Hyper-Relation Fusion for Solving Multi-depot Vehicle Routing Problems

1st Xingyan Liu
School of Future Technology
South China University of Technology
Guangzhou, China
ftlxy0068@mail.scut.edu.cn

2nd Yang Wang
School of Future Technology
South China University of Technology
Guangzhou, China
ftwangyang@mail.scut.edu.com

3rd Fansen Meng
School of Future Technology
South China University of Technology
Guangzhou, China
202364800101@mail.scut.edu.cn

4th Ya-Hui Jia*
School of Future Technology
South China University of Technology
Guangzhou, China
jia.yahui@foxmail.com

***Abstract*—Multi-Depot Vehicle Routing Problem (MDVRP) requires constructing routes from multiple depots to geographically dispersed customers under capacity constraints. Unlike single-depot routing problems, MDVRP requires determining not only the routing relationship between customers but also the assignment relationship of customers to depots. In this paper, we propose a Hyper-Relation Fusion (HRF) neural combinatorial optimization algorithm to solve MDVRP, considering both heterogeneous relationships and homogeneous relationships between depots and customers. The heterogeneous relationships of depot-customer and customer-customer are captured through graph attention to distinguish different types of connectivity. The homogeneous relationships are learned by aggregating the features of all nodes via a graph convolutional network. Finally, HRF fuses the original node features, heterogeneous features, and homogeneous features, which are further processed through an encoder-decoder architecture to generate the solution. Comprehensive experiments on synthetic and benchmark datasets demonstrate that HRF surpasses the state-of-the-art metaheuristics and learning-based methods in solution quality. Our code is available at https://github.com/lxy0068/HRF-MDVRP.**

***Index Terms*—Multi-Depot Vehicle Routing Problems, Hyper-Relation Fusion, Graph Convolution Networks, Deep Reinforcement Learning.**

I. Introduction

Multi-Depot Vehicle Routing Problem (MDVRP) [1], [2] is a classic combinatorial optimization problem in logistics and transportation. The goal of MDVRP is to construct cost-minimized transportation routes between multiple depots and geographically distributed customers. Traditional methods for MDVRP encompass both exact and metaheuristic methods. While exact solvers [3] guarantee optimal solutions, they exhibit severe scalability limitations, rendering them impractical for large instances. Metaheuristic methods like tabu search or genetic algorithms [4] have good scalability but rely heavily on domain knowledge and extensive tuning [5].

In recent years, Neural Combinatorial Optimization (NCO) [6], which integrates Deep Reinforcement Learning (DRL) and neural networks to learn routing strategies directly from data, has shown strong performance in single-depot vehicle routing problems (VRP) [7], [8]. However, when multiple depots are involved, the relationship between depots and customers becomes more complex than the single-depot problem. Specifically, in MDVRP, not only should routing sequences between customers should be considered, but also the assignment of customers to depots should be determined. These two aspects of decision-making mutually affect each other. In essence, depots and customers have both homogeneous relationships (they are all points on a map) and heterogeneous relationships (the depot itself is not a task and does not consume the vehicle's carrying capacity) [9]. Effectively identifying and utilizing these relationships is key to making good assignment and routing decisions for multi-depot problems. Li *et al.* [10] attempted to improve the model by designing a Multi-Depot with Multi-Type Attention mechanism (MD-MTA), but the complex relationships in the multi-depot environment are still insufficiently captured. There is still plenty of room for improvement.

To effectively integrate both homogeneous and heterogeneous relationships in MDVRP, we propose a Hyper-Relation Fusion (HRF) mechanism with two different Graph Neural Networks (GNNs), denoted as HoGNN and HeGNN, respectively. Specifically, HoGNN extracts homogeneous features by configuring a Graph Convolutional Network (GCN) to exploit the spatial and topological similarities among points on the map. Meanwhile, HeGNN is a specialized Graph Attention Network (GAT) to discern heterogeneous interactions involving distinct node roles and integrate key attributes like capacity constraints. We then fuse these extracted features with the original node attributes, thereby enhancing the learned feature representation. The contributions of this paper are summarized

*Corresponding author.

This work is supported in part by the National Natural Science Foundation of China under Grant 62206098, in part by the Guangdong Basic and Applied Basic Research Foundation 2023A1515012896, and in part by the Department of Science and Technology of Guangdong Province 2023ZT10L145.

as follows:

1) We propose a novel HRF mechanism that integrates homogeneous and heterogeneous relationships in the MDVRP. This mechanism utilizes two distinct GNNs to extract and combine spatial, topological, and task-specific features from the problem's structure.
2) The HRF mechanism employs two specialized GNNs: a) HoGNN leverages a GCN to capture spatial and topological similarities among nodes by propagating multi-relational messages across node neighborhoods at varying granularities, and b) HeGNN uses a GAT to model complex interactions and constraints, such as vehicle capacity, between depots and customers. The outputs of both networks are then fused to enrich the feature representation, leading to improved routing decisions.

The rest of this paper is organized as follows. Section II reviews relevant research on MDVRP, including traditional and NCO methods. Section III describes the Markov process for MDVRP. Section IV provides a detailed description of HRF. Section V presents and analyzes the experimental results. Finally, Section VI concludes the paper.

II. Related Work

A. Traditional Methods for Solving MDVRPs

MDVRP has been addressed using a range of heuristic and metaheuristic frameworks, including tabu search [1] and genetic algorithms [11]. These methods typically seek near-optimal solutions by iteratively improving feasible routes, leveraging mechanisms such as memory structures or biologically inspired operators. Later research combined multiple neighborhood search with tabu-based techniques to explore solution spaces more effectively [12], whereas hybridization strategies, such as injecting granular local searches and shaking mechanisms, have been proposed to enhance both intensification and diversification phases [13].

Despite these advancements, traditional methods face several prominent shortcomings. First, they often exhibit prohibitive computational requirements on large-scale instances, as the complexity of neighborhood exploration grows rapidly. Second, hand-crafted heuristics and parameter tuning can be excessively time-consuming and highly sensitive to instance characteristics.

B. NCO Methods for Solving VRPs

NCO has shown remarkable potential for tackling VRPs by leveraging data-driven policies that minimize human intervention and parameter tuning. Early explorations employed sequence-to-sequence frameworks such as the pointer network [14], [15] in conjunction with policy gradient algorithms to solve the Traveling Salesman Problem (TSP). These methods were later refined by incorporating self-attention [16], giving rise to the attention model [7], which demonstrated superior performance on TSP and Capacitated VRP (CVRP). Subsequent studies introduced more sophisticated sampling strategies to improve training stability and solution quality. For instance, Kwon *et al.* [17] proposed Policy Optimization with Multiple Optima (POMO), which uses multiple symmetric solutions of a single problem instance as baselines, achieving near-optimal performance on TSP and CVRP.

To handle multi-depot settings, numerous works modify attention-based models to incorporate depot-specific embeddings. Zou *et al.* [18] introduced an attention-to-attention (TAOA) mechanism for MDVRP, whereas Zhang *et al.* [19] proposed a Graph Attention Network (GAT-RL) to address time-window constraints. Recently, Li *et al.* [10] presented a multi-type attention framework tailored to solve MDVRP and Multi-Depot Open Vehicle Routing Problem (MDOVRP), integrating multiple attention layers that separately process depot-related contexts and vehicle interactions.

Existing NCO methods perform well in solving VRPs, especially in single-depot environments with remarkable results. However, these methods fail to deal effectively with heterogeneous relationships between depots and customers and between customers. Many methods treat all edges as homogeneous relationships and ignore the essential differences between different types of edges (i.e., depot-customer and customer-customer). The lack of a clear distinction between these edges leads to poor model performance in dealing with complex multiple dependencies between depots.

III. Problem Formulation

The representation of MDVRP can also be formulated as a Markov Decision Process (MDP), which can then be solved using DRL. Our formulation assumes static problem conditions to isolate the core routing dynamics from real-time uncertainties, enabling tractable policy learning. Formally, an MDP is defined by:

$$\mathcal{M} = \big(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}\big), \tag{1}$$

where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{T}$ is the transition function, and $\mathcal{R}$ is the reward function.

State: At each time step t, the state S^t consists of a vehicle state M^t and a node state X^t. The vehicle state $M^t = \big(C^t, L^t, O^t\big)$ includes the remaining capacity C^t, total traveled distance L^t, and the visiting sequence O^t. The node state $X^t = \big\{\big(G_i, D_i^t\big)\big\}$ contains the coordinates G_i and current demands D_i^t.

Action: An action $A^t \in \mathcal{A}$ specifies the next node x_j^t to visit.

Transition Function: The state evolves from S^t to S^{t+1} as follows:

$$C^{t+1} = \begin{cases} \max\big(0,\, C^t - D_j^t\big), & \text{if } x_j \text{ is a customer,} \\ C_{\max}, & \text{if } x_j \text{ is a depot,} \end{cases} \tag{2}$$

$$L^{t+1} = L^t + Z\big(x_i, x_j\big), \tag{3}$$

$$O^{t+1} = [\,O^t,\, x_j^t\,], \quad D_j^{t+1} = 0, \tag{4}$$

where $Z(x_i, x_j)$ denotes the travel cost from node x_i to node x_j.

Reward Function: Upon reaching a terminal state S^f with total route length L^f, we assign a terminal reward $R = -L^f$.

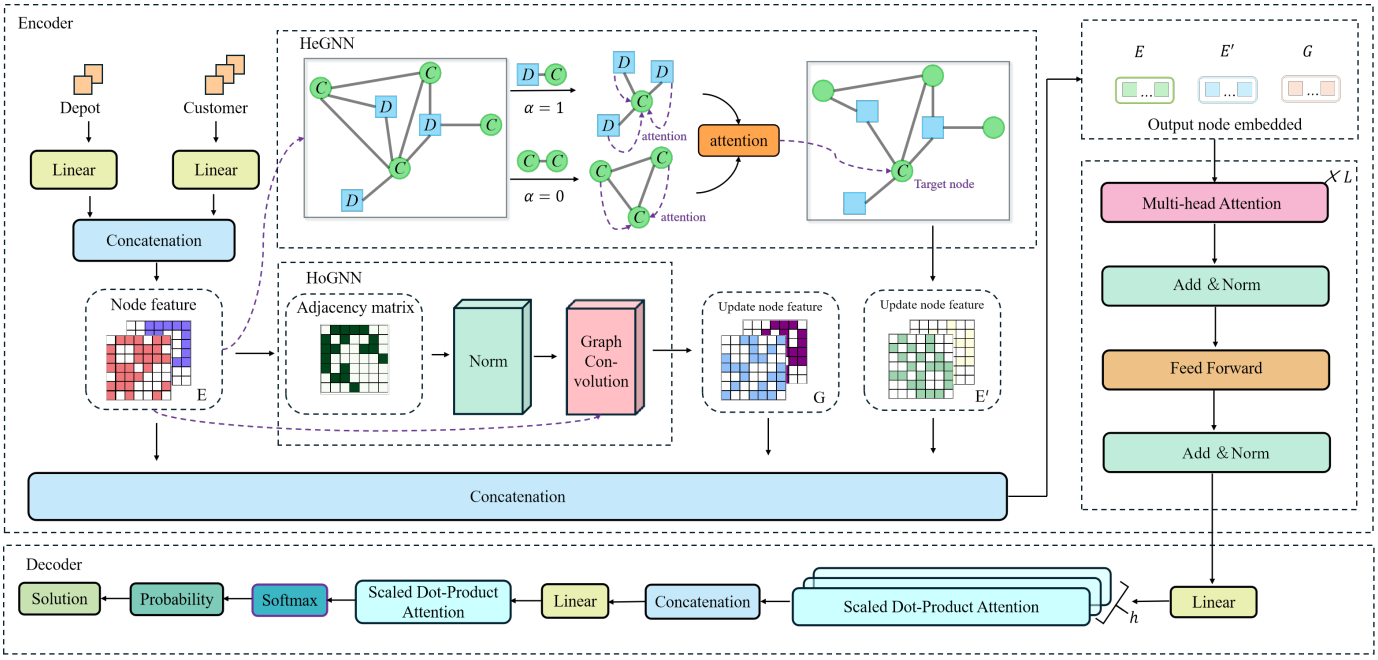

Fig. 1: Overview of the HRF framework.

Policy: A policy π_θ parameterized by θ maps each state S^t to a probability distribution over possible actions A^t. The probability of generating a valid route trajectory $\tau = (S^0, A^0, \ldots, S^f)$ under π_θ is:

$$p(S^f \mid S^0) = \prod_{t=0}^{f-1} \pi_\theta(A^t \mid S^t). \tag{5}$$

The objective is to learn an optimal policy that maximizes expected cumulative rewards, essentially minimizing the total traveling distance.

IV. Hyper-Relation Fusion Model

A. Architecture

1) Overview: HRF follows an Encoder-Decoder framework. The encoder embeds features of depots and customers into unified node representations. These representations are processed separately by two specialized GNNs. HoGNN employs GCN to aggregate neighborhood information uniformly via fixed adjacency weights, whereas HeGNN adopts GAT to learn dynamic edge-specific attention weights—the former captures static spatial topology while the latter adapts to context-aware relational constraints. HoGNN focuses on spatial and topological patterns among all nodes, while HeGNN considers more complex factors, such as the roles of depots and capacity constraints. The extracted homogeneous and heterogeneous features are then fused with the original node embeddings, resulting in enhanced hyper-relation features. The decoder uses these enriched features to iteratively generate the next node selection probabilities, constructing them into a valid solution step-by-step.

2) Encoder: The encoder extracts features of the depot and customer from multiple perspectives and constructs a high-dimensional representation space from these features for capturing multiple relationships between the depot and customer. First, we embed the features of depot and customer separately into a unified high-dimensional space $\mathbf{X}_{\text{depot}} \in \mathbb{R}^{B\times D\times 2}$ and $\mathbf{X}_{\text{cust}} \in \mathbb{R}^{B\times C\times 3}$ denote the depot and customer feature tensors, respectively.

$$\mathbf{E}_{\text{depot}} = \mathbf{W}_d\, \mathbf{X}_{\text{depot}} + \mathbf{b}_d, \tag{6}$$

$$\mathbf{E}_{\text{cust}} = \mathbf{W}_c\, \mathbf{X}_{\text{cust}} + \mathbf{b}_c, \tag{7}$$

where $\mathbf{W}_d$ and $\mathbf{W}_c$ are the learned weight matrix and $\mathbf{b}_d$ and $\mathbf{b}_c$ are the bias term. These two embedding vectors are spliced along the node dimensions to obtain the overall node embedding matrix:

$$\mathbf{E} = \left[\mathbf{E}_{\text{depot}}, \mathbf{E}_{\text{cust}}\right] \in \mathbb{R}^{B\times(D+C)\times E}. \tag{8}$$

HeGNN. To capture rich relational information beyond simple adjacency, we introduce a heterogeneous-relational concept by defining edge types $\alpha \in \{0, 1\}$ for Depot–Customer ($\alpha = 0$) and Customer–Customer ($\alpha = 1$) connections. Let $\mathbf{\Gamma}$ be an edge-index specifying the directed pairs (i, j) of nodes along with an edge-type tensor $\boldsymbol{\alpha}$. Denoting $\mathbf{E} \in \mathbb{R}^{N\times E}$ as the concatenated embeddings (with $N = D + C$), the HeGNN applies type-specific transformations:

$$\mathbf{m}_{ij}^{(\alpha)} = \mathbf{E}_j \mathbf{W}_{\text{edge}}^{(\alpha)} \in \mathbb{R}^{B\times E}. \tag{9}$$

Then, we compute the importance coefficient of each edge through the attention mechanism:

$$\mathbf{a}_{ij}^{(\alpha)} = \text{Softmax}_j\Big(\sigma\Big(\mathbf{W}_{\text{att}}^{(\alpha)}\left[\mathbf{E}_i \,\|\, \mathbf{m}_{ij}^{(\alpha)}\right]\Big)\Big), \tag{10}$$

where $\|$ denotes concatenation and $\sigma(\cdot)$ denotes a leaky ReLU activation. Each node i is updated via:

$$\mathbf{E}'_i = \sum_{\alpha\in\{0,1\}} \sum_{j\in\mathcal{N}_i^{(\alpha)}} \mathbf{a}_{ij}^{(\alpha)} \mathbf{m}_{ij}^{(\alpha)} \in \mathbb{R}^{B\times E}. \tag{11}$$

HoGNN. In parallel to HeGNN, we employ HoGNN to capture homogeneous spatial and topological relationships between nodes. By computing the normalized adjacency matrix:

$$\widetilde{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}, \quad \mathbf{D}_{ii} = \sum_{k=1}^{N} \mathbf{A}_{ik}, \tag{12}$$

where $\mathbf{A}$ is the adjacency matrix and $\mathbf{D}$ is the degree matrix, the normalized adjacency matrix is used to update the node features:

$$\mathbf{G} = \sigma\big(\widetilde{\mathbf{A}}\mathbf{E}\mathbf{W}_g\big). \tag{13}$$

Fusion. To effectively fuse hyper-relations while preventing feature dimensionality explosion, the outputs of HeGNN ($\mathbf{E}'$) and HoGNN ($\mathbf{G}$) are concatenated with the original embeddings $\mathbf{E}$ and projected via:

$$\mathbf{h} = \big[\,\mathbf{E}\,\|\,\mathbf{E}'\,\|\,\mathbf{G}\,\big]\mathbf{W}_f \in \mathbb{R}^{B\times N\times E}, \quad \mathbf{W}_f \in \mathbb{R}^{3E\times E}, \tag{14}$$

where $\big[\|\big]$ indicates hyper-relations information fusion.

Then, we employ a Multi-Head Attention (MHA) within each encoder layer. In each MHA layer, we apply skip connections and layer normalization (LN), as shown in the following recursive formulation:

$$\hat{\mathbf{h}}_i^{(l)} = \mathrm{LN}^l\left(\mathbf{h}_i^{(l-1)}\right) + \mathrm{MHA}^{(l)}\left(\mathbf{h}_i^{(l-1)}\right), \tag{15}$$

where l and $l-1$ represent the current and previous MHA layers, respectively. Additionally, a feedforward (FF) network is applied after the MHA block:

$$\mathbf{h}_i^{(l)} = \mathrm{LN}^l\left(\hat{\mathbf{h}}_i^{(l)}\right) + \mathrm{FF}^l\left(\hat{\mathbf{h}}_i^{(l)}\right), \tag{16}$$

where the FF network contains a hidden sublayer with ReLU activations. The recursive application of MHA and FF layers generates the final node embeddings after N layers, denoted by $\mathbf{h}_i^{(N)}$. These final embeddings are used as static node representations, which are then reused in the decoding step.

The encoding process is performed only once and generates the static node embeddings $\mathbf{h}_i^{(N)}$, which are passed on to the decoder for further processing.

3) Decoder: The decoder architecture in this study consists of two main components: an MHA layer followed by a single-attention (SA) layer, as proposed by Kool *et al.* [20]. In the MHA layer, the computation begins by generating the queries, keys, and values for each node. These are computed using learned weight matrices $\mathbf{W}^Q$, $\mathbf{W}^K$, and $\mathbf{W}^V$ as follows:

$$\mathbf{Q}_c = \mathbf{W}^Q\mathbf{h}_c, \quad \mathbf{K}_i = \mathbf{W}^K\mathbf{h}_i, \quad \mathbf{V}_i = \mathbf{W}^V\mathbf{h}_i, \tag{17}$$

where $\mathbf{h}_c$ is the concatenated vector formed by the embedding $\mathbf{h}_t$ of the current visited node and the attribute vector $\mathbf{a}_t$:

$$\mathbf{h}_c = \mathrm{Concat}(\mathbf{h}_t, \mathbf{a}_t), \tag{18}$$

where $\mathbf{h}_t$ represents the embedding of the currently visited node at time t, and $\mathbf{a}_t$ represents the corresponding attribute vector associated with the node.

After the MHA mechanism, the decoder passes through an SA layer. In this layer, the compatibility score $\mathbf{u}_{cj}$ between the current query $\mathbf{q}_c$ and the key $\mathbf{k}_j$ for each node is computed. This score quantifies how relevant a node is to the current node being processed. The compatibility score is computed using the following equation:

$$\mathbf{u}_{cj} = \begin{cases} 10\cdot\tanh\left(\frac{\mathbf{q}_c^\top \mathbf{k}_j}{\sqrt{d_k}}\right) & \text{if } j\notin m_t, \\ -\infty & \text{if } j\in m_t, \end{cases} \tag{19}$$

where d_k is the dimension of the key vectors, and m_t denotes the set of masked nodes that should be excluded from the computation (e.g., nodes that are not yet available for selection). The hyperbolic tangent ($\tanh$) is applied to scale the compatibility score, ensuring that it remains within a reasonable range.

Once the compatibility scores $\mathbf{u}_{cj}$ are computed, they are passed through a softmax function to produce a probability distribution over all possible node selections:

$$p_i = \frac{e^{\mathbf{u}_i}}{\sum_j e^{\mathbf{u}_j}}, \tag{20}$$

where p_i represents the probability of selecting node i as the next node in the decoding sequence. This softmax operation normalizes the compatibility scores into a valid probability distribution, ensuring that the sum of probabilities across all nodes is equal to 1. The node with the highest probability p_i is then selected, guiding the decoder in making the next decision in the sequence.

The HRF training procedure is presented in Algorithm 1.

Algorithm 1 HRF Training

Input: Model Θ, instance $\mathcal{I}$, augmentations A, samples S
Output: Optimal route $\mathcal{R}^*$
Augment: For $a = 1$ to A, generate $\mathcal{I}^{(a)} \leftarrow \textsc{Augment}(\mathcal{I}, a)$
Initialize $\mathcal{R}^* \leftarrow \varnothing$, $\quad \mathrm{Cost}(\mathcal{R}^*) \leftarrow \infty$
for $a = 1$ to A **do**
  for $s = 1$ to S **do**
    $\mathcal{R} \leftarrow \mathrm{HRF}(\Theta, \mathcal{I}^{(a)})$
    if $\mathrm{Cost}(\mathcal{R}) < \mathrm{Cost}(\mathcal{R}^*)$ **then**
      $\mathcal{R}^* \leftarrow \mathcal{R}$
    end if
  end for
end for
return $\mathcal{R}^*$

B. Inference

The three strategies used for inference in this paper are as follows:

Greedy Strategy: This strategy selects the node with the highest probability to visit, meaning that the same result will

TABLE I: The Result of Synthetic Dataset on MDVRP-3D.

Method	MDVRP-20C3D			MDVRP-50C3D			MDVRP-100C3D		
	Obj.	Gap (%)	Time (m)	Obj.	Gap (%)	Time (m)	Obj.	Gap (%)	Time (m)
Gurobi	4.13	**0.00**	2.97	-	-	-	-	-	-
Gurobi(60s)	4.13	**0.00**	2.97	7.39	0.68	857.46	12.38	3.25	2001.62
OR-Tools(60s)	4.43	7.26	1982.36	7.86	7.08	2000.96	12.50	4.25	2001.23
HGA(60s)	4.15	0.48	83.46	7.39	0.68	347.36	12.40	3.42	2001.31
VTNS(60s)	4.14	0.24	320.95	7.41	0.95	1032.72	12.31	2.67	2000.97
TAOA(aug.8, g.)	4.20	1.69	0.03	7.57	3.13	0.09	12.41	3.50	0.52
TAOA(aug.8, s.400)	4.16	0.72	10.86	7.46	1.63	34.12	12.23	2.00	162.74
GAT-RL(aug.8, g.)	4.19	1.45	0.04	7.53	2.59	0.11	12.36	3.08	0.56
GAT-RL(aug.8, s.400)	4.15	0.48	12.32	7.43	1.22	36.52	12.19	1.67	153.48
POMO (aug.8, g.)	4.18	1.21	0.03	7.47	1.77	0.09	12.26	2.25	0.47
POMO (aug.8, s.400)	4.14	0.24	10.19	7.39	0.68	33.92	12.15	1.33	151.84
MD-MTA (aug.10, g.)	4.16	0.72	0.07	7.42	1.09	0.20	12.17	1.50	0.80
MD-MTA (aug.10, s.200)	4.13	**0.00**	10.28	7.36	0.27	32.27	12.01	0.17	156.18
HRF(aug.10, g.)	4.14	0.24	0.06	7.35	0.13	0.18	12.00	0.08	0.98
HRF(aug.10, s.200)	4.13	**0.00**	9.89	7.34	**0.00**	30.58	11.99	**0.00**	160.42

TABLE II: The Result of Synthetic Dataset on MDOVRP-3D.

Method	MDOVRP-20C3D			MDOVRP-50C3D			MDOVRP-100C3D		
	Obj.	Gap (%)	Time (m)	Obj.	Gap (%)	Time (m)	Obj.	Gap (%)	Time (m)
Gurobi	2.95	**0.00**	2.28	-	-	-	-	-	-
Gurobi(60s)	2.95	**0.00**	2.28	5.15	0.98	793.24	8.31	3.23	2000.84
OR-Tools(60s)	3.18	7.80	1238.35	5.43	6.47	2001.25	8.47	5.22	2001.65
HGA(60s)	2.98	1.02	78.62	5.18	1.57	325.35	8.41	4.47	2001.28
VTNS(60s)	2.96	0.34	159.63	5.18	1.57	487.84	8.24	2.36	1336.86
TAOA(aug.8, g.)	3.04	3.05	0.03	5.29	3.73	0.09	8.42	4.60	0.51
TAOA(aug.8, s.400)	3.01	2.03	10.86	5.22	2.35	33.52	8.26	2.61	154.51
GAT-RL(aug.8, g.)	3.02	2.37	0.04	5.26	3.14	0.11	8.37	3.98	0.53
GAT-RL(aug.8, s.400)	3.00	1.69	12.32	5.20	1.96	35.73	8.23	2.24	151.61
POMO (aug.8, g.)	2.99	1.36	0.03	5.26	3.14	0.08	8.29	2.98	0.44
POMO (aug.8, s.400)	2.97	0.68	9.18	5.18	1.57	30.12	8.22	2.11	142.14
MD-MTA (aug.10, g.)	2.99	1.36	0.07	5.21	2.16	0.19	8.20	1.86	0.78
MD-MTA (aug.10, s.200)	2.96	0.34	9.88	5.14	0.78	28.60	8.13	0.99	147.24
HRF(aug.10, g.)	2.96	0.34	0.06	5.12	0.39	0.17	8.10	0.62	0.98
HRF(aug.10, s.200)	2.95	**0.00**	10.02	5.10	**0.00**	29.58	8.05	**0.00**	145.62

be obtained for each instance, even if the inference is repeated multiple times. We denote the NCO method with the greedy strategy as 'g.'.

Sampling Strategy: This strategy samples a node according to its visiting probability, meaning that different solutions can be obtained on each inference. We denote the NCO method with the sampling strategy as 's.', and 's.400' indicates the case where 400 solutions are sampled per (augmented) instance.

Augmentation Strategy: By augmenting the original instance, multiple distinct instances are generated, enriching the solution space. In this paper, the augmentation is achieved through depot rotation augmentation [10] and coordinate rotation augmentation [17]. We use 'aug.8' to denote the DRL method featuring coordinate rotation augmentation, and 'aug.10' to refer to the NCO method with augmentation schemes applied to MDVRP-3D, where 3D represents three depots.

V. Experimental Results

A. Experimental Setting

In this section, we present a comprehensive experimental evaluation to validate the effectiveness of HRF. Our experiments are designed to assess three key aspects: performance superiority compared to baseline methods, generalization capacity across different problem settings, and component contribution through detailed ablation studies. We evaluate all competing methods on synthetic instances of MDVRP and MDOVRP, each configured with three depots. We report three primary metrics: mean objective value (Obj.), average optimality gap (Gap), and total computational time in minutes (Time (m)).

The experimental setup ensures practical feasibility by limiting all methods to a maximum of 60 seconds per instance. The evaluations are conducted on an Intel i9-14900HX CPU for CPU-based baselines and a single RTX 4090D GPU for NCO methods. For the hyperparameter configuration, the embedding dimension is set to 256, with six encoder layers. The query, key, and value dimensions are each set to 16, distributed across 16 attention heads. We employ a $\tanh$-clip value of 10 and an FF hidden dimension of 512. Training spans 1,000 epochs with 10,000 instances per epoch, using synthetic datasets with varying customer sizes ($\{20, 50, 100\}$) and three depots. Vehicle capacities are fixed at 50, with customer demands

randomly sampled from {1, 2, ..., 10}. The Adam optimizer is employed for all NCO methods, with training time dependent on problem size.

The proposed HRF is compared against a diverse set of baselines, categorized into three groups: exact solvers, meta-heuristics, and NCO methods.

- **Exact Solvers:** OR-Tools [21] and Gurobi [22] methods guarantee optimal solutions for the problem, serving as the gold standard for performance comparison.
- **Meta-Heuristic Approaches:** Hybrid Genetic Algorithm (HGA) [23] and Variable Neighborhood Tabu Search (VTNS) [24] methods provide good approximations to the optimal solution, particularly for large-scale problems where exact solvers may be computationally expensive.
- **NCO Methods:** POMO [17], TAOA [18] , GAT-RL [19], MD-MTA [10] methods leverage DRL and deep neural networks for sequential decision-making and optimization.

For a fair comparison, all baselines adhere to their recommended parameter settings. The goal is to assess HRF's ability to outperform these various methods in terms of solution quality, computational efficiency, and generalization across different problem sizes.

B. Experimental Results on Synthetic Datasets

Table I presents the outcomes for MDVRP instances with 20, 50, and 100 customers. We first observe that Gurobi under unlimited time is only tractable for the smallest problem size (20C). When a strict time limit of 60 seconds is imposed, Gurobi's performance deteriorates substantially as the instance size grows. Meanwhile, OR-Tools, HGA, and VTNS run to completion but either exceed computational budgets or produce solutions with relatively larger gaps.

By contrast, NCO methods (TAOA, GAT-RL, POMO, MD-MTA, and our HRF) exhibit significantly lower running times while achieving higher-quality solutions. For instance, comparing MD-MTA (g.) with POMO (g.) indicates that MD-MTA obtains smaller gaps at a comparable or lower computational cost. Similarly, MD-MTA (s.200) further improves solution quality by sampling additional solutions, consistently outperforming traditional baselines across all tested sizes while slightly underperforming Gurobi on the smallest 20C instances. This is not unexpected, since exact methods can be effective for small-scale cases but encounter scalability issues on larger problems.

Focusing on HRF, we find that even with a greedy rollout, the model obtains notably reduced gaps for 50C and 100C scenarios. When allowed to sample additional solutions, it achieves the best performance in nearly all cases, often beating MD-MTA and other NCO solvers by a clear margin. These results confirm that explicitly modeling heterogeneous relationships significantly improves routing efficacy and scalability.

Table II summarizes outcomes for the MDOVRP variants. Across all problem sizes, NCO methods maintain a substantial advantage in both solution quality and time. The proposed HRF emerges as the most robust approach: HRF (aug.10, g.) and HRF (aug.10, s.200) consistently yield lower objective values and narrower gaps than other strategies, verifying that the hyper-relational fusion concept generalizes effectively to open-route configurations.

TABLE III: The Training Time (h) of NCO Methods on MDVRP and MDOVRP.

Method	**20C3D**	**50C3D**	**100C3D**
POMO (MDVRP)	2.47	5.13	12.04
MD-MTA (MDVRP)	3.64	7.81	19.24
HRF (MDVRP)	4.73	10.15	24.85
POMO (MDOVRP)	1.98	4.59	10.53
MD-MTA (MDOVRP)	3.36	6.97	17.89
HRF (MDOVRP)	4.25	9.16	24.05

TABLE IV: Ablation Study on HRF.

Modules	**Augment**	**MDVRP**		**MDOVRP**	
		Obj.	**Gap (%)**	**Obj.**	**Gap (%)**
Base Model	w/o	12.58	3.11	8.49	3.29
+HeGNN Only	w/o	12.35	1.49	8.36	1.72
+HoGNN Only	w/o	12.38	1.74	8.33	1.34
HRF	w/o	12.22	0.65	8.26	0.49
HRF (Full)	aug.10	**12.14**	**0.00**	**8.22**	**0.00**

Table III summarizes the training time for MDVRP and MDOVRP. Although HRF invests up to about 30% more training time than other NCO methods on large-scale instances, it benefits from advanced fusion operations that enable more expressive representations of the routing problem. Consequently, this additional computational effort is a reasonable trade-off, as it yields superior route optimization performance and stronger generalization capabilities, justifying the extra training overhead for complex multi-depot scenarios.

C. Ablation Studies on Hyper-Relational Encoding and Augmentation

We conduct a systematic ablation analysis to elucidate the contributions of the key innovations in our HRF framework, namely HeGNN, HoGNN, and rotation-based instance augmentation. To isolate their individual and combined effects on routing performance, we train and evaluate a series of model variants under an identical optimization schedule and dataset configuration for both MDVRP and MDOVRP of the same size.

We begin by constructing a base model that omits both HeGNN and HoGNN, thereby retaining only the self-attention encoder and decoder for node feature processing. Next, we activate the HeGNN alone (to capture heterogeneity in depot–customer *vs.* customer–customer links) or the HoGNN alone (to leverage uniform adjacency), thus assessing the standalone impact of each module on representation learning. We then combine both HeGNN and HoGNN to form a *Dual Graph Encoder*, although we do not yet employ rotation-based augmentation. Finally, we incorporate depot and coordinate rotation (*aug.10*), distributing the same total of 2,000 sampled solutions across the augmented variants to complete the HRF

TABLE V: The result of generalization experiments on MDVRP.

Ins.	Dep.	Cus.	BKS	HRF (aug.9-12, s.200)				MD-MTA (aug.9-12, s.200)				POMO (aug.8, s.400)				GAT-RL (aug.8, s.400)			
				B.O.	B.G. (%)	A.O.	A.G. (%)	B.O.	B.G. (%)	A.O.	A.G. (%)	B.O.	B.G. (%)	A.O.	A.G. (%)	B.O.	B.G. (%)	A.O.	A.G. (%)
p01	4	50	577	577	**0.00**	579	0.35	577	**0.00**	581	0.68	581	0.69	589	2.06	584	1.30	591	2.52
p02	4	50	474	474	**0.00**	474	**0.00**	474	**0.00**	475	0.24	477	0.63	478	0.94	482	1.84	486	2.69
p03	5	75	641	641	**0.00**	648	1.09	641	**0.00**	650	1.33	648	1.09	651	3.09	652	1.73	663	3.44
p04	2	100	1001	1001	**0.00**	1011	1.00	1001	**0.00**	1013	1.17	1011	1.00	1020	3.85	1017	1.56	1040	3.85
p05	2	100	750	750	**0.00**	754	0.53	750	**0.00**	755	0.66	760	1.33	771	2.85	767	2.32	772	2.98
p06	3	100	877	877	**0.00**	878	0.11	877	**0.00**	879	0.33	883	0.68	894	1.97	887	1.24	897	2.38
p07	4	100	882	882	**0.00**	884	0.23	882	**0.00**	885	0.31	897	1.70	911	3.29	905	2.59	913	3.49
p08	2	249	4370	4450	1.83	4580	4.81	4467	2.22	4599	5.25	4587	4.97	4721	8.02	4661	6.67	4796	9.76
p09	3	249	3859	3901	1.09	4001	3.68	3919	1.55	4026	4.33	3985	3.27	4124	6.88	4064	5.33	4196	8.75
p10	4	249	3630	3759	3.55	3851	6.08	3770	3.86	3874	6.72	3820	5.23	4008	10.43	3934	8.39	4079	12.39
p11	5	249	3545	3685	3.95	3802	7.25	3692	4.15	3816	7.63	3723	5.02	3912	10.34	3903	10.09	4053	14.32
Average			1873	1909	0.95	1951	2.28	1302	1.04	1330	2.39	1334	2.72	1359	4.06	1361	4.32	1387	6.06

TABLE VI: The result of generalization experiments on MDOVRP.

Ins.	Dep.	Cus.	BKS	HRF (aug.9-12, s.200)				MD-MTA (aug.9-12, s.200)				POMO (aug.8, s.400)				GAT-RL (aug.8, s.400)			
				B.O.	B.G. (%)	A.O.	A.G. (%)	B.O.	B.G. (%)	A.O.	A.G. (%)	B.O.	B.G. (%)	A.O.	A.G. (%)	B.O.	B.G. (%)	A.O.	A.G. (%)
p01	4	50	386	386	**0.00**	387	0.26	386	**0.00**	388	0.35	387	0.22	388	0.48	391	1.31	396	2.61
p02	4	50	376	376	**0.00**	377	0.27	376	**0.00**	378	0.60	377	0.39	379	0.92	384	2.03	390	3.63
p03	5	75	475	475	**0.00**	478	0.63	475	**0.00**	481	1.38	480	1.06	484	1.91	485	2.28	489	3.12
p04	2	100	662	662	**0.00**	668	0.91	662	**0.00**	670	1.16	668	0.93	672	1.53	679	2.60	689	4.11
p05	2	100	608	608	**0.00**	608	0.00	608	**0.00**	608	0.06	618	1.78	618	1.78	622	2.37	634	4.31
p06	3	100	612	612	**0.00**	612	0.00	612	**0.00**	612	0.08	619	1.18	622	2.65	624	1.94	628	2.60
p07	4	100	608	608	**0.00**	609	0.16	608	**0.00**	610	0.26	613	0.78	621	2.09	617	1.51	630	3.65
p08	2	249	2776	2828	1.87	2900	4.46	2853	2.76	2940	5.90	2921	5.23	3024	8.94	2993	7.81	3064	10.39
p09	3	249	2578	2621	1.67	2650	2.79	2638	2.32	2676	3.79	2710	5.11	2720	5.49	2746	6.51	2813	9.11
p10	4	249	2482	2522	1.61	2610	5.16	2550	2.73	2636	6.18	2648	6.67	2720	9.58	2714	9.34	2763	11.31
p11	5	249	2468	2537	2.83	2600	5.35	2558	3.64	2628	6.48	2632	6.63	2676	8.45	2712	9.34	2760	11.81
Average			1276	1298	1.72	1318	1.82	1302	2.04	1330	2.39	1334	2.72	1359	4.06	1361	4.32	1387	6.06

pipeline. For example, a model using '*aug.10*" allocates 200 solutions per augmented variant, whereas a model without augmentation (*aug.none*) applies all 2000 samples to a single setup.

Table IV details the mean objective (*Obj.*) and the percentage gap (*Gap*) from the best-performing configuration for both MDVRP and MDOVRP. As observed, introducing HeGNN or HoGNN individually reduces the mean route cost relative to the base model, underscoring the importance of modeling heterogeneous edges and global connectivity. When both modules are enabled simultaneously, the model attains further gains, implying that hyper-relation features and homogeneous adjacency cues offer complementary benefits. Finally, once depot and coordinate rotation are added, the network achieves the best results on both problem variants, confirming that rotation-based augmentation can further refine solutions by mitigating local optima and enhancing search diversity.

D. Generalization on Benchmark Datasets

In this subsection, we examine the generalization capability of our proposed HRF model on well-known benchmark datasets for MDVRP. We adopt the same experimental setup used by other NCO baselines (i.e., POMO, GAT-RL, and TAOA) and follow previous works [17]–[19], [25], [26] to select 11 benchmark instances, denoted by {p01, p02, p03, p04, p05, p06, p07, p08, p09, p10, p11}. Specifically, {p01, p02, p03, p04, p05, p06, p07} are derived from the Christofides and Eilon dataset [25], while {p08, p09, p10, p11} are from Gillett and Miller [26]. Each NCO method is trained only on a 100C3D synthetic dataset but then evaluated 10 times on each benchmark instance to assess its Out-of-Distribution (OOD) performance. We evaluate both the Best Objective (B.O.) and the Average Objective (A.O.) across 10 independent runs, accompanied by their corresponding gaps to the Best-Known Solutions (BKS) [27]: the Best Gap (B.G.) for B.O. and the Average Gap (A.G.) for A.O., expressed as percentage deviations. Larger instances, such as p08–p11, involve 249 customers and pose a more challenging distribution shift since they deviate significantly from the synthetic training set size of 100 customers.

Table V presents the generalization results on MDVRP. For a fair comparison, all methods use comparable solving time (s.{200 or 400} steps) during inference. Notably, our HRF (aug.9-12, s.200) consistently attains lower best and average objective gaps compared to other NCO baselines, demonstrating superior OOD robustness across all benchmark instances. Even on the largest-scale instances (e.g., p08–p11), HRF maintains competitive quality, indicating that fusion mechanisms effectively capture global relationships and reduce performance degradation when encountering substantially different problem scales.

Table VI presents the results of MDOVRP, where similar patterns are observed. Our HRF demonstrates strong performance on MDOVRP.

VI. Conclusion and Future Work

In this paper, we proposed a novel HRF neural solver for MDVRP. By leveraging two specialized GNNs (HoGNN and HeGNN), our method captures both homogeneous spatial patterns and heterogeneous relational complexities. The fusion of these diverse features results in enhanced representations that improve solution quality and efficiency. Experimental results demonstrate that the proposed HRF mechanism outperforms baseline methods, achieving better performance on a variety of

MDVRP instances. Future work will explore dynamic routing scenarios, time-window constraints, and hybrid methods that combine learned policies with classical optimization techniques.

References

[1] J. Renaud, G. Laporte, and F. F. Boctor, "A tabu search heuristic for the multi-depot vehicle routing problem," *Computers & Operations Research*, vol. 23, no. 3, pp. 229–235, 1996.

[2] P. Repoussis, C. Tarantilis, O. Bräysy, and G. Ioannou, "A hybrid evolution strategy for the open vehicle routing problem," *Computers & Operations Research*, vol. 37, no. 3, pp. 443–455, 2010, hybrid Metaheuristics. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0305054808002207

[3] C. Contardo and R. Martinelli, "A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints," *Discrete Optimization*, vol. 12, pp. 129–146, 2014.

[4] W. Ho, G. T. Ho, P. Ji, and H. C. Lau, "A hybrid genetic algorithm for the multi-depot vehicle routing problem," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 4, pp. 548–557, 2008.

[5] F. W. Glover and É. D. Taillard, "A user's guide to tabu search," *Annals of Operations Research*, vol. 41, pp. 1–28, 1993.

[6] Y. Bengio, A. Lodi, and A. Prouvost, "Machine learning for combinatorial optimization: a methodological tour d'horizon," *European Journal of Operational Research*, vol. 290, no. 2, pp. 405–421, 2021.

[7] W. Kool, H. van Hoof, and M. Welling, "Attention, learn to solve routing problems!" 2019.

[8] M. Nazari, A. Oroojlooy, M. Takáč, and L. V. Snyder, "Reinforcement learning for solving the vehicle routing problem," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPS'18. Red Hook, NY, USA: Curran Associates Inc., 2018, p. 9861–9871.

[9] A. Arishi and K. Krishnan, "A multi-agent deep reinforcement learning approach for solving the multi-depot vehicle routing problem," *Journal of Management Analytics*, vol. 10, no. 3, pp. 493–515, 2023.

[10] J. Li, B. Tian Dai, Y. Niu, J. Xiao, and Y. Wu, "Multi-type attention for solving multi-depot vehicle routing problems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 17 831–17 840, 2024.

[11] B. Liu, Q. Li, F. Liu, and B. Zhu, "Hybrid genetic algorithm for the multi-depot vehicle routing problem with the same distribution center of 3pl enterprises," *International Journal of Computer Integrated Manufacturing*, vol. 22, no. 4, pp. 308–319, 2009.

[12] R. Soto, M. Casazza, and F. Tohmé, "Multiple neighborhood search coupled with a tabu search for the multi-depot open vehicle routing problem," *Annals of Operations Research*, vol. 259, no. 1, pp. 241–265, 2017.

[13] X. Wang, Y. Sadati, and T. Lei, "A variable tabu neighborhood search algorithm for the multi-depot vehicle routing problem with time windows," *International Journal of Production Research*, vol. 54, no. 2, pp. 485–501, 2016.

[14] I. Bello, H. Pham, Q. V. Le, M. Norouzi, and S. Bengio, "Neural combinatorial optimization with reinforcement learning," *ArXiv*, vol. abs/1611.09940, 2016.

[15] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," in *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'15. Cambridge, MA, USA: MIT Press, 2015, p. 2692–2700.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[17] Y.-D. Kwon, J. Choo, B. Kim, I. Yoon, Y. Gwon, and S. Min, "Pomo: Policy optimization with multiple optima for reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21 188–21 198.

[18] Y. Zou, H. Wu, Y. Yin, L. Dhamotharan, D. Chen, and A. K. Tiwari, "An improved transformer model with multi-head attention and attention to attention for low-carbon multi-depot vehicle routing problem," *Ann. Oper. Res.*, pp. 1–20, Jun. 2022.

[19] K. Zhang, X. Lin, and M. Li, "Graph attention reinforcement learning with flexible matching policies for multi-depot vehicle routing problems," *Physica A: Statistical Mechanics and its Applications*, vol. 611, p. 128451, Feb. 2023.

[20] W. Kool, H. van Hoof, and M. Welling, "Attention, learn to solve routing problems!" in *International Conference on Learning Representations (ICLR)*, 2018.

[21] S. Kruk, *Practical Python AI Projects: Mathematical Models of Optimization Problems with Google OR-Tools*. Berlin, Germany: Springer, 2018.

[22] L. G. Optimization, *Gurobi Optimizer Reference Manual*, Gurobi Optimization, LLC, Beaverton, OR, USA, 2020.

[23] R. Liu, Z. Jiang, and N. Geng, "A hybrid genetic algorithm for the multi-depot open vehicle routing problem," *OR Spectr.*, vol. 36, no. 2, pp. 401–421, Mar. 2014.

[24] M. E. H. Sadati, B. Çatay, and D. Aksen, "An efficient variable neighborhood search with tabu shaking for a class of multi-depot vehicle routing problems," *Comput. Oper. Res.*, vol. 133, Sep. 2021.

[25] N. Christofides and S. Eilon, "An algorithm for the vehicle-dispatching problem," *OR*, vol. 20, no. 3, pp. 309–318, 1969. [Online]. Available: http://www.jstor.org/stable/3008733

[26] B. E. Gillett and L. R. Miller, "A heuristic algorithm for the vehicle-dispatch problem," *Operations Research*, vol. 22, no. 2, pp. 340–349, 1974. [Online]. Available: http://www.jstor.org/stable/169591

[27] M. E. H. Sadati, B. Çatay, and D. Aksen, "An efficient variable neighborhood search with tabu shaking for a class of multi-depot vehicle routing problems," *Computers & Operations Research*, vol. 133, September 2021.