IJCI V11-2(2024) 79–94
International Journal of Computers and Information (IJCI)
Available online at *https://ijci.journals.ekb.eg/*

Reinforcement Learning-Driven Enhancement of Medical Waste Collection within Capacity-Homogeneous Vehicle Routing

Norhan khallaf [a*], Osama Abdel Raouf [a], Mohy Hadhoud [b], Ahmed Kafafy [c]

[a] Machine Intelligence Dept, Faculty of Artificial Intelligence, Menoufia University, Egypt.
[b] Information Technology Dept, Faculty of Computer and information, Menoufia University, Egypt.
[c] Data Sciences Dept, Faculty of Artificial Intelligence, Menoufia University, Egypt.
nourhan.khalaf@ci.menofia.edu.eg, osamaabd@ci.menofia.edu.eg, mmhadhoud@ci.menofia.edu.eg , ahmedkafafy80@gmail.com ,

Abstract

Artificial intelligence is increasingly being used in various fields, including the management of hazardous medical waste. Medical waste poses an economic burden and a risk to public health, and it should be disposed of with care, preferably in areas far from residential areas. Data was collected on waste generated by 15 government hospitals in Menoufia Governorate and a single disposal site in Kafr Dawood, along with a central collection point for waste transport vehicles. This study addresses the issue of limited-capacity vehicle routing, which is considered a complex problem (NP-hard). Specific vehicles are designated to collect waste from hospitals and transport it to the disposal center, with the goal of finding the shortest route while maximizing the vehicle's capacity, which is limited to three tons. Reinforcement learning techniques were developed, treating the vehicle as an agent trained to choose the shortest, least costly route between hospitals. The SARSA algorithm was implemented and improved. Solutions include SARSA, Dijkstra, knapsack dynamic programming, and hybrid approaches that combine SARSA with Dijkstra and SARSA with knapsack dynamic programming. The result shows that the hybrid approach between SARSA and knapsack dynamic programming is the most effective, as it reduces the number of vehicles used for waste transport and maximizes the vehicle's capacity, determining the shortest routes between all hospitals. Finally, transportation costs were calculated to complete the mathematical model for medical waste management.

Keywords: *Reinforcement learning, Closed Capacity Vehicle Routing Problem, Dijkstra, knapsack problem*

1. Introduction

Medical waste is a crucial concern that must be taken into account as it can impact the spread of diseases among populations and holds economic significance in terms of its disposal [1]. These waste materials originate from hospitals, healthcare facilities, private clinics, outpatient centers, and more. Medical waste can be categorized into two types: hazardous and non-hazardous waste. It's worth noting that non-hazardous medical waste can sometimes be recycled, while some can be incinerated within hospital incinerators if available [2]. Hospitals without incinerators for non-hazardous waste can seek assistance from other hospitals with incineration facilities. On the other hand, hazardous medical waste requires careful handling. It must be transported to distant incineration facilities located away from residential areas. Hospitals generating hazardous medical waste collect and package the waste in specialized medical waste bags. These waste materials are then transported in dedicated vehicles.

The Ministry of Health specifies qualified vehicles for collecting hazardous medical waste to prevent pollution during transportation [3]. Additionally, trained and qualified drivers are assigned by the Ministry of Health to handle the transportation of the waste materials, ensuring safety protocols are followed in case of injuries or accidents [4]. Each of the vehicles follows a specific route to achieve the shortest path for waste collection and maximize the vehicle's load capacity, preventing vehicles from traveling empty. Indeed, this problem can be classified as a Vehicle Routing Problem (VRP) with the added challenge of capacity constraints [5], given that the vehicles transporting waste between hospitals have limited capacities. Consequently, it falls within the category of a Capacity Vehicle Routing Problem (CVRP). There are two main types of Capacity Vehicle Routing Problems: closed and open. Closed CVRP: In this type, each route must start and end at the same depot (collection center). It means that the vehicles return to the depot after serving all the hospitals. In an open CVRP, the routes do not necessarily have to start and end at the same depot. Vehicles can finish their routes at any hospital location. This type is more flexible and is suitable when vehicles do not need to return to a central depot after completing their routes [6]. The CVRP was classified as closed-CVRP. The problem of Capacity Vehicle Routing, especially when considering factors like optimizing routes, minimizing costs, and adhering to capacity constraints, can be effectively modeled and solved using various machine learning algorithms and techniques. Machine Learning: It is a branch of Artificial Intelligence that focuses on developing techniques and models that allow systems and software to learn from data and improve their performance over time. Machine Learning uses algorithms and mathematical models to discover patterns and guide towards a better outcome based on data. Machine learning can be categorized into several types based on the learning style and the nature of the learning process [7].

The main types of machine learning are: Supervised Learning, unsupervised Learning, Semi-supervised learning and Reinforcement Learning. Reinforcement learning (RL) differs from other types of machine learning in its primary goal, type of data, and methodology. The main objective of reinforcement learning is to help an agent make optimal decisions in a specific environment through trial and error, aiming to maximize long-term reward or benefit. This learning paradigm relies on state and reward feedback, where the agent learns from its experiences and adjusts its strategy based on past interactions. In contrast, supervised learning aims to predict output based on given input data, utilizing predefined input-output pairs, and the model learns by comparing its predictions with the actual outcomes. Unsupervised learning, on the other hand, seeks to discover hidden patterns in data without direct input-output signals. It involves continuous interaction with the environment, where the agent takes actions, receives feedback, and adjusts its strategy accordingly. RL has several methods and algorithms designed for various applications and scenarios for example Model-Free RL and Model-Based RL [8]. Model-Free RL is an approach where an agent learns to make decisions without explicitly building a model of the environment. It learns through trial and error by interacting with the environment, collecting data, and updating its policy based on the observed rewards and state transitions. Model-Based RL is an approach where an agent builds an explicit model of the environment, including the transition dynamics, rewards, and uncertainties. It uses this model for simulation and planning to make decisions.

Common RL algorithms include SARSA (State–action–reward–state–action), Q-Learning, Deep Q-Networks (DQN), and Policy Gradient methods [9]. For this problem, SARSA has been chosen, which is classified as a Model-Free RL algorithm. SARSA, an abbreviation for State-Action-Reward-State-Action, represents a reinforcement learning approach employed to train agents in making decisions within uncertain environments. In SARSA, the agent refines its policy by adapting to observed sequences of state-action-reward-state-action interactions. As an on-policy learning algorithm, SARSA continuously enhances its current policy, considering both immediate rewards and future states. This technique proves especially valuable in contexts where actions directly influence the agent's rewards and subsequent states. SARSA is widely used in applications like robotics, game playing, and autonomous systems where agents need to learn optimal decision-making policies through trial and error [10]. The SARSA algorithm is utilized to solve CVRP. To enhance the results, an integrated approach that combines SARSA with Dijkstra's algorithm and the Knapsack problem is used. Dijkstra’s algorithm is a widely used graph traversal and shortest path-finding algorithm. The primary objective of Dijkstra's algorithm is to find the shortest path between a specified source node and all other nodes in a weighted graph [11]. The algorithm operates on graphs where each edge has a non-negative weight or cost associated with it. Dijkstra's algorithm is a greedy approach that iteratively selects the node with the smallest tentative distance from the source and updates distances to its neighboring nodes. This algorithm has numerous applications in network routing, GPS (Global Positioning System) navigation, transportation planning, and various optimization problems [12]. Combining Dijkstra's greedy algorithm with SARSA can introduce various challenges. The algorithm often relies on specific assumptions about data representation and graph structure.

The challenges associated with this integration extend beyond the algorithm's inherent greediness, encompassing additional complexities related to different data handling approaches. Another type of integration is introduced as the Knapsack solver. The Knapsack problem is a classic optimization problem in mathematics and computer science. The goal is to choose items to fit into a knapsack (or backpack) with limited capacity such that the total value of the selected items is maximized without exceeding the knapsack's weight limit [13]. The Knapsack problem has various variants, including the 0/1 Knapsack problem (where items can be either selected or not) and the Fractional Knapsack problem (where items can be partially selected). It has practical applications in resource allocation, financial portfolio optimization, and various decision-making scenarios where limited resources must be allocated optimally [14]. There are several approaches to solving the knapsack problem categorized into (Dynamic programming, greedy algorithms, Branch and Bound, Approximation Algorithms, Genetic Algorithms and Integer Linear Programming (ILP)) [15] [16]. The knapsack problem is solved using a dynamic programming approach. The challenge of this approach lies in its capacity to efficiently find an optimal solution by eliminating redundant computations [17]. Combining SARSA and a knapsack solver

is suitable for scenarios where the agent learns by interacting with an environment and receives feedback in the form of rewards. It is used for optimization problems where items have different values and weights, and the objective is to maximize the total value without surpassing a specified weight limit. Five distinct methods are employed to solve this model.
In the next sections, other aspects of the research are explored, with Section 2 focusing on discussing related work. Section 3 discuss the definition of the problem and RL model for CVRP. Section 4 discuss the solution approach like as SARSA, Dijkstra, the integration between SARSA and Dijkstra, the knapsack problem and then the integration between SARSA and knapsack solver. Section 5 discuss the presentation and Analysis of the results. Section 6 discuss the comparative evaluation of the Five Distinct Algorithms. Section 7 summarize the Findings of this Study.

2. Related work

When previous studies are examined, it is seen that there are many studies on the subject of medical waste disposal. papers that provide comprehensive discussions on the vehicle routing problem were discussed first, then publications dedicated to the management of medical wastes. Lastly, the literature pertaining to reinforcement learning techniques were discussed.

In the referenced study [18] it introduces a novel approach employing reinforcement learning techniques to discover optimal routes from a depot to a set of customers. This approach takes into consideration the capacity constraints of the vehicles, ultimately aiming to minimize the overall cost of transporting goods and services. To address the CVRP, different methodologies were used, such as the exact method of column generation, Google's Operations Research tool, and reinforcement learning. The ultimate objective is to attain optimality in solving large-scale vehicle routing problems. One of the disadvantages of this research is the computational complexity associated with certain algorithms utilized, notably column generation and RL. These algorithms, while powerful, can demand significant computational resources and expertise to be implemented effectively. This complexity may pose a challenge for organizations with limited computational resources or expertise. To enhance the practical relevance of the research and its applicability to real-world scenarios, future work should consider extending the model to include time window constraints, resulting in a more comprehensive CVR with time-windows (CVRPTW) solution.

In the referenced study [19] it presents a solution to the problem of container scheduling in cloud platforms using a predictive reinforcement learning algorithm called A-SARSA. It discusses the challenges faced by traditional reinforcement learning methods in container scheduling, such as untimely scheduling, lack of decision-making accuracy, and poor adaptability to changing workloads. The A-SARSA algorithm addresses these issues by combining the ARIMA model and neural network model to ensure predictability, accuracy, and adaptability in scaling strategies. Through extensive experiments, A-SARSA has been shown to significantly reduce SLA violation rates while maintaining resource utilization levels. This approach used in this text includes untimely resource scheduling, inaccurate scaling decisions, and repeated resource scheduling in RL applied to container scheduling scenarios. To address these issues, one potential solution is to enhance the prediction accuracy of workload by incorporating more sophisticated forecasting models or techniques. Additionally, improving the efficiency of action selection and decision-making processes through the refinement of neural network models or the integration of more advanced algorithms could lead to a better outcome. Furthermore, optimizing the RL algorithm parameters or exploring alternative RL approaches tailored specifically for container scheduling tasks may also help mitigate these shortcomings.

In the referenced study [20] it explores the advantages of autonomous vehicles in managing urban traffic, emphasizing the use of SARSA (λ)-based Adaptive Traffic Signal Controller (ATSC) systems. Recognizing shortcomings in traditional SARSA (λ) models, the study suggests enhancements like employing a Gaussian function for decay regulation and adopting MaxAbs scaled state values. Additionally, combining the A-star routing algorithm with the proposed model enhances performance. Evaluation in a SUMO-based simulation on a realistic 21-intersections network demonstrates notable reductions in vehicle wait times and stops, along with improved trip speeds with the A-star combined controller. One of the drawbacks of the traditional method was inefficient weight updating. One proposed solution is to use a Gaussian decaying approach for the eligibility trace vector, which improves weight updating efficiency.

In the referenced study [21] it addresses this limitation by introducing a multi-path routing algorithm based on an enhanced breadth-first search. The proposed approach begins by employing the improved breadth-first search algorithm to gather front hop node information for the destination node, taking into account the inter-satellite network topology. Subsequently, all shortest paths are derived by backtracking through the front hop nodes. Through simulation experiments, the proposed algorithm demonstrates improvements in the throughput of the inter-satellite network, coupled with reduced time delay and packet loss rates. The limitations highlighted in this paper include the constraint of a single shortest path in traditional routing algorithms. The adaptability to network dynamics is addressed by the improved inter-satellite multipath algorithm, combined with reinforcement learning, overcoming the shortcomings of traditional algorithms. This strategy proves effective in preventing data traffic congestion during inter-satellite network communication, leading to a lower latency, a reduced packet loss rates, and an improved overall network throughput performance.

After a comprehensive review of collaborative research and identifying existing limitations, it was observed that there is a gap in addressing the issue of hospital waste transportation in the governorates of Egypt using either mathematical models or artificial intelligence methods. Some studies utilizing artificial intelligence techniques were found to use the algorithm without introducing any enhancements or integrating it with other artificial intelligence methods. The study approach involves applying the waste transportation problem in Egypt using a reinforcement learning method. Moreover, one of the reinforcement learning algorithms is improved and integrated with other algorithms, conducting a comparative analysis of the results. This research is distinctive because it avoids the shortcomings identified in several collaborative studies.

3. Problem definition

Health-care waste is classified as a subcategory of hazardous waste in many countries. The study aims to solve the problem of hazardous waste from Health-care centers and transfer it to the disposal site (destination site) and then return to collection Centers (depot site), but these Health-care centers are sited in different locations on the google map in the Menoufia governorate as shown in Table 1.

Table 1. Length of latitude and longitude for hospital nodes and its index

Index	Hospital Name	x	y	Demands	Demands
S	Directorate of Health Affairs	30.55	31.13	0	0
1	Quwisna General	30.55	31.13	0.904	0.957
2	Qasr	30.57	31.01	1.472	1.503
3	Zawiya Al Naoura Fever	30.54	30.86	0.325	0.427
4	Mit Khalaf Fever	30.50	31.13	1.076	1.106
5	El-Bajur General	30.43	31.02	0.604	0.711
6	Sers El-Lyan	30.44	30.96	0.271	0.191
7	Menouf Central	30.47	30.92	0.658	0.511
8	Menouf Fever	30.47	30.92	0.761	0.779
9	Berket El-Sabaa	30.63	31.09	0.334	0.346
10	Shentena Al-Hagar Fever	30.64	31.05	0.236	0.314
11	Tala Central	30.68	30.95	0.491	0.427
12	Tala Fever	30.67	30.93	0.355	0.473
13	Al-Shuhada Central	30.59	30.90	0.623	0.458
14	Ashmoun General	30.29	30.98	0.582	0.657
15	National Liver Institute	30.57	31.01	3	2.989
D	Kafr Dawood Al-Sadat	30.46	30.82	0	0

In Table 1, the first column represents the number of each hospital. The second column represents the names of the hospitals used in this paper. The third and fourth columns represent the latitudinal and longitudinal locations of the hospitals on Google Maps. For instance, Node 1, which corresponds to Quwisna General Hospital, is located at coordinates (30.55m, 31.13m) on the x and y axes, respectively. The fifth and sixth columns represent the amount of waste produced by each hospital twice a week, meaning the vehicles visit the hospitals twice a week. During these visits, the waste is collected and stored twice a week, awaiting the vehicles' visits to each hospital. Collected a statistical sample of data on hospital waste, along with the locations of each hospital and the disposal site from various sources, including the Directorate of Health Affairs in Menoufia, the Public Mobilization and Statistics (CAPMAS), the Ministry of Health, and Google Maps [22][23]. The method for collecting data on hospital waste involved approaching the Directorate of Health Affairs in Menoufia, which serves as the central hub for all information and statistics regarding all government hospitals in the province. They have detailed information on waste quantities, number of beds, number of deaths, number of births, and so on for each hospital. The data is collected and analyzed by certain personnel in the Directorate to ensure its accuracy. A sample of this data was taken to create a mathematical model to solve the vehicle capacity problem, and this sample was from the first week of January 2024. In Menoufia government and its suburbs, there are a total of 15 hospitals, one collection center (depot), and one disposal site in the City of Sadat.

The proposed mathematical model is a multi-hazardous waste, multi-period waste, and multi-vehicle model that includes waste production nodes (hospitals), potential one depot (Directorate of Health Affairs) and one disposal sites (Kafr Dawood Al-Sadat). The objective of this model is to simultaneously minimize the total cost, total time, total distance, and pollution risk, while maximizing the total capacity of the homogenous fleet vehicles. This problem falls under the category of closed capacity vehicle routing problem (CVRP), where vehicles depart from the collection centers, travel to visit hospital node until reach to the disposal sites, and then return to the collection centers. A CVRP application is defined as a set of nodes n (hospitals) that index from $1 \cdots n$ with n equal to 15 in this specific case. Each node is associated with specific demand d_i;The distance from hospital i to hospital j is defined by Δ_{ij} .That information can be extracted from Table 1. Node S mean depot and the disposal site has index with node D in Table 1. Each vehicle has fixed capacity Q as shown in the following formula that equal to 3 tons. By adopting a single route scenario, the model aims to streamline the waste management process and optimize the efficiency of vehicle utilization. It effectively minimizes unnecessary detours, reduces travel distances, and optimizes resource allocation. Furthermore, it ensures that waste collection and transportation operations originate from the collection center, establishing routes between hospitals to efficiently dispose of the collected waste at the disposal site according to vehicle capacity. This approach leads to significant cost savings, diminished pollution risks, and an overall enhancement of the waste management system's effectiveness as shown in the following mathematical model [24].

- $N\ is\ set\ of\ hospitals$ with depot and disposal site node $\mathrm{N} = \{\mathrm{S}, 1,2,3, \ldots, 15, \mathrm{D}\}$.
- $A\ is\ set\ of\ arcs, with\ A = \{(i,j) \in N^2 : i \neq j\}$
- $c_{ij}\ is\ cost\ of\ travel\ over\ arc\ (i,j) \in A$
- $Q\ is\ the\ vehicle\ capacity$
- $d_i\ is\ the\ amount\ that\ has\ extracted\ from\ hospitals\ and\ deliver\ to\ disposal\ site\ i \in N$
- $t_{ij}\ is\ time\ of\ travel\ over\ arc\ (i,j) \in A, T\ mean\ summation\ of\ total\ time\ \left(t_{ij}\right).$
- $dis_{ij}\ is\ distance\ of\ travel\ over\ arc\ (i,j)\ A\ dis\ is\ a\ summation\ of\ total\ time\ \left(dis_{ij}\right)$

Then the formulation is the following:

$$\min \quad \sum_{i,j \in A} c_{ij} X_{ij} \tag{1}$$

$$s.t. \quad \sum_{j \in N, j \neq i} X_{ij} = 1 \tag{2}$$

$$\sum_{i \in N, i \neq j} X_{ij} = 1 \tag{3}$$

$$\sum_{i \in N} \sum_{j \in N\{0\}, i \neq j} d_i\, x_{ij}{}^n \leq Q \qquad \forall n \in N \tag{4}$$

$$\sum_{i \in N} \sum_{j \in N\{0\}, i \neq j} t_i\, x_{ij}{}^n = T \qquad t_i \in T \tag{5}$$

$$\sum_{i \in N} \sum_{j \in N\{0\}, i \neq j} dis_i\, x_{ij}{}^n = Dis \qquad dis_i \in Dis \tag{6}$$

To achieve this goal, CVPR was solved by SARSA algorithm. Then SARSA was integrated with two different techniques, as discussed in the next sections.

4. Proposed approach

Solutions to this problem are presented in this section using three artificial intelligence approaches.

4.1 The Proposed Algorithm

SARSA serves as a reinforcement learning algorithm designed for sequential decision-making in an environment. Functioning as a model-free and on-policy approach, it involves the agent learning a policy based on its current state, receiving rewards, and refining its understanding of the environment. In the SARSA process, the agent assesses its present state, chooses an action, observes the subsequent state, receives a reward, and adjusts its policy in response. SARSA is introduced from the perspective of our problem. A group of agents, denoted as $V = \{1,2, \ldots, V\}$ (referred to as vehicles), are tasked with collecting waste from hospitals, denoted as $h = \{1,2, \ldots, 15\}$, and transporting it to a disposal node $h = D$, then returning to the depot $h = S$. This algorithm involves taking an action (A) in the current state (S), receiving a reward (R), transitioning to the next state (S_1), and then taking action (A_1) in S_1.The various elements of our model are further explained in depth in Figure 1 below.

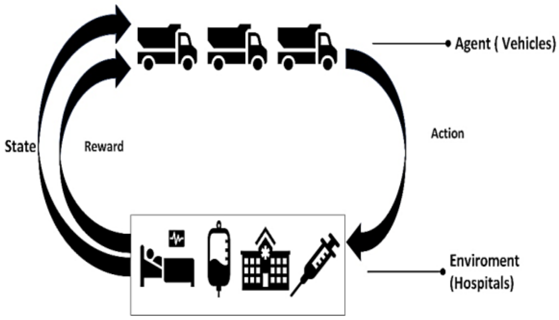

Figure 1. SARSA model for CVRP

In Figure 1, the SARSA framework adopted for the specific case study is illustrated. The model is designed to determine the most efficient route between hospitals, considering waste collection capacity. A 17x17 grid was established with distance values in each cell, initially set to zero in the Q-matrix. SARSA algorithm is utilized to iteratively refine these values through trial-and-error learning. The system updates values by evaluating differences between current and target values, calculated using the Bellman equation [25].

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma\, Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \tag{7}$$

where $\alpha \in (0,1)$ is the learning rate.

The SARSA procedure begins with the initialization of $Q(S, A)$ to arbitrary values. During this phase, the initial current state (S) is established, and the initial action (A) is chosen using an epsilon-greedy algorithm policy based on current Q-values. An epsilon-greedy policy strikes a balance between exploitation and exploration methods during the learning process, ensuring the selection of the action with the highest estimated reward. Exploitation entails leveraging existing, estimated values to maximize previously acquired rewards during the learning process. Exploration, on the other hand, entails seeking new insights into actions, which might lead to sub-optimal actions in the short term but could yield long-term benefits in identifying the optimal action and reward. Clarify the meaning of state, action, and reward in relation to our problem.

4.1.1 States

In the medical waste management algorithm, states were defined as $\mathcal{S} = \{v\ , h, d\ \}$:

- v represent Agent (vehicles).
- h represent hospitals node.
- d represent disposal site where wastes generated from hospitals is incinerated.

The vehicles' itinerary commences at the collection center under the administration of the Directorate of Health Affairs. From there, the shortest route between hospitals is determined to gather waste from each hospital and transport it to the disposal site for proper disposal as shown in figure 2.

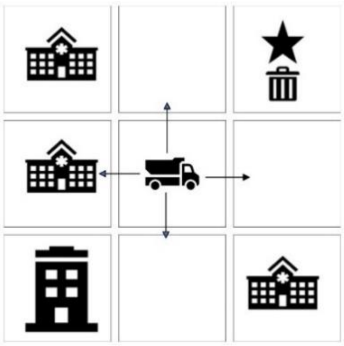

Figure 2. State Diagram inside matrix

Figure 2 presents a sample episode where the agent moves from the current state to collect waste and then proceeds to the terminal state (defined in this matrix at the cell combining the first row and third column), representing the disposal site, to obtain the optimal reward, all while ensuring that the vehicle load does not exceed its maximum capacity.

4.1.2 Actions

For each vehicle v in the set x, the available actions include:

$x = \{\text{left, right up, down, pick} - \text{up, deliver}\}$.

The first four actions indicate the direction for the agent to move from the current state to the next state, while the remaining actions determine whether the agent picks up waste from a hospital or delivers waste to the disposal site when it reaches the next state. The agent attempts to learn the optimal action that leads to the maximum reward. It explores different actions through trial and error to determine the best action from the current state to transition to the next state.

4.1.3 Reward

The primary objective of any agent is to maximize the reward, which implies that the agent aims to reach a solution that is close to optimal. In the study, rewards are classified within a range from negative values to 100. The reward assigned to each action corresponds to how close the resulting state is to the disposal site. Although the goal is to achieve the shortest distance between hospitals, making the objective function a minimization function, the reward for teaching the agent is the highest possible reward, making it a max value. The distance between hospitals is included in a reward table representing the relationship between state and action. This table is represented in the form of an adjacent matrix, where the values represent the distances between hospitals, but inverted, so the value is 1/distance. In this setup, the highest reward is set to a random value of 100, and no distance exceeds 100 in the inverted form. Thus, a balance has been achieved between the minimum objective function and maximum reward.

4.1.4 Time Complexity

The time complexity for a single step in the SARSA approach (updating Q-values) is $O(1)$.Thus, the overall time complexity for running SARSA for n episodes is $O(n * t)$ where, n as the number of episodes, t as the average number of steps per episode.

4.1.5 CSARSA (Capacity-SARSA)

After implementing the SARSA algorithm in the problem, the output resembles individual shortest routes for each node. Each path starts from a specific node and follows the shortest route until reaching the disposal site. The required number of vehicles is then determined to service these distributed paths. The number of vehicles is calculated based on the number of hospitals and the distances between them, considering that each vehicle has a capacity of 3 tons. The result is a distribution of vehicles serving 15 hospitals according to the implemented CVRP using the SARSA algorithm (CSARSA).

4.2 SARSA with Dijkstra ($\boldsymbol{SADJ}$)

Dijkstra's algorithm prioritizes the exploration of route with the shortest accumulated distance and cost. Dijkstra's algorithms estimate as the greedy process as shown in the following equations [26].

$$load \leftarrow dist[u] + Graph.Edges(u, v) \tag{8}$$

$$\textbf{\textit{if}}\ load < dist[v] \tag{9}$$

$$dist[v] \leftarrow load \tag{10}$$

$$prev[v] \leftarrow u \tag{11}$$

$$\textbf{\textit{return}}\ dist[\quad], prev[\quad] \tag{12}$$

Initially, Dijkstra's algorithm was implemented independently for our specific problem, CVRP, but the obtained results were not satisfactory. Dijkstra's algorithm, being a greedy algorithm, makes locally optimal choices at each step. Although effective in many scenarios for finding the shortest path, it does not ensure a globally optimal solution. Recognizing this limitation, the approach was enhanced by combining the strengths of SARSA with the local optimization of Dijkstra's algorithm. The hybrid approach of SARSA with Dijkstra's algorithm (SADJ) synergizes the advantages of both techniques. SARSA, rooted in reinforcement learning, enables the system to learn optimal actions through exploration and exploitation. Meanwhile, Dijkstra's algorithm, a classic graph-based method, focuses on finding the shortest path based on accumulated distances and cost. The performance of the SARSA algorithm aligns with that of the Dijkstra algorithm until the vehicle load reaches its maximum capacity. As the problem's load capacity increases, the SARSA algorithm starts exploring alternative paths to deviate from the shortest route and reduce costs. Combine Q-values and Dijkstra's distances to create hybrid values. This combination could involve adding Q-values to Dijkstra's distances or using a weighted sum. For each episode, use hybrid values to make decisions. Then Choose the action that maximizes the hybrid value for the current state. Continue SARSA and Dijkstra's algorithm iterations until they converge, or a stopping criterion is met. After that, adjust learning parameters to balance exploration and exploitation in

SARSA. Then, evaluate the performance of the hybrid approach against other methods like combining SARSA with the knapsack problem. This approach harnesses the benefits of both algorithms, resulting in a combined complexity comparable to the individual complexities of each method, such as $O(v\ log\ v) + O(n * t))$ where v is the number of vertices (states).

4.3 SARSA with knapsack problem($SAKP$)

The knapsack problem is a classic optimization problem in computer science and mathematics. Modeling our approach as a 0/1 knapsack problem using Dynamic Programming, where a binary choice is made for each hospital's waste - either selecting all waste (1) or excluding all waste from the hospital (0). In the 0/1 knapsack problem, you are given a set of items (hospitals location), each with a weight (maximum vehicle capacity) and a value (distance between hospitals). The goal is to select a subset of these items to maximize the total value, while keeping the total weight within a given capacity (knapsack's weight limit). Each item can either be included (0) or excluded (1), hence the name "0/1" knapsack problem. The challenge is to find the optimal combination of items that maximizes the value while respecting the weight constraint. Create a table (usually a 2D array) with dimensions $[n + 1][W + 1]$, where $'n'$ is the number of hospitals and $'W'$ is the maximum weight capacity of the knapsack (vehicle capacity equal to 3 tons). Initialize all cells to 0. Create a list of $'n'$ items, each with a weight $(w[i])$ and a value mean distance between each hospital $(v[i])$. These items are the ones you want to choose from to maximize the total value within the weight constraint as shown in the following equation [27].

$$\mathbf{dp[i][j] = \max\big(dp[i-1][j], dp[i-1][j - w[i]] + v[i]\big)} \tag{13}$$

where, $\mathrm{dp[i-1][j]}$ represent the maximum value achieved by excluding item i.
$\mathrm{dp[i-1][j - w[i]] + v[i]}$ represents the maximum value achieved by including item i.

Complete the table by considering all items and weight capacities. Initially implemented the Knapsack algorithm independently for our specific problem CVRP. Knapsack algorithms typically assume discrete values for items, which might not accurately model real-world scenarios where items can have continuous or fractional values. Then hybridized SARSA with the knapsack problem (SAKP). This integration combines reinforcement learning with the optimization problem of selecting items to maximize value while staying within a capacity constraint. Here's a general outline of how you could approach this combination: At each state, consider the current state's available items as candidates for selection. Solve the knapsack problem to select a subset of items that maximizes the value while staying within the knapsack's capacity. Use the Q-values to influence the item selection process, favoring actions that are more valuable based on Q-values. To do he Hybridization Mechanism: Combine the results of SARSA and the knapsack optimization to guide decision-making. One approach is to use Q-values to rank candidate items and then use the knapsack solution to make the final selection. Then select the action that leads to the highest expected cumulative reward, considering both Q-values and the knapsack selection. Therefore, while SARSA can aid in optimizing strategies or guiding exploration, the primary time complexity for solving the 0/1 Knapsack problem remains $O(n \times W)$ where n refers to the number of items available in the 0/1 Knapsack problem and w Represents the maximum weight capacity of the knapsack.

5. Study Results

Results were Illustrated by applying five separate mathematical models. In the waste collection schedule, it's essential to note that each hospital receives waste collection services twice a week. Vehicles are tasked with visiting each hospital two times during the week, with the first visit occurring after three days and the second visit after four days. However, there is a unique scenario involving the "National Liver Institute," with an index of 15 node which requires special attention. And this is due to the amount of waste generated during the vehicle visits approaches nearly 3 tons, which is equivalent to the total capacity of a single vehicle, this constitutes a fundamental constraint in the problem. Improved the implementation of the SARSA model by introducing a constraint for vehicle capacity, referred to as Capacity SARSA (CSARSA). This constraint posed a challenge in achieving near to optimal solutions. In Table 2 below, the meanings of the abbreviations used in the results are presented.

Table 2. Abbreviation of used Symbols

Symbol	Abbreviation
V	Number of vehicles
S	Depot (Directorate of Health Affairs)
D	Disposal Site (Kafr Dawood Al-Sadat)
N	Hospital Node index from $(1,2,\ldots,15)$

5.1 SARSA Result

After estimating the SARSA algorithm to optimize this problem, the goal is to find the shortest route between hospitals while considering the amount of waste from each hospital. That means it may take this hospital in a route that achieve shortest route, but the capacity of vehicle reaches to maximum. Thus, this hospital must not be visited. However, another vehicle serves this non-visited hospital to achieve optimal load capacity and the shortest route. This problem aimed at achieving the objectives. CSARSA prioritizes the shortest path when the agent is located at a specific hospital, without considering its relationship with the depot or the hospitals that precede it. Aim to gather distinct routes for each hospital, emphasizing the connections between each hospital in the route, other hospital routes, the depot, and ultimately the disposal site. This process requires optimization to determine the number of vehicles needed, taking into consideration the capacity constraints of the vehicles as shown in Table 3.

Table 3. Shortest route for each vehicle using CSARSA

No.	Vehicle Route in the week	Cap	Vehicle Route in the week	Cap
$\boldsymbol{V_1}$	$S \to 15 \to D \to S$	3	$S \to 15 \to D \to S$	3
$\boldsymbol{V_2}$	$S \to 1 \to 2 \to D \to S$	2.376	$S \to 1 \to 2 \to D \to S$	2.46
$\boldsymbol{V_3}$	$S \to 3 \to 7 \to 8 \to 13 \to D \to S$	2.367	$S \to 3 \to 7 \to 8 \to 13 \to D \to S$	2.175
$\boldsymbol{V_4}$	$S \to 4 \to 9 \to 10 \to D \to S$	1.646	$S \to 4 \to 9 \to 10 \to D \to S$	1.766
$\boldsymbol{V_5}$	$S \to 5 \to 6 \to D \to S$	0.875	$S \to 5 \to 6 \to D \to S$	0.902
$\boldsymbol{V_6}$	$S \to 11 \to 12 \to D \to S$	0.846	$S \to 11 \to 12 \to D \to S$	0.9
$\boldsymbol{V_7}$	$S \to 14 \to D \to S$	0.582	$S \to 14 \to D \to S$	0.657

In Table 3, the first column represents the number of vehicles used to solve the problem with this algorithm. The second column represents the route of each vehicle from its departure from the depot, passing through nearby hospitals, until it reaches the waste disposal site during the first visit of the week. The third column represents the total waste capacity through the vehicle's route for visiting the hospitals specified in the second column. The fourth column represents the vehicle routes during the second visit in the same week, which means three days after the first visit. The fifth column represents the waste capacity generated by the hospitals during the second visit of the week. Seven vehicles were utilized to solve the hospital waste management problem. When programming CSARSA, the output is the shortest route for each hospital. These solutions for each hospital are then stored in a linked list. Then, the pointer was used to navigate between the equal nodes and connect the solutions to each other without repeating the same node twice. The process repeats until the solutions are integrated according to the equality of the specific nodes between different routes. Thus, it was found that 7 vehicles efficiently address this problem. For example in the first (1st) visit , V_2 initiates its weekly route from node S, then moves to node 1, followed by 2 , and finally, disposing of waste collection at node D before returning to the depot. The total capacity of V_2 equal to 2.376 that near to 3 tons. The identical route of V_2 is replicated during the second (2nd) visit, with the only variation being the capacity due to the different weekly amount of wastes from each hospital. V_1 has a unique scenario where it serves the National Liver Institute hospital (node 15), and the vehicle visits it three times a week due to the total capacity reaching approximately 8.5 tons. The algorithm might get stuck in local optima (suboptimal) and fail to explore less obvious, but better, solutions. All of the mentioned points are considered as disadvantages of this approach. To achieve an optimal solution for CVRP, it is recommended to combine SARSA with Dijkstra's algorithm or integrating it with the Knapsack problem solver. This integration aims to refine the solutions obtained from SARSA and enhance capacity utilization, ultimately leading to an optimal routing solution as presented in the upcoming sections.

5.2 Combination between SARSA and Dijkstra Result ($SADJ$)

Dijkstra's algorithm provides deterministic and guaranteed optimal solutions. When dealing with capacity constraints in CVRP, ensuring that the initial routes are optimal in terms of distance can be crucial as shown in Table 4 by using only five vehicles.

Table 4. optimal route for each vehicle using Dijkstra

No.	Vehicle Route in the week	Cap	Vehicle Route in the week	Cap
V_1	$S \to 15 \to D \to S$	3	$S \to 15 \to D \to S$	3
V_2	$S \to 2 \to 13 \to D \to S$	2.095	$S \to 2 \to 13 \to D \to S$	1.961
V_3	$S \to 1 \to 9 \to 10 \to 11 \to 12 \to D \to S$	2.32	$S \to 1 \to 9 \to 10 \to 11 \to 12 \to D \to S$	2.517
V_4	$S \to 4 \to 5 \to 6 \to 3 \to 7 \to D \to S$	2.934	$S \to 4 \to 5 \to 6 \to 3 \to 7 \to D \to S$	2.946
V_5	$S \to 8 \to 14 \to D \to S$	1.343	$S \to 8 \to 14 \to D \to S$	1.436

Table 4 illustrates the implementation of the Dijkstra algorithm for determining the shortest route in CVRP. For instance, V_2 initiates its weekly route from depot node S, then moves to node 2, followed by 13, and finally, it collects waste at node D before returning to the depot. It's crucial to highlight that the capacity is 2.095, falling short of the complete 3 tons' approximation. Similarly, the rest of the cases follow the same explanation expect V_1 ,which visits the National Liver Institute node. This is attributed to Dijkstra's prioritization of the shortest route when confronted with the choice between maximum load capacity and the shortest path. So, this is a significant limitation that discourages relying on Dijkstra in isolation. This combination of SADJ proves especially potent due to their complementary attributes. This combination can ensure that you exploit the known shortest paths while still allowing for exploration to discover potentially better paths. Table 5 illustrates the shortest route between hospitals using 5 vehicles while considering the vehicle's capacity constraints and the total capacity in tons for each vehicle route.

Table 5. optimal route for each vehicle with total capacity using SADJ.

No.	Vehicle Route in the week	Cap	Vehicle Route in the week	Cap
V_1	$S \to 15 \to D \to S$	3	$S \to 15 \to D \to S$	3
V_2	$S \to 1 \to 2 \to 13 \to D \to S$	2.999	$S \to 1 \to 2 \to 13 \to D \to S$	2.918
V_3	$S \to 4 \to 9 \to 11 \to 12 \to 7 \to D \to S$	2.914	$S \to 4 \to 9 \to 11 \to 12 \to 7 \to D \to S$	2.863
V_4	$S \to 10 \to 5 \to 6 \to 8 \to 3 \to 14 \to D \to S$	2.779	$S \to 10 \to 5 \to 6 \to 8 \to 3 \to D \to S$	2.422
V_5	________________	3	$S \to 14 \to D \to S$	0.657

Table 5 illustrate our estimations indicate the need for a fleet of five homogeneous vehicles to efficiently serve the waste disposal needs of the sixteen hospitals. For example, V_2 follows a route from depot node S to node 1, then node 2, followed by node 13, and subsequently progresses towards node D. This represents the optimized shortest route for vehicle V_2. Then V_2 stops at node 13 instead of completing its route to search another nearest hospital. The vehicle reaches its maximum capacity, which is nearly 3 tons (2.999 tons). V_2 initiates its first visit to hospitals after three days in the week, but the week remains unfinished. This next visit may include revisiting some hospitals or neglecting others, contingent on the vehicle's capacity. V_2 subsequent visit occurs after four days from the initial one, with a total capacity of 2.918 tons. This variation in total capacity stems from the differences in the amount of waste extracted during each collection center. The contrast between the shortest routes followed by V_4 and V_5, in the case of V_4 does not revisit the same hospital during its second visit, primarily because the amount of waste at hospital 14, if added to the previous nodes in the same route, would surpass the vehicle's capacity limit of 3 tons. Consequently, another dedicated vehicle is assigned to serve these additional waste collection needs. SADJ might find a solution quickly, but it may not be the best one. To overcome this problem, SARSA is integrated with knapsack dynamic programming.

5.3 Combination of Q-learning and knapsack solving Result ($QLKP$)

Implementing Knapsack using Dynamic Programming is a powerful technique for solving the 0/1 knapsack problem optimally, as shown in Table 6.

Table 6. optimal route for each vehicle using Knapsack problem

No.	Vehicle Route in the week	Cap	Vehicle Route in the week	Cap
V_1	$S \rightarrow 15 \rightarrow D \rightarrow S$	3	$S \rightarrow 15 \rightarrow D \rightarrow S$	3
V_2	$S \rightarrow 1 \rightarrow 2 \rightarrow 13 \rightarrow D \rightarrow S$	2.999	$S \rightarrow 1 \rightarrow 2 \rightarrow 13 \rightarrow D \rightarrow S$	2.918
V_3	$S \rightarrow 4 \rightarrow 5 \rightarrow 7 \rightarrow 14 \rightarrow D \rightarrow S$	2.92	$S \rightarrow 4 \rightarrow 5 \rightarrow 7 \rightarrow 14 \rightarrow D \rightarrow S$	2.985
V_4	$S \rightarrow 10 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 6 \rightarrow 3 \rightarrow 8 \rightarrow D \rightarrow S$	2.773	$S \rightarrow 10 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 6 \rightarrow 3 \rightarrow 8 \rightarrow D \rightarrow S$	2.957

Table 6 illustrates the implementation of the knapsack algorithm to determine the shortest route in CVRP by using only four vehicles. For instance, V_2 commences its weekly route from node S, progresses to node 1, then to node 2, followed by 13, and finally, move to node D. Significantly, the capacity is 2.999 nearly 3 tons. This pattern is repeated in the second visit of the week. The remaining cases follow a similar explanation, except for V_1 that visit National Liver Institute hospital. Then implement SARSA with the Knapsack Problem (SAKP). Dynamic programming algorithms guarantees an optimal solution because they consider all possible combinations of sub problems and systematically find the best solution based on previously computed sub problem results according to SARSA exploration to discover optimal policies through trial and error as shown in Table 7.

Table 7. optimal route for each vehicle using SAKP.

No.	Vehicle Route in the week	Cap	Vehicle Route in the week	Cap
V_1	$S \rightarrow 15 \rightarrow D \rightarrow S$	3	$S \rightarrow 15 \rightarrow D \rightarrow S$	3
V_2	$S \rightarrow 1 \rightarrow 2 \rightarrow 13 \rightarrow D \rightarrow S$	2.999	$S \rightarrow 1 \rightarrow 2 \rightarrow 13 \rightarrow D \rightarrow S$	2.918
V_3	$S \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 3 \rightarrow D \rightarrow S$	2.934	$S \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 3 \rightarrow D \rightarrow S$	2.946
V_4	$S \rightarrow 10 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 9 \rightarrow 14 \rightarrow D \rightarrow S$	2.759	$S \rightarrow 10 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 9 \rightarrow 14 \rightarrow D \rightarrow S$	2.996

Table 7 showcasing the seamless integration of SAKP to address the prevailing challenge. Our estimations indicate the need for a fleet of only four homogeneous vehicles to efficiently serve the waste disposal needs of the sixteen hospitals. For example V_2 follows a route from collection center node S to node 1, then node 2, followed by node 13, and subsequently progresses towards node D. This represents the optimized shortest route for vehicle V_2. Then V_2 stops at node 13 instead of completing its route to another hospital. The vehicle reaches its maximum load capacity, which is nearly 3 tons (2.999 tons). During V_2 second visit of the week, it revisit the same hospitals with different load capacity 2.918 tons. It's important to note that each of the homogeneous vehicles has now approached a near to maximum capacity unlike the previous mathematical model. The knapsack dynamic programming approach, or SAKP, utilized only four vehicles, in contrast to Dijkstra, SADJ, or CSARSA.

6. Discussion

The differences between the algorithms presented above are discussed in the results section. The implementation of SARSA to serve our problem with respect to capacity vehicle routing problem. But there is some disadvantage, because CVRP can involve a large number of possible states and actions, leading to a high-dimensional Q-table. This can make learning and convergence computationally expensive and slow. SARSA involves a trade-off between exploration (trying new routes) and exploitation (choosing the best-known routes). Striking the right balance can be challenging, especially when the number of possible routes is vast. Comparing the five algorithms, it is observed that CSARSA requires 7 vehicles in week one to serve hospital wastes based on vehicle capacity and the amount of waste at each hospital. But Dijkstra and SADJ need from 4 to 5 vehicles in week one to serve wastes of hospitals. But Knapsack and SAKP need only 4 vehicles to serve the wastes of hospitals. If solving the problem using the maximum number of vehicles obtained from the implemented approach (CSARSA), which is seven, and then assigning V_1 as the name of the vehicle serving node 15 for each approach, considering the number of vehicles already in use in each approach

A reduction in the number of vehicles used signifies an enhancement brought about by the proposed mathematical model. Opting for a smaller fleet of vehicles offers distinct preferable, because it saves on vehicle costs, driver expenses, vehicle maintenance, and fuel costs. It should be highlighted that with SAKP, the load capacity of vehicles consistently approaches 3 tons in all cases, contrasting with SADJ and CSARSA. All of these factors can be a burden on the government's budget. Therefore, reducing these expenses is advantageous for the budget of the Ministry of Health. Therefore, SAKP is a superior approach compared to SADJ and CSARSA. When the homogeneous fleet of vehicles selects the shortest route for a group of hospitals, it becomes crucial to estimate both the total distance traveled by each vehicle and the associated time in five algorithms as illustrated in Figure 3,4,5 and 6.

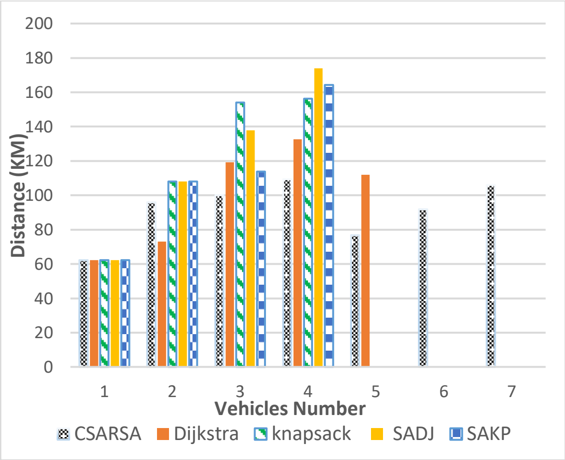

Figure 3. Distance estimated per vehicles in 1^{st} visit

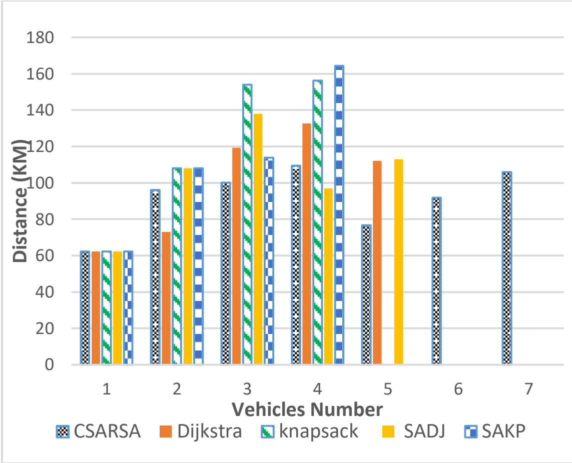

Figure 4. Distance estimated per vehicles in 2^{nd} visit

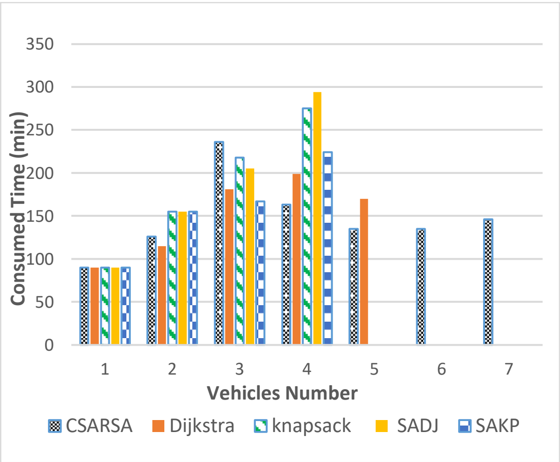

Figure 5. Time consumed per vehicles in 1^{st} visit

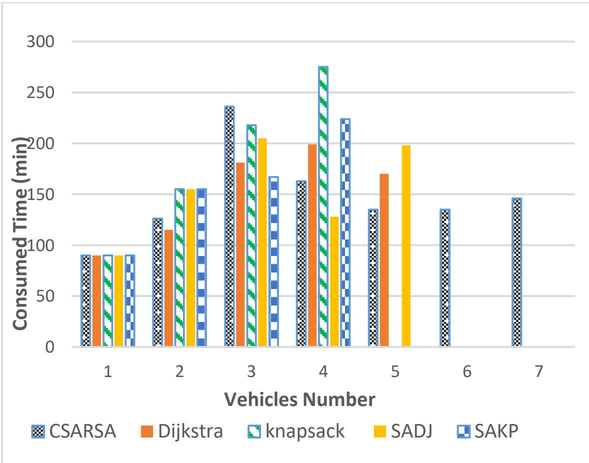

Figure 6. Time consumed per vehicles in 2^{nd} visit

Figures (3, 4, 5 and 6) depict the total distance per kilometer and the total time per minuets covered by each vehicle during both the initial and subsequent visits within the same week. In the two figures, observe the number of vehicles visiting hospitals to collect waste while achieving the shortest route. These vehicles start their journey from the depot, follow the most efficient route, reach the disposal site, and then return to the depot based on the applied algorithm. To estimate the total distance and time traveled between each node, Google Maps was utilized. In Figure (3) and (4) the outcomes of implementing five algorithm individually remained consistent during the first and second visits. Both methods prioritize finding an optimal solution based on capacity, lacking the trial-and-error iterations seen in SARSA that enhance the solution by aiming to achieve optimal capacity with the shortest route. In Figure (3), SAKP stands out as the superior option due to its minimized distance consumption. In Figure (4) show total distance in the 2nd visit. It is noted that v_4 has the longest recorded distance. However, this is not a drawback as it contributes to the reduction in the number of vehicles used. It efficiently collects the remaining available waste from hospitals along its route instead of employing another vehicle. In the case of v_1 the distance recorded remains consistent across all five applied algorithms. This uniformity is due to the special circumstances of node 15, where the vehicle makes three visits in the week, determined by the amount of extracted waste. The distance covered by each approach's vehicle corresponds to the time consumed for it. Figure (5) and (6) illustrated the time consumed in the first and the second visit of each vehicle based on applied approach according to the estimated distance. In conclusion, SAKP stands out as the best choice in all visitation scenarios, utilizing the minimal number of vehicles, achieving fuel savings, covering the shortest distance, and consuming less time overall. The Ministry of Health enters into agreements with companies to lease a group of vehicles for waste transportation services from hospitals to the waste disposal site in Kafr Dawood Al-Sadat. It's worth noting that when these vehicles make trips to hospitals, they calculate their charges based on the weight of the waste carried. The cost per ton is a fixed value equal to

1$. This amount is considered the cost for handling and transporting the load to the incineration facility as shown in Figure 7 and 8.

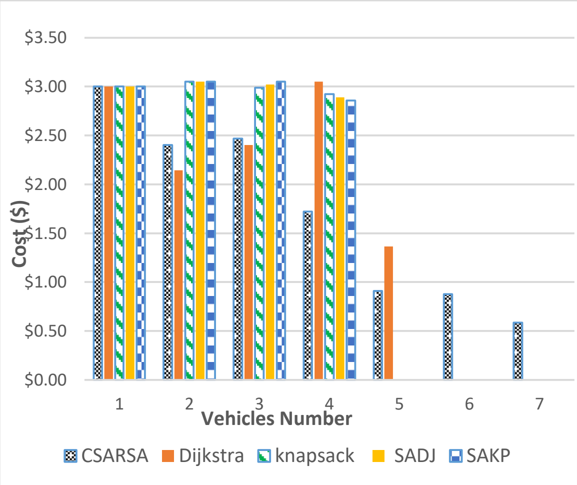

Figure 7. cost for each vehicles in $\mathbf{1}^{st}$ visit

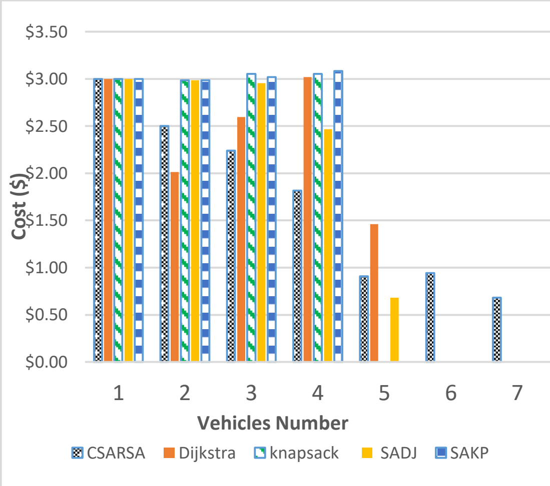

Figure 8. cost for each vehicles in $\mathbf{2}^{nd}$ visit

In Figures (7) and (8), the total cost of transporting hospital wastes to the disposal site is depicted twice a week. As mentioned earlier v_1 is special case. For example v_2 has the same route and cost in Knapsack, SADJ and SAKP. In the first and second visit v_2 follows the shortest route while considering its load capacity, resulting in a total cost of 3.05 $ and 2.99 $ for both the knapsack ,SADJ and SAKP respectively in contract Dijkstra and CSARSA. For instance, the shortest route, which goes from depot to Quwisna General Hospital, then to Qasr Hospital, followed by Al-Shuhada Central Hospital, and finally to the disposal site, corresponds to specific amounts of waste the first visit for each hospital as follows: $0 \to 0.904 \to 1.472 \to .623 \to 0 \to 0$ per ton. The total cost is calculated as $0 \to 0.909 \to 1.493 \to 0.649 \to 0 \to 0$ per$. However, during the second visit, the vehicle still adheres to the same optimal route, with each hospital having distinct capacities, as detailed: $0 \to 0.957 \to 1.503 \to .458 \to 0 \to 0$ per ton. The total cost is calculated as $0 \to 0.974 \to 1.525 \to 0.487 \to 0 \to 0$ per$. But in the first visit, the shortest route of Dijkstra $0 \to 1.472 \to .623 \to 0 \to 0$ per ton. The total cost is calculated as $0 \to 1.493 \to 0.649 \to 0 \to 0$ per$. However, during the second visit, hospitals capacity as detailed: $0 \to 1.503 \to .458 \to 0 \to 0$ per ton. The total cost is calculated as $0 \to 1.525 \to 0.487 \to 0 \to 0$ per$. But in the first visit, the shortest route of CSARSA $0 \to 0.904 \to 1.472 \to 0 \to 0$ per ton. The total cost is calculated as $0 \to 0.909 \to 1.493 \to 0 \to 0$ per$. However, during the second visit, hospitals capacity as detailed:$0 \to 0.957 \to 1.503 \to 0 \to 0$ per ton. The total cost is calculated as$0 \to 0.974 \to 1.525 \to 0 \to 0$ per$.

7. Conclusions

Medical waste poses a significant environmental burden in terms of pollution and disease transmission. Economically, it requires incineration. These incinerators should be located in secure locations away from populated areas to protect public health. Other factors to consider include the need for specific types of transport vehicles designated by the Ministry of Health, as well as the requirement for well-trained drivers to handle waste and transport it from hospitals to disposal sites, especially in cases of injuries or external factors during transport. These vehicles have a fixed capacity, typically set at 3 tons by the Ministry of Health. The vehicles follow specific routes to collect waste from hospitals, with a priority given to nearby hospitals before moving on to dispose of the collected waste. These issues are classified as the vehicle routing problem with capacity constraints. Three distinct algorithms were employed: SARSA, an integration of SARSA and Dijkstra, and an integration of SARSA with a knapsack solver. The SAKP model shows that the vehicles take the shortest routes between hospitals, reach maximum loads close to 3 tons, minimize costs, reduce time, and use fewer vehicles compared to other mathematical models. For future work, when dealing with a larger number of hospitals or large-scale problems, alternative learning methods should be considered besides SARSA and Q-learning. These methods are necessary because they often do not yield accurate results, and the agent's learning may not be precise in most cases. Therefore, using other artificial intelligence algorithms, such as deep neural networks, Double DQN, DDQN and deep learning approaches could be beneficial. Additionally, time windows can be incorporated to prioritize hospital visits and determine the optimal time for waste collection. Furthermore, a model can be developed by dividing hospitals into zones and utilizing vehicles with different capacities, enabling the sharing of cluster zones.

References

[1] E. A. Fawkia, M. A. Mahmoud, M. F. Nehal and A. Gehad, "Evaluation of hazardous medical wastes management system in great cairo governorates," Journal of Environmental Science, vol. 46, p. 1–250, 2019.

[2] M. Attrah, A. Elmanadely, D. Akter and E. R. Rene, "A review on medical waste management: treatment, recycling, and disposal options," Environments, vol. 9, p. 146, 2022.

[3] H.-L. Zhao, L. Wang, F. Liu, H.-Q. Liu, N. Zhang and Y.-W. Zhu, "Energy, environment and economy assessment of medical waste disposal technologies in China," Science of The Total Environment, vol. 796, p. 148964, 2021.

[4] M. M. Gharib, G. M. Elkhouly and M. S. Bokhary, "WASTE MANAGEMENT AND INFECTION CONTROL IN HEALTH CARE ORGANIZATIONS. STRATEGIES," Journal of Environmental Science, vol. 50, p. 599–637, 2021.

[5] P. Toth and D. Vigo, Vehicle routing: problems, methods, and applications, SIAM, 2014.

[6] Z. Borcinova, "Two models of the capacitated vehicle routing problem," Croatian Operational Research Review, p. 463–469, 2017.

[7] Z.-H. Zhou, "Open-environment machine learning," National Science Review, vol. 9, p. nwac123, 2022.

[8] N. Mazyavkina, S. Sviridov, S. Ivanov and E. Burnaev, "Reinforcement learning for combinatorial optimization: A survey," Computers & Operations Research, vol. 134, p. 105400, 2021.

[9] A. Bdeir, S. Boeder, T. Dernedde, K. Tkachuk, J. K. Falkner and L. Schmidt-Thieme, "RP-DQN: An application of Q-learning to vehicle routing problems," in KI 2021: Advances in Artificial Intelligence: 44th German Conference on AI, Virtual Event, September 27–October 1, 2021, Proceedings 44, 2021.

[10] A. Mesquita, Y. Nogueira, C. Vidal, J. Cavalcante-Neto and P. Serafim, "Autonomous foraging with sarsa-based deep reinforcement learning," in 2020 22nd Symposium on Virtual and Augmented Reality (SVR), 2020.

[11] H. Wei, S. Zhang and X. He, "Shortest path algorithm in dynamic restricted area based on unidirectional road network model," Sensors, vol. 21, p. 203, 2020.

[12] P. Udhan, A. Ganeshkar, P. Murugesan, A. R. Permani, S. Sanjeeva and P. Deshpande, "Vehicle Route Planning using Dynamically Weighted Dijkstra's Algorithm with Traffic Prediction," arXiv preprint arXiv:2205.15190, 2022.

[13] R. Campos, P. Munari and L. C. Coelho, “Compact formulations for the robust vehicle routing problem with time windows under demand and travel time uncertainty”, Faculté des sciences de l'administration, Université Laval, 2022.

[14] G. N. Fauziah, "Application Of The Greedy Algorithm In Multiple Constrain Knapsack Optimization Problems In Goods Transportation," Jurnal Syntax Admiration, vol. 3, p. 725–732, 2022.

[15] S. Coniglio, F. Furini and P. San Segundo, "A new combinatorial branch-and-bound algorithm for the knapsack problem with conflicts," European Journal of Operational Research, vol. 289, p. 435–455, 2021.

[16] C. Hojny, T. Gally, O. Habeck, H. Lüthen, F. Matter, M. E. Pfetsch and A. Schmitt, "Knapsack polytopes: a survey," Annals of Operations Research, vol. 292, p. 469–517, 2020.

[17] N. Alqahtani and F. Nadeem, "Improving the effectiveness of e-learning processes through dynamic programming: A survey," International Journal of Advanced Computer Science and Applications, vol. 12, p. 818–825, 2021.

[18] A. A. Ibrahim, N. Lo, R. O. Abdulaziz and J. A. Ishaya, Capacitated vehicle routing problem, Éditions universitaires européennes, 2019.

[19] S. Zhang, T. Wu, M. Pan, C. Zhang and Y. Yu, "A-SARSA: A predictive container auto-scaling algorithm based on reinforcement learning," in 2020 IEEE international conference on web services (ICWS), 2020.

[20] S. Reza, M. C. Ferreira, J. J. M. Machado and J. M. R. S. Tavares, "A citywide TD-learning based intelligent traffic signal control for autonomous vehicles: Performance evaluation using SUMO," Expert Systems, p. e13301, 2023.

[21] Y. Shi, Z. Yuan, X. Zhu and H. Zhu, "An adaptive routing algorithm for inter-satellite networks based on the combination of multipath transmission and Q-Learning," Processes, vol. 11, p. 167, 2023.

[22] A. A. Way, Egypt demographic and health survey, 2005, Ministry of Health and Population, 2006.

[23] S. Elnakib, M. Elsallab, M. A. Wanis, S. Elshiwy, N. P. Krishnapalan and N. A. Naja, "Understanding the impacts of child marriage on the health and well-being of adolescent girls and young women residing in urban areas in Egypt," Reproductive health, vol. 19, p. 8, 2022.

[24] Z. Borčinová, "Kernel search for the capacitated vehicle routing problem," Applied Sciences, vol. 12, p. 11421, 2022.

[25] W. Qiang and Z. Zhongli, "Reinforcement learning model, algorithms and its application," in 2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC), 2011.

[26] A. A. Zubir, A. Andriansyah and S. Derisma, "Determining the Distribution Route Using the Clustered Generalised Vehicle Routing Problem Model and Dijkstra's Algorithm," Journal of Science, Technology, and Visual Culture, vol. 2, p. 232–240, 2022.

[27] E. Angelelli, R. Mansini and M. G. Speranza, "Kernel search: A general heuristic for the multi-dimensional knapsack problem," Computers & Operations Research, vol. 37, p. 2017–2026, 2010.

المجلد (..) العدد (..) (السنة)

المجلة الدولية للحاسبات والمعلومات

متاح على الإنترنت على الرابط : https://ijci.journals.ekb.eg/

تحسين جمع النفايات الطبية بواسطة التعلم التعزيزي ضمن توجيه المركبات ذات السعة المتجانسة

نورهان خلاف[1] , أسامه عبدالرؤوف[1] , محي هدهود[2] , أحمد كفافي[3]

قسم ذكاء الألة – كلية الذكاء الاصطناعي – جامعة المنوفية[1]
قسم تكنولوجيا المعلومات– كلية الحاسبات والمعلومات – جامعة المنوفية[2]
قسم علوم البيانات – كلية الذكاء الاصطناعي – جامعة المنوفية[3]

nourhan.khalaf@ci.menofia.edu.eg, osamaabd@ci.menofia.edu.eg, mmhadhoud@ci.menofia.edu.eg, *ahmedkafafy80@gmail.com*

ملخص البحث

الذكاء الاصطناعي يُستخدم بشكل متزايد في مجالات متعددة، بما في ذلك إدارة النفايات الطبية الخطرة. تشكل النفايات الطبية عبئًا اقتصاديًا ومخاطرًا على الصحة العامة، ويجب التخلص منها بحذر، ويفضل أن يكون ذلك في مناطق بعيدة عن المناطق السكنية. تم جمع بيانات حول النفايات الناتجة من 15 مستشفى حكومي في محافظة المنوفية وموقع التخلص الوحيد في كفر داود، بالإضافة إلى نقطة تجميع مركزية لمركبات نقل النفايات. تعالج هذه الدراسة مشكلة توجيه المركبات ذات السعة المحدودة، والتي تعتبر مشكلة (صعبة من الناحية الحسابية) معقدة. تم تخصيص مركبات محددة لجمع النفايات من المستشفيات ونقلها إلى مركز التخلص، بهدف إيجاد أقصر طريق مع تحقيق أقصى استفادة من سعة المركبة، التي تقتصر على ثلاثة أطنان. تم تطوير تقنيات التعلم التعزيزي، حيث تم معاملة المركبة كوكيل مدرب على اختيار أقصر وأقل طريق تكلفة بين المستشفيات. تم تنفيذ وتحسين خوارزمية "سارسا" (خوارزمية التعلم الذاتي). تشمل الحلول خوارزميات "سارسا" (خوارزمية التعلم الذاتي) و"ديكسترا" و"برمجة ديناميكية للحقيبة"، ونهجًا هجينًا يجمع بين "سارسا" و"ديكسترا" و"سارسا" مع "برمجة ديناميكية للحقيبة". أظهرت النتائج أن النهج الهجين بين "سارسا" و"برمجة ديناميكية للحقيبة" هو الأكثر فعالية، حيث يقلل من عدد المركبات المستخدمة لنقل النفايات ويزيد من سعة المركبة، مما يحدد أقصر الطرق بين جميع المستشفيات. وأخيرًا، تم حساب تكاليف النقل لاستكمال النموذج الرياضي لإدارة النفايات الطبية .
