Feasibility-Aware Masked Transformer for the Pickup-and-Delivery Problem with Time Windows (Student Abstract)

Kaede Saito[1], **Ryota Higa**[2], **Hiromu Imura** [2], **Masaaki Kondo**[1]
[1]Keio University, Japan
[2]NEC Corporation, Japan
sitkd@acsl.ics.keio.ac.jp, r-higaryouta@nec.com, imura.h@nec.com, kondo@acsl.ics.keio.ac.jp

Abstract

The Pickup-and-Delivery Problem with Time Windows (PDPTW) is a time-constrained variant of the vehicle-routing problem (VRP). Complex time constraints make it difficult to solve using existing NCO methods. In this paper, we present the Feasibility-Aware Masked Transformer (FAM-Trans) specialized for PDPTW. FAM-Trans integrates a lightweight side encoder with a context-aware embedding scheme that effectively captures temporal dependencies. A dynamic key-value module continuously updates node embeddings as the route progresses. During inference, a feasibility-guided post-inference filtering strategy suppresses constraint violations without post-hoc repair. Experiments on standard PDPTW benchmarks show that FAM-Trans outperforms NCO baselines by 20–35% in solution quality and constraint satisfaction.

Introduction

Neural combinatorial optimization (NCO) methods have emerged as a framework for vehicle routing problems(VRPs), enabling the generation of high-quality schedules without problem-specific expert knowledge (Kool, Van Hoof, and Welling 2018; Kwon et al. 2020).

The Pickup and Delivery Problem with Time Windows (PDPTW) is a variant of the VRP that involves paired pickup and delivery locations and time window constraints. A primary challenge in solving the PDPTW with NCO solvers is the complex interaction between precedence and temporal constraints. While NCO solvers typically use feasibility masks to exclude actions that violate constraints, this approach has limitations in the PDPTW. Even if selecting a pickup node satisfies its own time window, the interdependence with its delivery partner can lead to a downstream infeasibility, which degrades model performance. Therefore, a model that can learn temporal dependencies is necessary.

In this paper, we propose the Feasibility-Aware Masked Transformer (FAM-Trans). We propose a mechanism that introduces a node-specific dynamic key-value, incorporating both global state and pairwise spatial information into the decoder's attention process. This includes a dynamic context embedding that ensures the decoder's query understands the changing route state. During training, we treat time windows as soft constraints, using a penalty function to encourage exploration and avoid overly conservative policies. The model is also enhanced by two auxiliary mechanisms: a one-step look-ahead mask that proactively reduces the search space, and a post-inference filter to ensure feasibility.

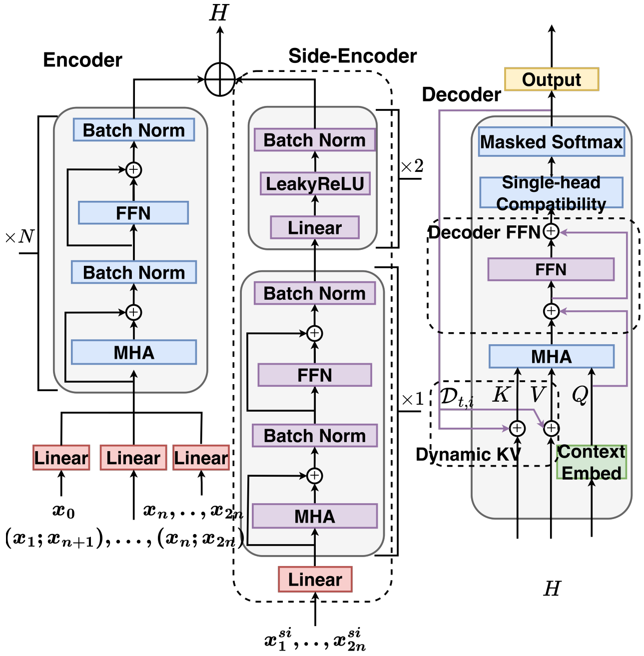

Figure 1: The architecture of our proposed FAM-Trans model.

Overall FAM-Trans Architecture

FAM-Trans is an encoder-decoder architecture model based on the Attention Model (AM) (Kool, Van Hoof, and Welling 2018).The overall model architecture is illustrated in Fig 1. The FAM-Trans introduces four important additional modules to the backbone model:

1. Side-Encoder: A lightweight encoder layer similar to the side adapter (Lin et al. 2024) to learn temporal features intensively. This module improves the main encoder's output by providing additional information about tempo-

---

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Methods	PDPTW20				PDPTW50				PDPTW100			
	Obj.	Gap	IFS	Time	Obj.	Gap	IFS	Time	Obj.	Gap	IFS	Time
LKH3	7.56	-6.65%	0	5h07m	-	-	-	-	-	-	-	-
LKH3(100s)	7.57	-6.59%	8	4h19m	15.38	-10.69%	48	8h12m	27.93	-9.49%	6039	8h03m
OR-Tools	8.10	0.00%	0	9m	17.22	0.00%	0	1h15m	30.86	0.00%	0	6h17m
AM(greedy)	10.35	27.77%	424	1s	23.32	35.44%	655	3s	43.35	40.49%	705	7s
AM(sample)	9.78	20.74%	72	3m	21.89	27.17%	86	10m48	41.20	33.53%	91	36m
Ours(greedy)	8.57	5.82%	123	2s	18.20	5.74%	389	5s	32.73	6.08%	450	10s
Ours(sample)	8.00	-1.22%	47	4.8m	16.69	-3.04%	99	18m32	30.14	-2.33%	93	1h12m
AM*(greedy)	10.09	24.61%	0	4s	22.87	32.86%	0	8s	43.35	40.49%	0	5s
AM*(sample)	9.77	20.67%	0	3m	21.90	27.18%	0	10m41s	41.20	33.52%	0	31m
Ours*(greedy)	8.53	5.30%	0	5s	18.02	4.67%	0	10s	32.50	5.32%	0	15s
Ours*(sample)	8.00	-1.22%	0	5m	16.70	-3.02%	0	18m13s	30.14	-2.32%	0	1h04m

Table 1: Comparison Results for 10000 PDPTW instances. FAM-Trans(FAM) attains 99% feasibility across all constraints without any post-inference filtering in sampling method. Results marked with * were obtained with our post-inference filtering method. The time shows the total execution time for all instances.

ral features.

2. Context Embedding: The context vector h_c provides all the dynamic information needed for decision-making, such as the graph embedding $\bar{h}$ (the means of the encoder's output), the vehicle's state C and the most urgent deadlines $s_t^{deliv\_min}$ for timestep t.

$$h_c = \text{Concat}(\bar{h}; h_{\pi_{t-1}}; C; s_t^{deliv\_min}) \tag{1}$$

3. Dynamic Key Value (KV): Dynamic information for node i (the time window slack s_t^i and node distance from $c_{\pi_{t-1},i}$) based on the current state is added to the decoder's attention keys and values using the dynamic vector $\mathcal{D}_{t,i}$.

$$\mathcal{D}_{t,i} = \text{Concat}(s_t^i, c_{\pi_{t-1},i}) \tag{2}$$

$$k_{t,i} = W_i^K h_i + W_i^{DK} \mathcal{D}_{t,i}, \tag{3}$$

$$v_{t,i} = W_i^V h_i + W_i^{DV} \mathcal{D}_{t,i} \tag{4}$$

4. Decoder Feed Forward Network (FFN): To enhance the decoder's expressive power, we adopt a non-linear glimpse mechanism inspired by Huang et al. (Huang et al. 2025).

To more effectively prune the search space, we introduce an additional mask based on one-step look-ahead masking. This heuristic proactively identifies and filters out pickup nodes that would inevitably result in a time-window violation.

We also employ a Post-Inference Process to ensure solution feasibility. During inference, if the output solution is found to be infeasible, we filter it and trigger a re-inference process. Instead of generating new solutions from the same input, we enhance the inputs through data augmentation(Kwon et al. 2020), creating new outputs.

Experimental Results

Results and Observations We evaluate FAM-Trans on randomly generated PDPTW instances of sizes $n = 20, 50, 100$. We compare against classical solvers (LKH3, Google OR-Tools) and a neural baseline, AM. All neural methods were run on a single NVIDIA V100 32GB GPU with instances processed in parallel. Classical solvers were evaluated on an Intel Xeon Gold 6240 CPU. Results for LKH3 with $n = 50$ and $n = 100$ are omitted due to excessive computation times. The LKH3(100s) variant has a 100-second time limit, with unsolved instances counted as failures. The evaluation results are summarized in Table 1, which compares the average objective value (mean of the sum of total length and violation penalty, where a shorter value is better), the optimality gap from OR-Tools, the number of infeasible solutions (IFS), and the total running time (10000 instances). Compared to Google OR-Tools, our sampling strategy produces superior solutions in significantly less time. In terms of IFS, FAM-Trans (FAM) is highly effective as well. Even without post-inference filtering, our sampling strategy achieves few IFS, and the simple greedy strategy also outperforms AM. After applying our post-inference filter, we achieve zero IFS across all strategies, which demonstrates its effectiveness in eliminating invalid solutions.

Although the AM baseline also secures low IFS, we believe this is due to the model adopting an overly conservative strategy. For instance, it often chooses oversimplified solutions, such as visiting a delivery node immediately after a pickup node. While this approach satisfies the constraints, it is suboptimal with respect to the overall objective.

While the classical heuristic LKH3 performs well on smaller instances, its computation time increases exponentially with larger problem sizes. Notably, for $n = 100$, LKH3 (100s) fails on the majority of cases. These results show that FAM effectively balances solution quality and computational efficiency, especially for larger-scale problems.

Conclusion

We propose the Feasibility-Aware Masked Transformer (FAM-Trans), a novel neural model for the PDPTW. Experimental results show that FAM-Trans achieves a high feasibility rate and produces better solutions than the baseline model. Furthermore, FAM-Trans's performance is comparable to leading classical solvers like LKH3 and OR-Tools, while significantly reducing computation time.

References

Huang, Z.; Zhou, J.; Cao, Z.; and Xu, Y. 2025. Rethinking light decoder-based solvers for vehicle routing problems. *arXiv preprint arXiv:2503.00753*.

Kool, W.; Van Hoof, H.; and Welling, M. 2018. Attention, learn to solve routing problems! *arXiv preprint arXiv:1803.08475*.

Kwon, Y.-D.; Choo, J.; Kim, B.; Yoon, I.; Gwon, Y.; and Min, S. 2020. Pomo: Policy optimization with multiple optima for reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 21188–21198.

Lin, Z.; Wu, Y.; Zhou, B.; Cao, Z.; Song, W.; Zhang, Y.; and Jayavelu, S. 2024. Cross-problem learning for solving vehicle routing problems. *arXiv preprint arXiv:2404.11677*.