S. Kosolsombat and C. Ratanavilisagul, "Applied Deep Reinforcement Learning for Solving the Vehicle Routing Problem with Time Windows," 2023 8th International Conference on Computational Intelligence and Applications (ICCIA), Haikou, China, 2023, pp. 21-25, doi: 10.1109/ICCIA59741.2023.00012.

Applied Deep Reinforcement Learning for Solving the Vehicle Routing Problem with Time Windows

Somkiat Kosolsombat[1st]
Department of Computre and Information Science, Faculty of Applied Science
King Mongkut's University of Technology North Bangkok (KMUTNB), Bangkok, Thailand
somkiatk@tu.ac.th, kosolsombat@hotmail.com

Chiabwoot Ratanavilisagu[2nd]
Department of Computre and Information Science, Faculty of Applied Science
King Mongkut's University of Technology North Bangkok (KMUTNB), Bangkok, Thailand
chaibwoot@gmail.com, chaibwoot.r@sci.kmutnb.ac.th

***Abstract*—** The Vehicle Routing Problem with Time Windows (VRPTW) is a challenging combinatorial optimization problem with many real-world applications. In this paper, we propose a Deep Reinforcement Learning (DRL) approach for solving the VRPTW using the Deep Q-Network (DQN) algorithm. Specifically, we use a variant of the DQN algorithm that incorporates the target network and experience replay techniques to improve the learning process. We also introduce a hybrid approach that combines a Greedy algorithm with the DQN algorithm (GDQN) to further enhance the exploration-exploitation trade-off in the search process. We evaluate the performance of the proposed approach on the Solomon benchmarks, which are widely used in the literature to benchmark the performance of various algorithms for the VRPTW. Our experimental results show that the proposed GDQN outperforms the DQN approach in terms of the quality of solutions generated, for type 25, achieving an average total used vehicle of 41.37% compared to 58.63% and average total distances of 52.34% compared to 47.66% for the DQN approach. For type 50, achieving an average total used vehicle of 42.22% compared to 57.78% and average total distances of 47.56% compared to 52.44% for the DQN approach. The proposed GDQN method provides an effective solution for the VRPTW problem in total used vehicle and total distances and use the same average total waiting time.

Keywords—Deep Q-Network, Greedy hybrid DQN, VRPTW

I. Introduction

The Vehicle Routing Problem (VRP) is a well-known optimization problem that has been studied extensively in the field of transportation and logistics management. The VRP involves finding the optimal routes for a fleet of vehicles to service a set of customers with the objective of minimizing the total cost. The problem is further complicated by the inclusion of time windows, which specify the time frame in which each customer can be serviced. The VRP with Time Windows (VRPTW) is a more realistic and challenging variant of the VRP, and has important practical applications in areas such as package delivery, public transportation, time limitation and emergency services.

Deep Reinforcement Learning (DRL) is a powerful technique for solving complex optimization problems that involves learning through trial and error. DRL has shown promising results in solving various optimization problems, including the Traveling Salesman Problem (TSP) and the Capacitated Vehicle Routing Problem (CVRP). A success case study of applying RL to the VRPTW was reported in [1], where the RL algorithm was able to significantly improve the efficiency of delivery routes for a major logistics company. Despite the success of the case study, there are challenges to applying the RL algorithm to real-world VRPTW problems, such as ensuring the stability and convergence of the algorithm, and dealing with the large state space of the problem.

The iMOPSE dataset is a benchmark dataset for multi-objective project scheduling and resource allocation problems. The name iMOPSE stands for "Improved Multi-Objective Project Scheduling and Resource Allocation Problem Instances." It includes a set of 110 problem instances with varying levels of complexity and resource requirements. The multi-skills resource-constrained project scheduling problem (MS-RCPSP) is a complex optimization problem that involves scheduling tasks with limited resources and different skill requirements. The objective of the problem is to minimize the project's completion time or cost while meeting all the project's constraints and are used to compare the ACO, HACO [2], and the proposed algorithm. The results show that the solution quality of the proposed algorithm is better than others comparing to ACO algorithms in the MS-RCPSP iMOPSE datasets.

In this paper, we propose a novel hybrid Greedy with Deep Q-Network approach comparing Traditional Deep Q-Network for solving the VRPTW and evaluate its effectiveness on benchmark datasets.

The rest of this paper is organized as following by literature review explains the related works. Next explains the proposed method. The experiment results are present in a later section.

And the last, concludes the research with a brief summary and directions for development algorithm in future work.

II. RELATED WORK

The VRPTW has been studied extensively in the literature, and various exact and heuristic methods have been proposed for solving the problem [2]. Exact methods, such as branch-and-bound and dynamic programming, can guarantee optimal solutions but are computationally expensive and often infeasible for large instances. Heuristic methods, such as the Clarke and Wright Savings Algorithm [2] [12] [13] [14] [15] and the Sweep Algorithm [16] [17] [18], are faster but may not provide optimal solutions.

Deep Reinforcement Learning (DRL) has recently emerged as a promising approach for solving combinatorial optimization problems [3]. DRL combines deep neural networks with reinforcement learning to learn a policy that maps problem instances to solutions. DRL has been successfully applied to various optimization problems, including the TSP and the CVRP [4][5]. DRL-based approaches have shown to achieve near-optimal solutions with significantly reduced computation time compared to exact methods.

One popular benchmark for evaluating the performance of these algorithms is the Solomon Benchmark [6]. The Solomon Benchmark consists of six sets of problems, each designed to highlight different factors that affect the behavior of routing and scheduling algorithms. The sets are differentiated by geographical data (random, clustered or mixed), the number of customers serviced by a vehicle, the percent of time-constrained customers, and the tightness and positioning of the time windows [6].

In the VRPTW domain, several optimization algorithms have been proposed including Ant Colony Algorithms (ACA) [7], Q-Learning [8] and Deep Q-Networks (DQN) [9]. In comparing these algorithms, it was found that DQN outperforms Q-Learning and ACA in terms of solution quality and convergence speed [10].

III. THE PROPOSED WORK

We propose a DRL-based approach for solving the VRPTW using the Deep Q-Network (DQN) algorithm. Specifically, we use a variant of the DQN algorithm that incorporates the target network and experience replay techniques to improve the stability and convergence of the learning process. We also introduce a hybrid approach that combines a Greedy algorithm with the DQN (GDQN) [11] to further enhance the exploration-exploitation trade-off in the search process finding the best routing, number of vehicles, total distances and total waiting time.

To evaluate the performance of the proposed approach, we use the Solomon benchmarks, which are widely used in the literature to benchmark the performance of various algorithms for the VRPTW. The Solomon Benchmark consists of sets of problems, each designed to highlight different factors that affect the behavior of routing and scheduling algorithms. The sets are differentiated by geographical data (random, clustered or mixed), the number of customers serviced by a vehicle, the percent of time-constrained customers, and the tightness and positioning of the time windows.

The main concept to develop for the early stages of DQN, we use random methods to select clients in the early learning phase to allow the model to explore the environment, and then the model is gradually de-randomized to use the Q values from the model to choose the customers to go. And for Greedy hybrid DQN (GDQN), in the beginning, we use the greedy method to select the customer and gradually decrease, then the chance that we will use the model will increase. This model will not do precalculated to select the first customer with the number of vehicles. But we use this method to use a car and let it choose according for algorithm.

The pseudo code of Deep Q-Network solution as below:

<u>Main Pseudo code</u>

```
Initialize network weight
pre-calculate the route for each vehicle
for episode <- 1,2, ... n do
   while having a remaining customer do
      if time_frame is 1 do
         A <- get action from pre-calculate
      else do
         if random < epsilon do
            A <- random action
         else do
            A <- predict action score with the current state for each active vehicle
         end if
         for each remaining customer do
            if not have an active vehicle can serve a customer and have an inactive vehicle do
               A <- assign a new vehicle to serve a customer
            end if
         end for
      end
      S, R <- do action to each vehicle
      memorize S, A, R
   end while
   train model with experience replay
   decrease an epsilon
end for
```

<u>Pre-calculate route</u>

```
cv <- 0
while cv < number of maximum vehicle do
   sort remaining feasible customers with ready_time and distance in ascending
   cust <- get feasible customer from picking the first customer in a sorted list
   set first customer to a current vehicle with cust
   cvP <- position of cust
   delete cust from a customer list
   while cvW < maximum capacity or having a feasible customer to current vehicle
      sort remaining feasible customers with ready_time and distance from cvP in ascending
      cust <- first customer from a sorted list
      cvP <- position of cust
      cvW <- cvW + deamand of cust
   end while
   cv <- cv+1
end while
```

The pseudo code of Greedy hybrid Deep Q-Network solution as below:

```
Greedy DQN Pseud code
Initialize network weight
for episode <- 1,2, ... n do
    while having a remaining customer do
        if time_frame is 1 do
            set first vehicle to idle state
        if have an idle state vehicle:
            if random < epsilon do
                A <- set action with greedy approach by sorting a ready_time and distance
            else do
                A <- predict action score with the current state for each active vehicle
            end if
        for each remaining customer do
            if not have an active vehicle can serve a customer and have an inactive vehicle do
                A <- assign a new vehicle to serve a customer
            end if
        end for
        S, R <- do action to each vehicle
        memorize S, A, R
    end while
    train model with experience replay
    decrease an epsilon
end for
```

IV. Experiment and Results

We compare the performance of the proposed DQN approach with the Greedy hybrid DQN method. The GDQN method involves a greedy algorithm that selects the next action based on the highest Q-value. We compare the two methodologies in terms of the performance of service solutions offering.

We measure the quality of solutions using the total travel distance of the routes, total waiting time, and total vehicle used generated by the algorithms. We report the results in terms of the number from the optimal solution, as the optimal solutions are not available for all instances in the Solomon benchmarks. Comparing algorithm performance uses the total used vehicle, total distances from runs because DQN and GDQN are reinforcement learning method. The measures of algorithm performance in the experiments are as follows: the average total used vehicle is the average of best total used vehicle in the final iteration from all runs. The average total distances is the average of summary distances in the final iteration from all runs. The total used vehicle indicates the solution services of an algorithm. The best is the less value that means the method use effectively. The research was tested on two types six classes Solomon benchmark datasets. The first type has twenty-five customers three classes. The second type has fifty customers three classes. The Solomon benchmark is a standard dataset and can be used to compare the performance of algorithm.

In our experiments, we set the hyperparameters of the DQN algorithm and the Greedy algorithm based on the recommendations in the literature [2] [19]. We run each algorithm that contain twenty-five and fifty customers, each of which consists of three classes of customers: C, R, and RC by following table below.

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

TABLE I. COMPARATIVE RESULTS OF DQN AND GDQN

#Customer	DQN (V)	GDQN (V)	DQN (D)	GDQN (D)	DQN (WT)	GDQN (WT)
25 – C1	11	5	701	580.33	17.22	17.22
25 – R1	15	14	1045.42	969.17	16.25	13.67
25 – RC1	15	14	1278.88	1239.75	15.38	10.75
50 – C1	17	9	1648.11	1479.90	34.89	32.78
50 – R1	24	23	2020.08	1925.08	27.92	21.75
50 – RC1	25	23	2746.13	2520.63	25.13	18.25
25 – C2	9	5	800.75	645.75	18.13	17.13
25 – R2	11	7	992.91	931.64	16.36	14
25 – RC2	10	7	1244.25	1154.13	17.88	16.13
50 – C2	13	5	1742.63	1572.38	37.63	37.25
50 – R2	19	13	2143.09	1918.73	29.10	25.91
50 – RC2	17	14	3029.88	2654.38	33	27.75

Table I Show the output for Solomon datasets: #Customer is the number of customers and dataset class and type; DQN: Deep Q-Network; GDQN: Greedy hybrid Deep Q-Network; V: Total vehicles, D: Total distances; WT: Total waiting time (in seconds)

The experiment results show that type twenty-five customers, the GDQN method used fewer total vehicles and total distances than the DQN method, and the waiting time is about the same.

In summary, we propose a DRL-based approach for solving the VRPTW using the DQN algorithm and a hybrid approach that combines the DQN with a Greedy algorithm. We evaluate the performance of the proposed approaches on the Solomon benchmarks and compare them in terms of the quality of solutions generated and the computation time required to generate them.

V. .Conclusion

In this study, we proposed a DRL-based approach for solving the VRPTW using the DQN algorithm and a hybrid approach that combines the DQN with a Greedy algorithm (GDQN). We evaluated the performance of the proposed approaches on the Solomon benchmarks and compared them in terms of the quality of solutions generated and the computation time required to generate them. The results show that the Greedy hybrid Deep Q-Network (GDQN) approach outperforms the DQN approach in terms of the total used vehicle, total distances and the same total waiting time. Overall, the proposed Deep Reinforcement Learning approach provides an effective solution for the VRPTW problem, achieving high-quality solutions with a low percentage deviation from the optimal solution. Furthermore, the hybrid approach that combines the DQN with a Greedy algorithm provides a good balance between exploration and exploitation, resulting in improved performance compared to using a single algorithm. Future work could explore the use of other DRL algorithms, such as actor-critic or policy gradient methods, to further improve the performance of the VRPTW problem.

REFERENCES

[1] T. Phiboonbanakit, T. Horanont, V.-N. Huynh, and T. Supnithi, "A Hybrid Reinforcement Learning-Based Model for the Vehicle Routing Problem in Transportation Logistics," *IEEE Access*, vol. 9, pp. 163325–163347, 2021.

[2] M. Gendreau, G. Laporte, "Vehicle routing problem with time windows," in Handbooks in Operations Research and Management Science, vol. 14, Elsevier, 2007, pp. 367-428.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski et al., "Human-level control through deep reinforcement learning," Nature, vol. 518, no. 7540, pp. 529-533, 2015.

[4] M. Nazari, F. Soleymani, "A Deep Reinforcement Learning Approach for the Capacitated Vehicle Routing Problem," in IEEE Access, vol. 9, pp. 7947-7959, 2021.

[5] T. B. Pedersen, M. G. Vidal, "A survey of matheuristics for the vehicle routing problem with time windows," EURO Journal on Transportation and Logistics, vol. 3, no. 1-2, pp. 1-23, 2014.

[6] R. Salomon, "Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms," Biosystems., vol. 39, no. 3, pp. 263–278, 1996.

[7] Y. Li, H. Soleimani, and M. Zohal, "An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives," J. Clean. Prod., vol. 227, pp. 1161–1172, Aug. 2019.

[8] J. Clifton and E. Laber, "Q-Learning: Theory and Applications," Annu. Rev. Stat. Appl., vol. 7, no. 1, pp. 279–301, Mar. 2020.

[9] V. Mnih et al., "Playing Atari with Deep Reinforcement Learning," arXiv [cs.LG], Dec. 19, 2013 [Online]. Available: http://arxiv.org/abs/1312.5602

[10] L. Pan, J. Liang, and B. Qu, Bio-inspired Computing: Theories and Applications: 14th International Conference, BIC-TA 2019, Zhengzhou, China, November 22–25, 2019, Revised Selected Papers, Part I. Springer Nature, 2020.

[11] H. G. Ramos, D. D. Silva, "A reinforcement learning approach for the vehicle routing problem with time windows," in Proceedings of the 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, United Kingdom, 2020, pp. 1-8.

[12] J. F. Clarke and G. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," Operations Research, vol. 12, no. 4, pp. 568-581, 1964.

[13] D. D. W. Clarke and J. W. Wright, "The application of the savings algorithm to the vehicle routing problem," Operations Research, vol. 16, no. 6, pp. 1052-1061, 1968.

[14] R. Baldacci, M. G. C. Resende, and G. Rinaldi, "The capacitated vehicle routing problem with route-dependent costs," Networks, vol. 54, no. 4, pp. 223-237, 2009. (This paper discusses an extension of the Clarke and Wright savings algorithm.)

[15] S. Salhi and A. Sari, "A review of metaheuristics for vehicle routing problems," Computers & Operations Research, vol. 35, no. 10, pp. 3291-3303, 2008. (This paper provides an overview of the Clarke and Wright savings algorithm and other metaheuristic approaches to the vehicle routing problem.)

[16] A. N. Christofides, "An algorithm for the vehicle-dispatch problem," Operational Research Quarterly, vol. 17, no. 4, pp. 411-418, 1966. (This paper introduces the Sweep algorithm for the vehicle routing problem.)

[17] R. Baldacci, M. G. C. Resende, and G. Rinaldi, "The capacitated vehicle routing problem with route-dependent costs," Networks, vol. 54, no. 4, pp. 223-237, 2009. (This paper compares the Sweep algorithm with other heuristic algorithms for the capacitated vehicle routing problem.)

[18] M. P. Bektaş and G. Laporte, "The pollution-routing problem," Transportation Research Part B: Methodological, vol. 45, no. 8, pp. 1230-1250, 2011. (This paper uses the Sweep algorithm to generate initial solutions for the pollution-routing problem.)

[19] J. Li, L. Xiao, Y. Li, L. Li, "Deep Reinforcement Learning for Vehicle Routing Problem with Time Windows," in IEEE Access, vol. 9, pp. 6625-6633, 2021.

AUTHORS' BACKGROUND

Your Name	Title	Research Field	Personal website
Somkiat Kosolsombat	Ph.D. Candidate	Artificial Intelligence, Natural language processing, Machine Learning, Optimization, Big Data	
Chiabwoot Ratanavilisagul	Aassistant Professor Dr.	Artificial Intelligence, Natural language processing, Machine learning, Searching Technique, Optimization	http://chaibwoot.no-ip.org/Chiabwoot.jsp