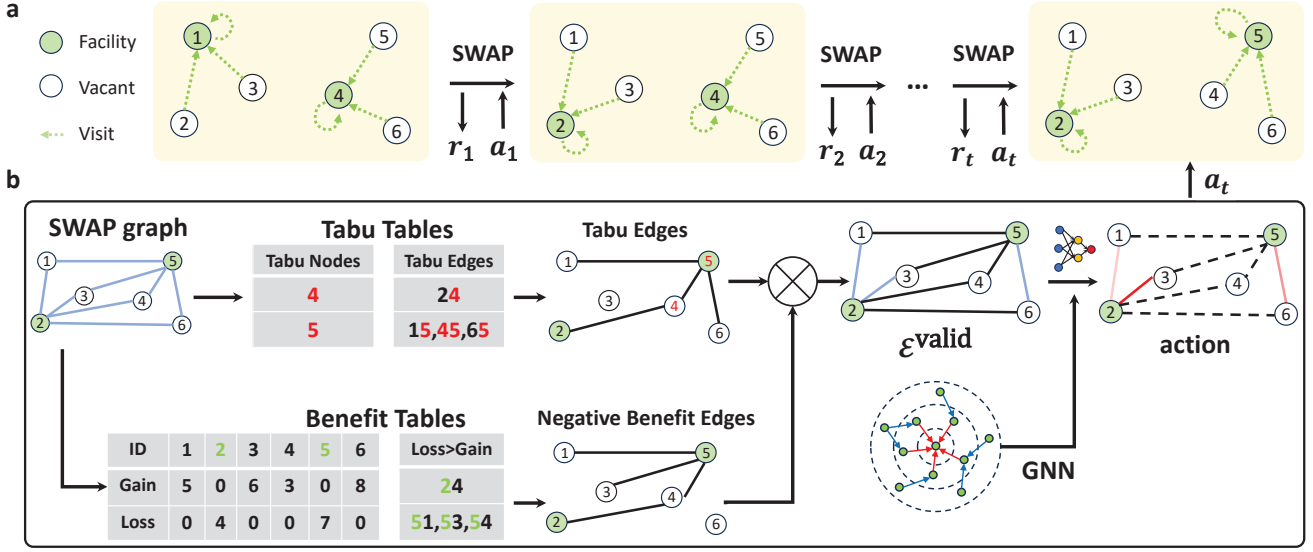
Figure 2: The overall RL framework solving the large-scale FLP. (a) FLP as a sequential decision-making problem on the swap graph, where one single facility is moved from its origin region (green) to one of the vacant regions (white) through the SWAP operator at each step. (b) The knowledge-informed RL agent for edge selection leverages the dynamic wiring technique to filter out both Tabu Edges and Negative Benefit Edges. Best viewed in color.

divided into three parts: the *gain* if only placing facility, the *loss* if only removing facility, and the *extra* term caused by simultaneously placing and removing. Notably, if two locations are far enough apart, the *extra* always equals to zero [6]. Consequently, if locations are far apart and the *gain* is less than the *loss*, the SWAP operator will increase the total access cost instead. Based on the above FLP knowledge, the dynamic wiring technique highlights the quality actions for facility placement and guarantees the accessibility performance of the obtained solutions.

GNN State Encoder. We design a GNN-based model to conduct SWAP operations, bypassing the time-consuming local search. Specifically, the proposed GNN considers both local topological relationship and global facility allocations to learn comprehensive embeddings for high-quality edges on the swap graph as follows,

$$\mathbf{n}_i^0 = \mathbf{W}_n^0 \mathbf{A}_{n_i}, \tag{4}$$

$$\mathbf{n}_i^{l+1} = \mathbf{n}_i^l + \tanh(\sum_{e_{ij}} \mathbf{W}_n^{l+1} \mathbf{n}_j^l), \tag{5}$$

$$\mathbf{e}_{ij} = [\mathbf{n}_i^L \parallel \mathbf{n}_j^L], \tag{6}$$

where $\mathbf{A}_{N_i^k}$ is the input attributes for nodes when placing facility k, $\mathbf{W}_n^l$ is the weight matrix of the l-th layer of the GNN, $\mathbf{e}_{ij}$ is the representation of edges on the swap graph, $\parallel$ means concentration, and $\mathbf{n}_i^L$ is the output of the last layer and L is a hyper-parameter for the number of GNN layers.

3.3 Divide-and-Conquer Framework

The proposed RL framework solves FLP with superior efficiency. We now extend it to address the incompatibility in the MFLP with a divide-and-conquer framework, which consists of two successive stages. In **Stage I**, the agent searches for the optimal locations

for each type of facility using the SWAP operator independently, starting from a greedily constructed initial solution. Since these independent solutions may conflict with the incompatibility constraints described by Equation (2), in **Stage II** the agent further refines these independent solutions by eliminating the conflicts via the SWAP operators as well. Stage II continues until all incompatibility conflicts are resolved.

With this divide-and-conquer framework, we can reuse the proposed RL algorithm for large-scale FLP to address the MFLP. On the one hand, both of the two stages share the aligned objective of minimizing the access cost, *i.e.*, generating separate optimal placements for each type of facility with lowest access cost in Stage I and resolving incompatibilities with the minimal increase in access cost in Stage II. On the other hand, we accomplish both stages employing the same universal SWAP operator. Consequently, our knowledge-informed RL approach kills two birds with one stone by intelligently conducting SWAP operators to optimize access cost, unifying the two challenging problems of FLP and MFLP.

4 EXPERIMENTS

We conduct extensive experiments based on the real world urban geospatial data and human mobility data from SafeGraph. Our method is compared against traditional methods such as genetic algorithm (GA), tabu search (TS), local search (LS). Moreover, advanced machine learning approach for CO problem (CardNN) [11] and commercial solver Gurobi are included.

4.1 Overall Performance

We evaluate our model across different large-scale cities with the results illustrated in Table 1, from which we have the following observations,

Table 1: Performance on MFLP, evaluated by access cost (AC) and algorithm runtime (RT) in seconds. Nan means the method fails to solve within 24h or with 100GB RAM.

Method	New York		Los Angeles		Chicago	
	AC↓	RT↓	AC↓	RT↓	AC↓	RT↓
GA	31.04	81.25	25.99	123.87	30.86	89.53
TS	28.86	3489.91	23.91	2714.75	28.16	3049.60
LS	27.10	311.54	23.63	498.35	28.93	286.62
VNS	26.94	1591.36	22.13	1189.88	26.71	1046.94
CardNN	26.62	906.29	21.76	784.11	26.72	609.28
Gurobi (120s)	27.48	120	23.43	120	28.04	120
Gurobi (optimal)	nan	nan	nan	nan	nan	nan
DRL-GNN (ours)	**26.29**	**36.62**	**21.08**	**24.65**	**25.58**	**26.86**
impr% / speed up	+4.9	24.8	+4.2	31.8	+4.4	39.0

- **Heuristic methods are ineffective for large-scale MFLP.** Classic heuristic methods like TS, LS, and VNS fail to generate high-quality solutions for MFLP, showing a noticeable relative gap of nearly 10% compared to the optimal solutions. In addition to the worse accessibility, the runtime is also significantly longer than other methods. These observations indicate the need for further improvements in search effectiveness.
- **Our methods balances effectiveness and efficiency as well.** Aside from Gurobi (optimal), our approach achieves the lowest access cost and offers the fastest solution generation speed across all four cities. Although the access cost is 4.2% higher than Gurobi (optimal) in the medium-sized Los Angeles scenario, the solution is generated 32 times faster. Moreover, our approach outperforms other ML4CO methods like CardNN, with a 2.5% lower access cost and a nearly 87-fold speed increase.

4.2 Performance on FLP

Since the complexity of the MFLP is much more higher than that of FLP, Gurobi often fails on solving large-scale MFLP. Considering that the performance of solving FLP serves as the foundation for MFLP, we conduct an additional comparison of different methods on the FLP, where Gurobi (optimal) can offer the optimal solutions. To investigate the performance on problems of different scales, we use synthetic data of urban regions and facility access needs, then evaluate the solutions generated by different methods.

As illustrated in Figure 3, our method excels in facility placement at remarkable speed across problems of different scales. Specifically, the relative gap of access cost between our approach and the optimal solutions remains below 3.7%. Notably, the speedup of our method reaches over 876 times when the city is huge and has numerous facilities, such as $(N, p) = (3000, 300)$, highlighting its outstanding performance in both solution quality and algorithm runtime.

5 CONCLUSION

In this paper, we present a knowledge-informed RL method tailored to solve real-world FLP and MFLP. By infusing rich prior knowledge into the dynamical wiring of GNN, our application can produce near-optimal solutions for large-scale FLP with up to 1000 times speedup compared to commercial solvers. Particularly, for New York city-scale scenarios, which are too large such that existing commercial solvers fail to handle, our method generates reliable
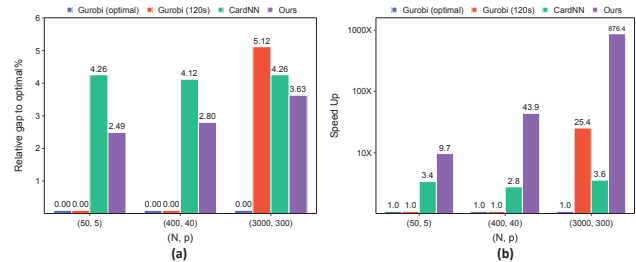

Figure 3: FLP performance comparison. For each pair (N,p), we generate 1000 simulated cities and calculate (a) the average relative gap% between each method and Gurobi (optimal). (b) the average speed up compared to Gurobi (optimal).

solutions within 40 seconds. Our method also enables a seamless trade-off between the accessibility performance and solution generation time through adjusting the number of swap steps performed.

ACKNOWLEDGMENTS

REFERENCES

[1] Erling D Andersen and Knud D Andersen. 2000. The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*. Springer, 197–232.
[2] Zvi Drezner and Horst W Hamacher. 2004. *Facility location: applications and theory*. Springer Science & Business Media.
[3] Harsha Gwalani, Chetan Tiwari, and Armin R Mikler. 2021. Evaluation of heuristics for the p-median problem: Scale and spatial demand distribution. *Computers, environment and urban systems* 88 (2021), 101656.
[4] Oded Kariv and S Louis Hakimi. 1979. An algorithmic approach to network location problems. I: The p-centers. *SIAM journal on applied mathematics* 37, 3 (1979), 513–538.
[5] FE Maranzana. 1964. On the location of supply points to minimize transport costs. *Journal of the Operational Research Society* 15, 3 (1964), 261–270.
[6] Mauricio GC Resende and Renato F Werneck. 2007. A fast swap-based local search procedure for location problems. *Annals of Operations Research* 150 (2007), 205–230.
[7] Erik Rolland, David A Schilling, and John R Current. 1997. An efficient tabu search procedure for the p-median problem. *European Journal of Operational Research* 96, 2 (1997), 329–342.
[8] David B Shmoys, Éva Tardos, and Karen Aardal. 1997. Approximation algorithms for facility location problems. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. 265–274.
[9] Hongyuan Su, Yu Zheng, Jingtao Ding, Depeng Jin, and Yong Li. 2024. Rumor Mitigation in Social Media Platforms with Deep Reinforcement Learning. In *Companion Proceedings of the ACM on Web Conference 2024*. 814–817.
[10] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. *Advances in neural information processing systems* 28 (2015).
[11] Runzhong Wang, Li Shen, Yiting Chen, Xiaokang Yang, Dacheng Tao, and Junchi Yan. 2022. Towards One-shot Neural Combinatorial Solvers: Theoretical and Empirical Notes on the Cardinality-Constrained Case. In *The Eleventh International Conference on Learning Representations*.
[12] Yanyan Xu, Luis E Olmos, Sofiane Abbar, and Marta C González. 2020. Deconstructing laws of accessibility and facility distribution in cities. *Science advances* 6, 37 (2020), eabb4112.
[13] Yu Zheng, Yuming Lin, Liang Zhao, Tinghai Wu, Depeng Jin, and Yong Li. 2023. Spatial planning of urban communities via deep reinforcement learning. *Nature Computational Science* 3, 9 (2023), 748–762.
[14] Yu Zheng, Hongyuan Su, Jingtao Ding, Depeng Jin, and Yong Li. 2023. Road Planning for Slums via Deep Reinforcement Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23)*. Association for Computing Machinery, New York, NY, USA, 5695–5706. https://doi.org/10.1145/3580305.3599901