2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

SmartPathfinder: Pushing the Limits of Heuristic Solutions for Vehicle Routing Problem with Drones Using Reinforcement Learning

Navid Mohammad Imran and Myounggyu Won

***Abstract*— The Vehicle Routing Problem with Drones (VRPD) seeks to optimize the routing paths for both trucks and drones, where the trucks are responsible for delivering parcels to customer locations, and the drones are dispatched from these trucks for parcel delivery, subsequently being retrieved by the trucks. Given the NP-Hard complexity of VRPD, numerous heuristic approaches have been introduced. However, improving solution quality, the definition of which can vary depending on various heuristic approaches, *e.g.,* the total operation time, remain significant challenges. In this paper, we conduct a comprehensive examination of heuristic methods designed for solving VRPD, distilling and standardizing them into core elements. We then develop a novel reinforcement learning (RL) framework that is seamlessly integrated with the heuristic solution components, establishing a set of universal principles for incorporating the RL framework with heuristic strategies in an aim to improve both the solution quality and computation speed, regardless of how the solution quality is defined. This integration has been applied to a state-of-the-art heuristic solution for VRPD, showcasing the substantial benefits of incorporating the RL framework. Our evaluation results demonstrated that the heuristic solution incorporated with our RL framework not only elevated the quality of solutions but also achieved rapid computation speeds, especially when dealing with extensive customer locations.**

I. Introduction

The adoption of drone delivery systems has attracted considerable attention due to its numerous benefits. Drones transcend traditional delivery barriers, offering freedom from the constraints of road networks and avoiding traffic congestion, thus enhancing efficiency and reliability in parcel delivery [1], [2]. Furthermore, drone delivery has potential to significantly reduce CO2 emissions, aligning with global sustainability goals [3], [4]. Additionally, the implementation of drones in delivery systems can lead to substantial cost reductions, primarily through the elimination of the need for truck drivers, which contributes to financial savings for logistics companies [5], [6].

These compelling advantages have prompted a wave of investment from logistics companies into drone technology, with several pioneering initiatives underscoring the growing commitment to this innovative delivery method. Notably, Amazon unveiled its Amazon Prime Air service in late 2013, marking a significant milestone in the commercial application of drones for delivery [7]. Alibaba has been experimenting with drone deliveries since 2015, demonstrating the feasibility and value of integrating drones into existing logistics frameworks [8]. In response to the COVID-19 pandemic, UPS initiated a drone delivery service in a Florida retirement community in 2020, showcasing the technology's potential in addressing urgent healthcare needs [9]. Additionally, DHL made headlines in 2019 by launching its first regular, fully-automated, and intelligent urban drone delivery service, setting a new standard for urban logistics solutions [10].

Navid Mohammad Imran and Myounggyu Won are with the Department of Computer Science, University of Memphis, Memphis, TN, United States {nimran, mwon}@memphis.edu

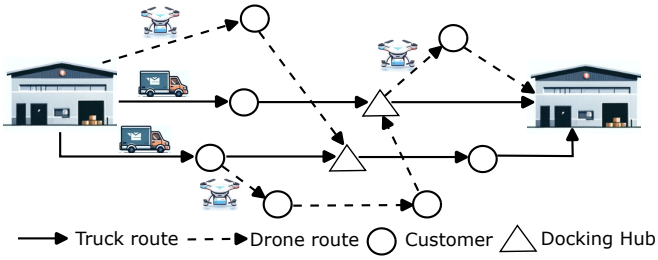

Fig. 1: An overview of a truck-drone operation in the vehicle routing problem with drones (VRPD). Trucks and drones collaborate to distribute parcels to customers, with the docking hubs serving as the retrieval points for trucks to collect returning drones.

The Vehicle Routing Problem with Drones (VRPD) represents an evolution of the traditional Vehicle Routing Problem (VRP), incorporating both trucks and drones into its operational framework [11], [12]. This novel problem was initially introduced by Wang and Sheu [13], distinguishing itself from VRP by utilizing two distinct modes of delivery: trucks, which not only serve customers but also launch drones, and the drones, which are deployed for deliveries from trucks (drones can also be deployed from depots), and subsequently retrieved at designated docking hubs [13]. Drones offer the flexibility of being launched and retrieved multiple times at various times and locations. The essence of VRPD lies in optimizing both the delivery routes for trucks and drones as well as the strategic selection of launch and retrieval points for drones, with the goal of minimizing the total operational costs.

Numerous solutions have been developed to tackle VRPD and its variants, each considering a variety of critical factors, including the speed of drones [14], the number of drones [15], drone energy consumption [16], and the impact on carbon emissions [17], among others. However, a notable limitation persists: most of these solutions resort to heuristic or meta-heuristic approaches due to the computational complexity stemming from the NP-Hard nature of the prob-

lem. Consequently, these solutions are often constrained to handling only a small number of instances, *i.e.,* the number of customer locations. Moreover, the stochastic nature of heuristic methods, which rely on random solution generation and modification, poses a significant challenge in ensuring high solution quality.

To address the limitations inherent in current heuristic methodologies for VRPD, our approach begins with a thorough assessment of these heuristics found in the literature. We systematically deconstruct these algorithms into universal components. Building upon the dissection of heuristic algorithms, we present SmartPathfinder that provides standardized guidelines for integrating a Reinforcement Learning (RL) framework with the fundamental components of heuristic algorithms aiming to enhance both computational efficiency and solution quality. It should be noted that the definition of 'solution quality' can vary across different approaches. The advantage of our method is its ability to enhance solution quality regardless of how it is defined, meaning our approach can be universally applied to various heuristic solutions designed to optimize the solution defined differently.

More specifically, we intertwine the solution space of heuristic algorithms with the decision-making process of an RL agent. This agent explores the solution space under the guidance of a novel reward function designed to optimize both solution quality and computation speed simultaneously, facilitating the computation of solutions for large-scale problems with extensive customer locations. Additionally, our approach incorporates a mechanism for 'solution escape', empowering the system to circumvent potential local optima, thereby refining the resilience and efficacy of the novel RL-heuristic algorithm integration. The outcomes of our integration, pairing the RL framework with a state-of-the-art heuristic algorithm for VRPD, demonstrate notable improvements in solution quality and computation speed by up to 28.4% and 27.3%, respectively, in comparison with the heuristic algorithm without RL integration. In summary, the contributions of our work can be summarized as follows.

- We perform an in-depth examination of heuristic solutions for the Vehicle Routing Problem with Drones (VRPD), breaking them down into their fundamental components for seamless integration with the reinforcement learning (RL) framework.
- We introduce novel guidelines for the integration of a RL framework within VRPD heuristic solutions.
- We implemented the RL integration for a state-of-the-art VRPD solution, showcasing our methodology's applicability and flexibility.
- Through a comprehensive computational study, we highlight the significant advantages of our RL-enhanced solution for VRPD, particularly in terms of computational efficiency and the superior quality of the solutions generated.

This paper is organized as follows. In Section II, we present a literature review on various solutions for VRPD and its variants. In Section III, we introduce the precise definition of VRPD and conduct a thorough analysis of existing heuristic approaches for VRPD, setting the stage for explaining how the RL framework is integrated with a heuristic solution. In Section IV, we present the detailed RL framework design and how it can be incorporated with a heuristic VRPD solution. We then present the evaluation results in Section V and conclude in Section VI.

II. Related Work

The goal of VRPD is to optimize the delivery routes for a fleet of trucks and drones, ensuring they operate in a synergistic manner to efficiently distribute parcels to customers [18]. Specifically, this involves the strategic deployment of drones from either the trucks or depots, enabling them to directly deliver parcels to customers. Upon the completion of their delivery, these drones are designed to reunite with the trucks at predetermined rendezvous points. This section delves into a comprehensive review of state-of-the-art approaches developed for tackling VRPD and its diverse variations, particularly focusing on papers published within five years.

Zhou *et al.* explored a unique variation of VRPD, focusing on the optimization of the number of drones assigned to each vehicle, a contrast to the conventional approach of static drone allocation [15]. To address this problem, they employed a branch-and-price algorithm with a tabu search strategy. Imran *et al.* marked a pioneering step towards the realization of a drone-based parcel delivery ecosystem, seamlessly incorporating autonomous vehicles (AV) into the VRPD framework [12]. Their approach seeks to minimize operational expenses by adeptly selecting AVs from a pool of available AVs through a crowd-sourced methodology. This strategy enables the dynamic assignment of AVs to specific customers, further refining the efficiency of route planning by leveraging real-time traffic data. Both Mixed Integer Linear Programming (MILP) and a greedy algorithm were designed to solve their problem.

Tamke *et al.* considered a variation of VRPD, dubbed the Vehicle Routing Problem with Drones and Drone Speed Selection (VRPD-DSS) [14]. Their model prioritizes the minimization of operational expenses while accommodating the selection of discrete speed settings for drones. Their findings reveal that adjusting drone speeds according to specific delivery requirements offers significant cost reductions over traditional methodologies that do not account for speed variability. Xia *et al.* enhanced the VRPD framework by introducing drone stations that are mainly used for collection, storage and recharge of drones as well as replenishing parcels for both trucks and drones [16] . The drone stations serve to bolster the synergy between trucks and drones while addressing the impact of varying load weights on drone energy consumption [16]. They designed an advanced branch-and-price-and-cut algorithm taking into account the drone stations.

The authors of [17] delve into a variant of VRPD with dual objectives of minimizing delivery times and reducing carbon

emissions. Leveraging the Non-Dominated Sorting Genetic Algorithm II (NSGA-II), their approach adeptly balances these objectives. Their findings demonstrated the significant potential of drones to enhance both environmental sustainability and delivery efficiency. Meanwhile, the research presented in [19] targets the application of VRPD within the context of disaster relief, acknowledging the inherent uncertainties in demand and travel times that characterize such scenarios. By formulating an enhanced branch-and-price-and-cut (BPC) algorithm, this study uncovers efficient routing strategies that underscore the substantial advantages of integrating trucks with drones over traditional truck-only methods, suggesting its potential for real-world disaster response.

Mara *et al.* [23] introduced the Electric Vehicle-Drone Routing Problem (EVDRP), where drones collaborate with electric vehicles for last-mile deliveries. They minimize the total completion time, considering EV battery limitations and recharging stations using the memetic algorithm with problem-specific operators to efficiently solve the problem. The authors in [28] proposed a new model for solving VRPD, allowing drones to perform multiple pickups and deliveries per flight. They develop a two-stage heuristic and show that allowing multiple visits and combined pickup/delivery significantly reduces the total cost compared to traditional approaches. Momeni *et al.* [22] proposed a novel VRPD model that minimizes delivery time while considering drone energy consumption at varying altitudes. They developed a two-phase heuristic algorithm based on nearest neighborhood and local search and demonstrated the model's effectiveness in optimizing routes, particularly for deliveries at different heights, for both small, large and real-world instances.

Previous studies have developed effective solutions for numerous variants of VRPD, with a significant emphasis on heuristic strategies owing to the NP-Hard complexity of the problem. However, the complexity of the problem limits such approaches, typically confining them to a relatively small scope of instances in terms of the number of customer locations. Additionally, the inherently probabilistic characteristic of heuristic algorithms, dependent on randomly generating and adjusting solutions, represents a notable obstacle in achieving consistently superior solution quality. In contrast, leveraging recent advances in machine learning [29], SmartPathfinder uniquely employs a novel machine learning-driven approach through seamless integration with heuristic algorithms to address the challenges.

III. Preliminaries

A. Definition of VRPD

The Vehicle Routing Problem with Drones (VRPD) is defined in a graph $G = (N, A)$, where N denotes a collection of nodes, including a depot node, customer nodes $C = \{c_1, c_2, ..., c_n\}$, and docking hub nodes $O = \{o_1, o_2, ..., o_m\}$. A denotes the arcs connecting these nodes. Vehicles and drones in this scenario are denoted by a set K and D, respectively. Drones are characterized by their constrained payload capacity of up to L^D weight units and the operational range of up to T^D. Drones benefit from the ability to navigate more direct paths, facilitating expedited deliveries to customers. Conversely, trucks can accommodate a more substantial cargo, up to L^R drones and L^T weight units of parcels, albeit they are subject to the limitations imposed by the road network. A parcel destined for a customer $i \in C$, weighing g_i, must not exceed L^T, *i.e.,* it qualifies for drone delivery if $g_i \leq L^D$. The travel times for trucks and drones from node $i \in N$ to $j \in N$ are denoted by t^T_{ij} and t^D_{ij} respectively, with drone travel time t^D_{ij} invariably being less than that of trucks t^T_{ij}, showcasing their advantage in speed and efficiency.

Drones may serve customers independently within their flight range, or be transported by trucks to customers located outside this range. It is assumed that the battery-swapping and drone-loading times are negligible. Additionally, the following variables are defined to set up the objective function.

- x_{ijk}: Equals 1 if the k-th truck travels arc $(i, j) \in A$ independently, and 0 otherwise.
- y_{ijd}: Equals 1 if the d-th drone travels arc $(i, j) \in A$ independently, and 0 otherwise.
- u_{ijk}: Equals 1 as long as the k-th truck carries one or more drones through arc $(i, j) \in A$, and 0 otherwise.

The objective of VRPD is to minimize the total operation cost, comprised of the fixed truck employment cost F^T and the variable transportation costs for both trucks C^T and drones C^D, per unit travel time which is defined as follows.

$$\min \quad \Bigg[F^T \Bigg(\sum_{(i,j)\in A} \sum_{k\in K} x_{ijk} + \sum_{(i,j)\in A} \sum_{k\in K} u_{ijk} \Bigg) + C^T \sum_{(i,j)\in A} \sum_{k\in K} t^T_{ij}(x_{ijk} + u_{ijk}) + C^D \sum_{(i,j)\in A} \sum_{d\in D} t^D_{ij} y_{ijd} \Bigg]. \tag{1}$$

The objective function (Eq 1) is subject to various constraints to ensure feasibility and efficiency in operations involving trucks and drones. Key among these constraints are two that guarantee both trucks and drones return to their starting depot. Another pair of constraints ensures that each customer node is visited exactly once, meaning there's only one incoming and outgoing connection for each node. Additionally, at every docking station, the number of arriving and departing drones is kept equal to maintain balance. To prevent overloading, two constraints limit the truck from carrying more drones than its capacity allows. Moreover, three specific constraints are in place to regulate the maximum flight duration of drones, ensuring their operations are within feasible limits. There are also five more constraints focusing on the parcel-carrying capacities of both drones and trucks to prevent overloading. Detailed mathematical formulations of these constraints are available in [13].

B. Analysis of Heuristic Algorithms for VRPD

In this section, we perform an analysis of heuristic methodologies applied to solving VRPD, aiming to decom-

TABLE I: Decomposition of heuristic algorithms for VRPD into four universal components.

Papers	Heuristic	Initialization	Solution Modification	Shuffling	Solution Evaluation
[20]	Variable Neighborhood Search (VNS)	✓	✓	✓	✓
[21] [22]	Largest/Nearest Neighborhood Search (LNS/NNS)	✓	✓		✓
[23] [17]	Genetic/Memetic Algorithm (GA/MA)	✓	✓	✓	✓
[24]	Ant Colony Optimization (ACO)	✓	✓		✓
[25] [26]	Artificial Bee Colony (ABC)	✓	✓	✓	✓
[27] [28]	Simulated Annealing (SA)	✓	✓		✓
[12]	Greedy Algorithm	✓			✓
[16] [19]	Branch-and-Price-and-Cut algorithm (BPC)	✓	✓		✓

pose these algorithms into their fundamental elements for incorporation with the RL framework. This analysis encompasses a comprehensive review of 13 papers that introduce heuristic solutions for VRPD, published between 2019 and 2023. As illustrated in Table I, despite the diverse operational mechanisms of these algorithms, our extensive analysis has identified four core components common to many heuristic approaches: Solution Initialization, Solution Modification, Solution Evaluation, and Solution Shuffling.

The Solution Initialization component is pivotal in establishing the solution structure and creating the baseline solutions upon which heuristic algorithms iteratively improve. The Solution Modification component stands as the crux of the algorithmic process, employing a tailored set of rules to iteratively refine the solution. In the Solution Evaluation component, the focus is on assessing the solution's efficacy, specifically measuring its impact on minimizing the total operational expenses incurred in servicing all customer locations. It is noteworthy that while the aforementioned components are universally present across most heuristics, the Solution Shuffling mechanism—which involves the random rearrangement of solution elements to avert stagnation at local optima—is not as uniformly integrated. This systematic breakdown of heuristic methodologies facilitates the seamless integration of the RL framework with the components of the heuristics. A detailed explanation of this integration process is explained in the subsequent section.

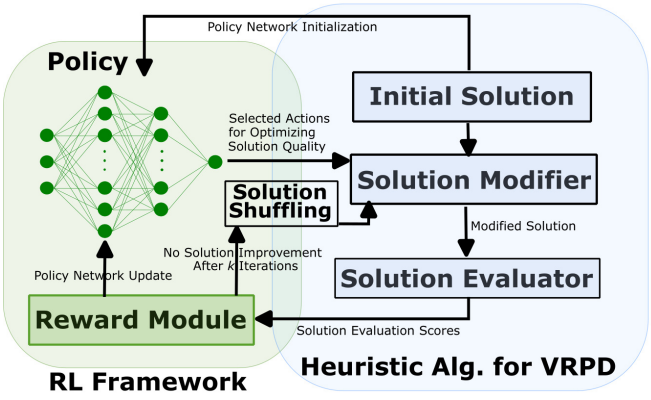

Fig. 2: An architecture of SmartPathfinder, illustrating integration with the RL framework with a heuristic algorithm for VRPD.

IV. Design of SmartPathfinder

A. Overview

Building on the universal components of heuristic algorithms for VRPD, SmartPathfinder aims to seamlessly incorporate a RL framework within the structure of a heuristic algorithm to improve both the quality of solutions and computational efficiency. Fig. 2 depicts the architecture of SmartPathfinder. As shown, it begins with an initial solution, which activates the RL framework's policy network which is essentially a multi-layer perceptron. The policy network, once initialized, outputs actions that are forwarded to the solution modifier. These actions, designed in accordance with the underlying heuristic algorithm, dictate the modification of the initial solution. Following this, the solution evaluator assesses the modified solution's efficacy. Subsequently, the evaluation scores of both the modified and the original solutions are relayed to the RL framework's reward module, where a reward is computed based on these evaluations. In particular, if the solution fails to improve after a predetermined number of iterations, the procedure diverts from policy network updating to solution shuffling. This involves rearranging the solution before subjecting it to reevaluation to avoid potential local optima. In situations where an improvement in the solution is observed, the reward module feeds the results of the evaluation back into the policy network for updates. The updated network then issues new actions aimed at further refining the solution. This cycle of solution evaluation, reward assessment, and solution modification iterates, fostering continuous improvement in the solution quality.

B. Action Space

The design of the action space is tailored to the solution modification capabilities inherent to heuristic algorithms, *i.e.,* each action represents a solution alteration method for a specific heuristic algorithm. Unlike the probabilistic solution alteration typical in heuristic algorithms [20][23][24][28], this approach advocates for a strategic selection of actions

driven by reward feedback, aiming for solutions of superior quality. For instance, key operations from a genetic algorithm (GA)-based approach to solving VRPD [23] such as the parent selection, crossover, and mutation, are integrated as specific actions within the RL framework. This integration allows for more strategic action selection, guided by the RL's policy network. Similarly, the core operations of the Neighborhood Search (NS) algorithm for VRPD [20], which include neighborhood moves essential for local search, such as node swaps, entire swaps, node insertion, whole insertion, node reversal, entire reversal, sortie removal, and sortie addition, are also adapted as actionable strategies within the RL framework. Since there are only specific actions allowed based on the inherent heuristics, the action space belongs to a discrete set.

C. State Space

The state space is a critical construct that defines all necessary information for an agent to make informed decisions regarding its actions. More specifically, the state space of SmartPathfinder includes information related to both the quality of potential solutions and the efficiency of computational processes which are real-valued parameters. Solution quality is gauged through the solution evaluation outcomes at any given time step t, denoted by $\mathcal{S}_t$. This metric is derived from the heuristic algorithm's solution evaluator, providing a snapshot of the current solution's effectiveness. Additionally, the evaluation result of the solution prior to the current one is denoted by $\mathcal{S}_{t-1}$, offering a comparative perspective on solution progression or regression. On the other hand, computational efficiency is quantified by the cumulative actions executed by the agent up to time step t, denoted by $\mathcal{A}_t$. This measurement reflects the agent's operational speed and efficiency, which are especially important in environments where computational resources or time are constrained. To navigate the trade-offs between solution quality and computational speed, the state space also integrates performance weights w_1 and w_2. These weights are utilized to balance and prioritize these performance dimensions, guiding the agent towards a more effective decision-making process.

D. Reward Function

The reward function of SmartPathfinder is designed to simultaneously enhance the solution quality and minimize computational time, embodying a dual objective within the RL framework. Furthermore, it offers configurability, allowing users to tailor the weighting between solution quality and computational efficiency according to specific needs. To quantitatively evaluate solution quality, the reward function employs the net improvement of the solution, calculated as $\mathcal{S}_t - \mathcal{S}_{t-1}$. This formulation inherently associates a negative reward with any deterioration in solution quality, incentivizing progress and penalizing regression. Also, the function prioritizes computational speed by incorporating the total number of actions executed by the agent, *i.e.,* by assigning a negative reward as the count of actions increases. Consequently, with these considerations, the reward function R_t at time step t is defined as follows.

$$R_t = w_1(\mathcal{S}_t - \mathcal{S}_{t-1}) - w_2\mathcal{A}_t, \tag{2}$$

where w_1 is the weighting parameter for the solution quality, w_2 is the weighting parameter for the computational time, and $\mathcal{A}_t$ is the total number actions taken. Through numerous simulations, these parameters are fixed at $w_1 = 3$ and $w_2 = 2$. However, it should be noted that the parameters can be configured differently depending on the heuristic approach used, as the definition of 'solution quality' may vary.

V. Evaluation Results

A. Evaluation Setup

To evaluate the performance of SmartPathfinder, we implemented the integration of the RL framework with a state-of-the-art heuristic algorithm designed based on the memetic algorithm [23]. It is important to note that changing the heuristic algorithm also requires a retraining of the RL network. Fig. 3 demonstrates the convergence of the reward value for SmartPathfinder integrated with the heuristic algorithm [23] after training for 1000 episodes. Additionally, for a more effective performance comparison, we implemented another heuristic algorithm based on neighborhood search [20]. Throughout this section, the memetic algorithm-based solution is referred to as MA, the neighborhood search-based approach as NS, and the RL-enhanced method as RL+MA. All implementation was performed on a computer equipped with an AMD Ryzen 7 7840HS CPU, 16GB RAM, NVIDIA GeForce RTX 4050 GPU, and Windows 11. The code was written in Python 3.10.

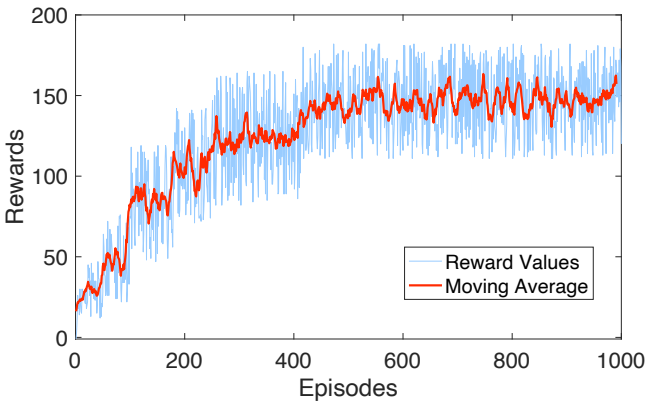

Fig. 3: The reward values of the memetic algorithm-based heuristic solution integrated with the RL framework, illustrating the convergence of the reward value.

Our evaluation hinges on two primary metrics: solution quality and computational efficiency. The quality of the solution is measured by the total operational time, which is the total amount of time needed to serve all customer locations. This measure has been widely adopted by numerous heuristic algorithms for VRPD for performance evaluation [30], [17], [18]. Computational efficiency, on the other hand, is gauged

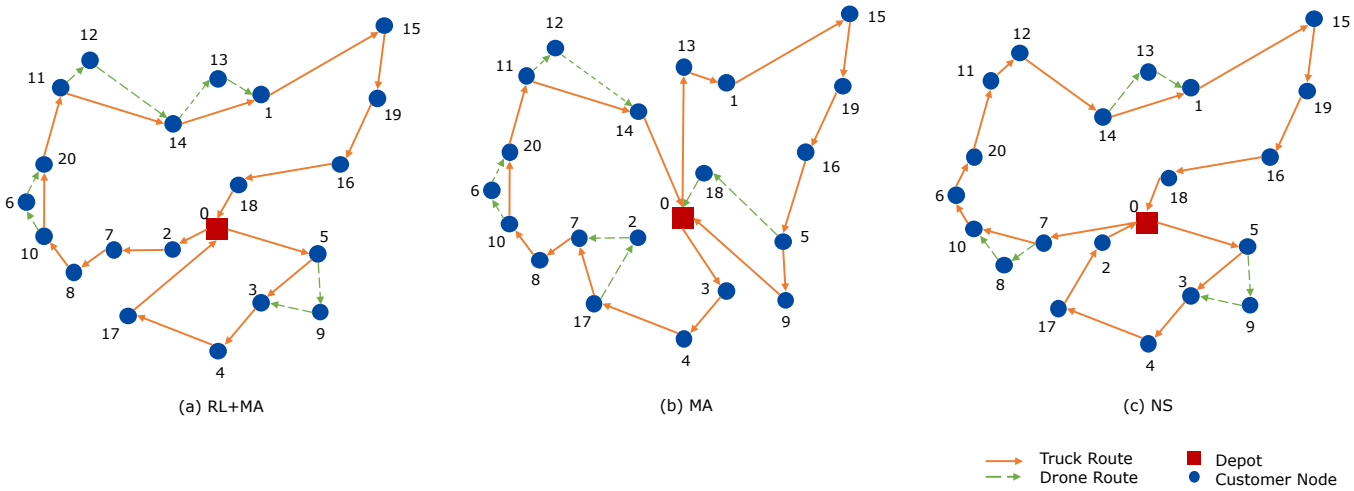

Fig. 4: An example solution generated with (a) RL+MA, (b) MA, and (c) NS. The integration of the RL framework with MA results in more efficient paths for both trucks and drones compared with MA and NS.

by the elapsed time from the initiation of the algorithm to the moment the final solution is derived. We evaluate the performance of our proposed solution across various scenarios, characterized by differing numbers of customer nodes by up to 100 customer nodes. For distance calculations, we adopt the Manhattan distance metric for truck movements and the Euclidean distance for drone operations, similar to the methods used in many VRPD heuristic solutions [23], [28], [21]. We adopted a dataset from [20] for performance evaluation. The data set comprises of 72 test instances with varying numbers of customer nodes. For fair performance comparison, we excluded the customer time windows from the dataset.

In configuring the memetic algorithm, we adhere to the parameter settings detailed by Mara *et al.* [23]. In particular, our evaluation method does not assume electric vehicles (EVs); hence, we employ the same neighborhood move strategies as those outlined in Kuo *et al.* [20] and Mara *et al.* [23], with the exception of EV-specific maneuvers such as recharging insertion, change station, moving station, and remove station.

B. Solution Quality

We compare the total operational time across three different strategies: RL+MA, MA, and NS. The total operational time was measured by varying the customer node counts, with all customer nodes being randomly deployed within the target area. The findings, illustrated in Fig. 5, demonstrate that RL+MA outperforms MA and NS, particularly as the complexity of the problem increases with more customer nodes. More specifically, initially, when the number of customer nodes is low, the distinction in performance between the RL-enhanced strategy and conventional methods is minimal. This trend is attributed to the constrained solution space, which allows all evaluated algorithms to relatively easily identify high-quality solutions. Nonetheless, the advantage of integrating RL becomes increasingly apparent with the growth in customer numbers. Specifically, for scenarios involving 100 customer nodes, the RL-enhanced strategy reduces the total operational time by up to 23.7% compared to MA, and by 28.4% relative to NS. These results highlight the significant benefits of incorporating RL into heuristic algorithms for tackling VRPD, especially in more demanding scenarios with a larger customer base.

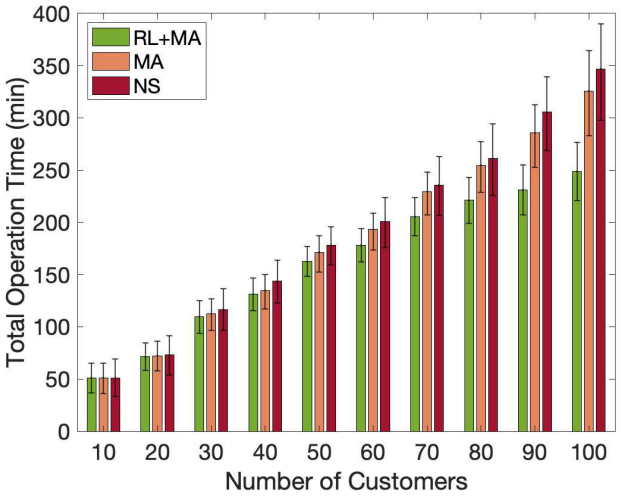

Fig. 5: The solution quality with varying numbers of customers. RL+MA significantly improves the solution quality especially for large-scale problems.

Fig. 4 presents example solutions created using RL+MA, MA, and NS algorithms. Interestingly, even with a modest customer base of 20, each method yields distinct solutions with varying levels of efficiency in terms of total operational time. Notably, the NS algorithm produces a solution with only 3 drone sorties, in contrast to the 4 drone sorties observed in the RL+MA and MA solutions. Since drone sorties are more efficient in general, this contributes to a reduction in overall operational time for RL+MA and MA. Additionally, it is observed that although RL+MA and MA both deploy the same two trucks, the routes taken by these trucks differ

significantly (*i.e.,* RL+MA results in smaller total operation time compared with that for MA). Such variations likely stem from the RL's ability to iteratively refine routing decisions through its adaptive learning mechanism, optimizing routes based on trial and error.

C. Computation Time

Another crucial part of our analysis focuses on assessing how incorporating the RL framework influences computational efficiency, especially when compared to MA and NS. The results are depicted in Fig. 6, demonstrating that the RL-enhanced approach, RL+MA, consistently achieves reductions in computation time across various problem sizes, even in scenarios with a relatively small number of customers. A key observation from our findings is that the RL framework not only maintains its efficiency advantage across all tested scales but also that this advantage becomes more significant as the number of customers increases. For instance, in cases involving 100 customers, the integration of RL leads to a decrease in computation time by approximately 13.2% compared to MA, and an even more substantial 27.3% compared to NS. This performance difference highlights the advantage of leveraging machine learning techniques to explore the solution space more effectively, as opposed to traditional methods that rely more on heuristic or stochastic solution adjustments.

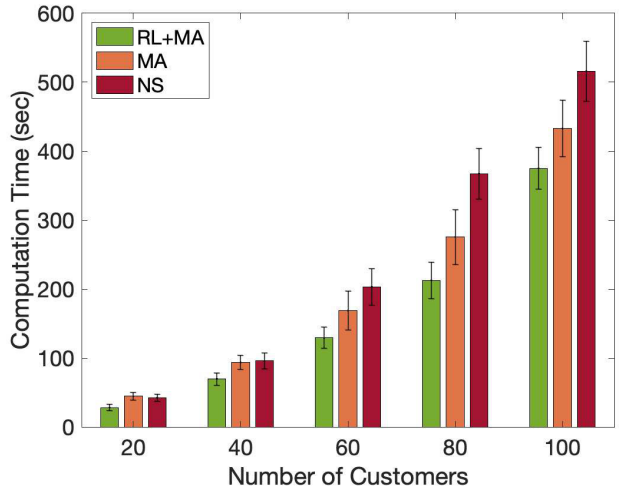

Fig. 6: The computation time with varying numbers of customers. RL+MA significantly improves the computation time across all test instances.

A notable advantage of the RL+MA approach is its capability to tackle considerably larger problems, which were previously beyond the reach of existing heuristic methods, not only because of the substantial decrease in computation time, but also the more effective exploration of the solution space. Specifically, with RL+MA, we were able to solve problems with up to 200 customers in just an average of 1,121 seconds. The result is particularly remarkable, considering that the existing benchmarks in the literature limit evaluations to a maximum of 100 customers for MA and 50 customers for NS.

D. Ablation Study

A key feature of SmartPathfinder is its ability to skip actions and randomly shuffle the solution if it doesn't show any improvement after a set number of k attempts. This technique is specifically engineered to avoid falling in local optima, thereby optimizing the solution discovery process. To systematically evaluate the influence of parameter k on the algorithm's efficiency and effectiveness, we undertook an ablation study. The objective of this study was to discern the impact of varying k values on the solution quality in terms of the operational time and the computational delay.

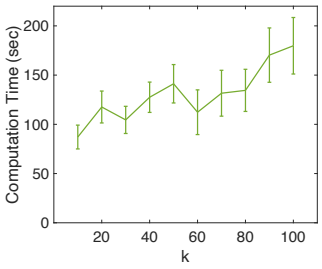

Fig. 7: The computation time measured with varying threshold k for initiating solution shuffling.

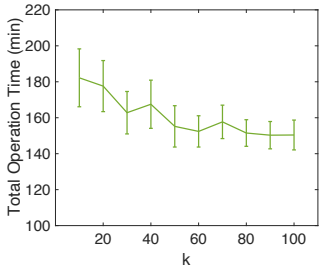

Fig. 8: The solution quality measured with varying threshold k for initiating solution shuffling.

First, we adjusted the value of k and tracked how it influenced the computation time. The results are detailed in Fig. 7, providing guidance for selecting an optimal k value. The results show that as the value of k increases, a notable increase in computation time is observed. This phenomenon can be attributed to the fact that a higher k value permits a more extensive examination of the solution space, as it delays the algorithm's exit from local optima in pursuit of potentially superior solutions. Consequently, this expanded exploration results in an increased computation time. Upon initial examination, the results appear to indicate that choosing a smaller k value could be beneficial for enhancing computational efficiency.

However, as depicted in Fig. 8, an improvement in performance, specifically regarding total operational time, is noted with an increase in k. This phenomenon highlights a discernible trade-off between computation time and overall performance when determining the optimal k value. Notably, the curve representing total operational time exhibits a plateau beyond a certain k value, which, in the context of this simulation, occurs approximately at $k = 60$. This observation implies that choosing a k value at the onset of this plateau phase—where additional increases in k no longer significantly enhance performance—presents a strategic approach for optimizing operational efficiency. In this experiment, the threshold value k was set to 60.

VI. Conclusion

We have presented SmartPathfinder, a novel approach to seamlessly integrate a RL framework with heuristic solutions for VRPD, targeting enhancements in both the solution quality and computational efficiency. This novel integration is facilitated by a thorough analysis and decomposition of heuristic solutions into universal components, followed by the design of the RL framework, and reassembly of the components seamlessly incorporating the RL framework. To evaluate the effectiveness of SmartPathfinder, we implemented a state-of-the-art heuristic solution for VRPD integrated with the RL framework and demonstrated significant improvements in both the effectiveness of the solutions and reduction in computational time. We believe that this pioneering effort marks a significant stride toward refining and accelerating the decision-making process in drone-assisted delivery systems.

As our future work, we aim to explore the theoretical limits of solution quality improvement achievable through integrating heuristic solutions with our RL framework. Additionally, we plan to investigate the adaptability of our proposed RL integration approach across various optimization problems, assessing its effectiveness with heuristic solutions tailored for different challenges. Lastly, we will conduct a broader assessment of SmartPathfinder by testing it with a diverse array of heuristic algorithms.

References

[1] N. Agatz, P. Bouman, and M. Schmidt, "Optimization approaches for the traveling salesman problem with drone," *Transportation Science*, vol. 52, no. 4, pp. 965–981, 2018.

[2] S. Lee, D. Hong, J. Kim, D. Baek, and N. Chang, "Congestion-aware multi-drone delivery routing framework," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 9, pp. 9384–9396, 2022.

[3] A. Goodchild and J. Toy, "Delivery by drone: An evaluation of unmanned aerial vehicle technology in reducing co2 emissions in the delivery service industry," *Transportation Research Part D: Transport and Environment*, vol. 61, pp. 58–67, 2018.

[4] W.-C. Chiang, Y. Li, J. Shang, and T. L. Urban, "Impact of drone delivery on sustainability and cost: Realizing the uav potential through vehicle routing optimization," *Applied energy*, vol. 242, pp. 1164–1175, 2019.

[5] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle routing problems for drone delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 70–85, 2016.

[6] L. D. P. Pugliese, F. Guerriero, and G. Macrina, "Using drones for parcels delivery process," *Procedia Manufacturing*, vol. 42, pp. 488–497, 2020.

[7] S. M. Shavarani, M. G. Nejad, F. Rismanchian, and G. Izbirak, "Application of hierarchical facility location problem for optimization of a drone delivery system: a case study of amazon prime air in the city of san francisco," *The International Journal of Advanced Manufacturing Technology*, vol. 95, pp. 3141–3153, 2018.

[8] B. R. Han, M. Li, Y. Zhang, and P. Li, "Value of autonomous last-mile delivery: Evidence from alibaba," *Available at SSRN*, 2024.

[9] Premack, Rachel, "America's largest retirement community can soon receive their prescriptions from CVS via a UPS Drone Delivery Service," https://www.businessinsider.com/ups-cvs-drone-deliveries-the-villages-florida-2020-4, online; accessed 7 March 2024.

[10] DHL, "DHL launches its first regular fully-automated and intelligent urban drone delivery service," online; accessed 7 March 2024. [Online]. Available: https://group.dhl.com/en/media-relations/press-releases/2019/dhl-launches-its-first-regular-fully-automated-and-intelligent-urban-drone-delivery-service.html

[11] N. Imran and M. Won, "Vrpd-dt: Vehicle routing problem with drones under dynamically changing traffic conditions," *arXiv preprint arXiv:2404.09065*, 2024.

[12] N. M. Imran, S. Mishra, and M. Won, "A-vrpd: Automating drone-based last-mile delivery using self-driving cars," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 9, pp. 9599–9612, 2023.

[13] Z. Wang and J.-B. Sheu, "Vehicle routing problem with drones," *Transportation research part B: methodological*, vol. 122, pp. 350–364, 2019.

[14] F. Tamke and U. Buscher, "The vehicle routing problem with drones and drone speed selection," *Computers & Operations Research*, vol. 152, p. 106112, 2023.

[15] H. Zhou, H. Qin, C. Cheng, and L.-M. Rousseau, "An exact algorithm for the two-echelon vehicle routing problem with drones," *Transportation Research Part B: Methodological*, vol. 168, pp. 124–150, 2023.

[16] Y. Xia, W. Zeng, C. Zhang, and H. Yang, "A branch-and-price-and-cut algorithm for the vehicle routing problem with load-dependent drones," *Transportation Research Part B: Methodological*, vol. 171, pp. 80–110, 2023.

[17] R. Kuo, E. Edbert, F. E. Zulvia, and S.-H. Lu, "Applying nsga-ii to vehicle routing problem with drones considering makespan and carbon emission," *Expert Systems with Applications*, vol. 221, p. 119777, 2023.

[18] D. Schermer, M. Moeini, and O. Wendt, "A matheuristic for the vehicle routing problem with drones and its variants," *Transportation Research Part C: Emerging Technologies*, vol. 106, pp. 166–204, 2019.

[19] Y. Yin, Y. Yang, Y. Yu, D. Wang, and T. Cheng, "Robust vehicle routing with drones under uncertain demands and truck travel times in humanitarian logistics," *Transportation Research Part B: Methodological*, vol. 174, p. 102781, 2023.

[20] R. Kuo, S.-H. Lu, P.-Y. Lai, and S. T. W. Mara, "Vehicle routing problem with drones considering time windows," *Expert Systems with Applications*, vol. 191, p. 116264, 2022.

[21] D. Sacramento, D. Pisinger, and S. Ropke, "An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones," *Transportation Research Part C: Emerging Technologies*, vol. 102, pp. 289–315, 2019.

[22] M. Momeni, S. Mirzapour Al-e Hashem, and A. Heidari, "A new truck-drone routing problem for parcel delivery by considering energy consumption and altitude," *Annals of Operations Research*, pp. 1–47, 2023.

[23] S. T. W. Mara, R. Sarker, D. Essam, and S. Elsayed, "Solving electric vehicle–drone routing problem using memetic algorithm," *Swarm and Evolutionary Computation*, vol. 79, p. 101295, 2023.

[24] S.-H. Huang, Y.-H. Huang, C. A. Blazquez, and C.-Y. Chen, "Solving the vehicle routing problem with drone for delivery services using an ant colony optimization algorithm," *Advanced Engineering Informatics*, vol. 51, p. 101536, 2022.

[25] Y.-q. Han, J.-q. Li, Z. Liu, C. Liu, and J. Tian, "Metaheuristic algorithm for solving the multi-objective vehicle routing problem with time window and drones," *International Journal of Advanced Robotic Systems*, vol. 17, no. 2, p. 1729881420920031, 2020.

[26] D. Lei, Z. Cui, and M. Li, "A dynamical artificial bee colony for vehicle routing problem with drones," *Engineering Applications of Artificial Intelligence*, vol. 107, p. 104510, 2022.

[27] Y. Liu, Z. Liu, J. Shi, G. Wu, and W. Pedrycz, "Two-echelon routing problem for parcel delivery by cooperated truck and drone," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 12, pp. 7450–7465, 2020.

[28] S. Meng, X. Guo, D. Li, and G. Liu, "The multi-visit drone routing problem for pickup and delivery services," *Transportation Research Part E: Logistics and Transportation Review*, vol. 169, p. 102990, 2023.

[29] T. Yu and H. Zhu, "Hyper-parameter optimization: A review of algorithms and applications," *arXiv preprint arXiv:2003.05689*, 2020.

[30] F. Tamke and U. Buscher, "A branch-and-cut algorithm for the vehicle routing problem with drones," *Transportation Research Part B: Methodological*, vol. 144, pp. 174–203, 2021.