QATAR UNIVERSITY

COLLEGE OF ENGINEERING

A DEEP REINFORCEMENT LEARNING-ENABLED DYNAMIC REDEPLOYMENT SYSTEM FOR MOBILE AMBULANCES IN QATAR

BY

REEM BASSAM TLULI

A Thesis Submitted to

the College of Engineering

in Partial Fulfillment of the Requirements for the Degree of

Masters of Science in Computing

June 2024

© 2024. Reem Bassam Tluli. All Rights Reserved.

COMMITTEE PAGE

The members of the Committee approve the Thesis of

Reem Bassam Tluli defended on 01/05/2024.

Dr. Saeed Mousa Ahmed Salem
Thesis Supervisor

Dr. Ahmed Badawy
Thesis Co-Supervisor

Dr. Committee One
Committee Member

Dr. Committee Two
Committee Member

Dr. Committee Three
Committee Member

Approved:

Khalid Kamal Naji, Dean, College of Engineering

ABSTRACT

TLULI, REEM, B.A.I., Masters : June : 2024, Masters of Science in Computing

Title: A Deep Reinforcement Learning-Enabled Dynamic Redeployment System for Mobile Ambulances in Qatar

Supervisor of Thesis: Dr. Saeed Mousa Ahmed Salem.

Efficient allocation of ambulances is crucial for Emergency Medical Services (EMS) to respond promptly and deliver life-saving care on time. The challenge lies in the ambulance redeployment problem, which aims to devise optimal deployment strategies to minimize response times and maximize coverage in a given area.

Traditional approaches to ambulance allocation problem rely on heuristics and predefined rules, often struggling to adapt to the dynamic nature of emergencies. In response, this thesis proposes a dynamic ambulance redeployment system to reduce ambulance response time, thus increasing the chances of saving lives.

When an ambulance becomes available, the system recognizes it and intelligently reallocates it to the appropriate ambulance station, including ambulances that have come via patient transfers. By doing this, ambulance stations become more equipped to handle crises in the future. It is necessary to take into account several dynamic factors at each station concurrently due to the complexity of this operation. It is almost hard to control these elements with manual rules. We propose combining and prioritizing the dynamic factors of each station into a single score by means of a DNN, which we term the deep score network, in order to overcome this complexity. Through the utilization of DNN, we propose a Deep Reinforcement Learning (DRL) framework that efficiently trains the deep scoring network. Our dynamic ambulance redeployment algorithm is presented here for real-world applications based on this learning. Similarly, we apply the

proposed framework on dynamic Charlie vehicle redeployment. Experimental results on real-world data from Qatar EMS show that our method clearly outperforms the state-of-the-art baseline methods. For example, for dynamic ambulance redeployment, the average response time of patients can be reduced by $\sim$ 100 seconds (20%) with our proposed method, and the percentage of patients picked up within 10 minutes can be improved from 64.8% to 79.8%. As for the dynamic redeployment of Charlie vehicles, the average response time of critical patients can be reduced by $\sim$ 125 seconds (13.33%) with our proposed method, and the percentage of critical patients treated within 10 minutes can be improved by approximately 11.08%.This improvement leads into more effective rescue operations for people in danger.

DEDICATION

To my homeland Palestine, the land of resilience and beauty;

The great martyrs and prisoners, the symbol of sacrifice;

To my family and friends for their unconditional love, support and faith in me

ACKNOWLEDGMENTS

I am deeply grateful to my parents and siblings for their unwavering support and encouragement throughout my academic journey. Their belief in me and their constant encouragement have been a source of strength and motivation.

I would also like to express my heartfelt gratitude to my advisors, Dr. Saeed Salem and Dr. Ahmed Badawy, for their exceptional guidance, support, and dedication. Their profound knowledge, insightful scientific vision, and resourceful guidance have been instrumental in shaping my research and navigating the challenges of my Master's thesis. Their mentorship has not only enriched my academic experience but has also inspired me to strive for excellence in my future endeavors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

It can be the difference between life and death to provide medical attention as soon as possible in the event of an accident or other life-threatening emergencies. Ambulances and ambulance staff are among the most significant providers of care in such situations. These are the individuals who respond to emergency situations, administer essential acute care, and transfer patients to a hospital for further treatment. After a call, an ambulance should typically reach the location in 15 minutes [1]. This target should be met in at least 95% of cases involving so-called A1 calls, which include life-threatening conditions. Figure 1 depicts a dose-response curve illustrating the relationship between emergency response time (X-axis, in minutes) and the odds of mortality (Y-axis, on the natural log scale). The curve was generated using locally weighted kernel smoothing, allowing for a visually smoothed representation of trends. The curve's shape provides insights into how the odds of mortality change with varying response times, highlighting critical points or trends in the relationship. It serves as a visual tool to analyze the impact of response time on mortality rates. It shows that as the response time extends beyond the initial 5 minutes, there's an escalation in the probability of mortality.

Rapid response and efficient ambulance deployment are critical factors in saving lives during medical emergencies [2]. The ability to accurately predict where ambulances should be stationed can significantly reduce response times, ensuring timely medical assistance for individuals in need. However, determining optimal ambulance locations is a complex problem that involves numerous variables, such as population density, traffic patterns, geographical features, and historical incident data.

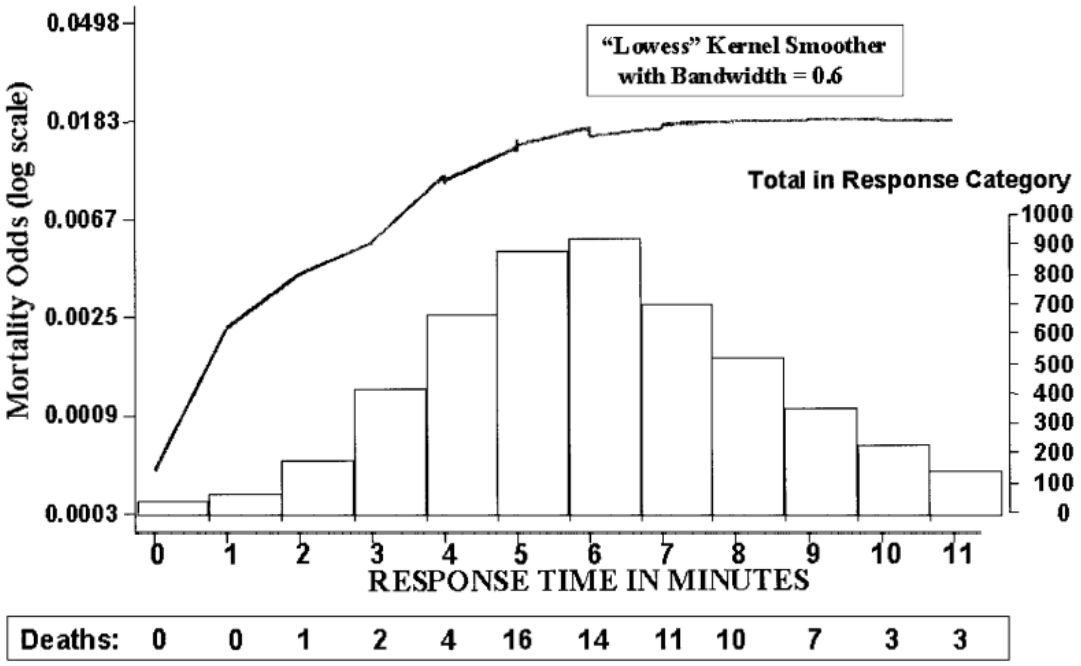


Figure 1.1. Smoother mortality odds by EMS [3]

1.1. Problem Significance

The significance of this project extends beyond mere technological innovation; it holds the potential to revolutionize emergency response services in Qatar. With a population of approximately 2.9 million inhabitants [4], Qatar faces the critical challenge of ensuring timely and efficient healthcare delivery to its residents and visitors. Primary Healthcare Corporation (PHC) and Hamad Medical Corporation (HMC) stand as the pillars of public healthcare in Qatar, continually evolving to meet the increasing demands of the population.

One indispensable component of Qatar's healthcare ecosystem is the HMC Ambulance Service (HMCAS), established in 1985 [5], which plays a pivotal role in providing lifesaving care through a spectrum of emergency and non-emergency pre-hospital services. Despite the commendable efforts of HMCAS, there remains a pressing need to further optimize emergency response mechanisms, particularly in the realm of ambu-

lance redeployment.

The average response time of HMC Ambulance Service to emergency calls, though faster than the targets set by Qatar's National Health Strategy in 2011, still presents room for improvement. In 2018 alone, HMCAS responded to approximately 115,000 priority one calls, achieving an average response time of approximately 13 minutes within the capital, Doha and 15.2 minutes in rural areas [6]. While these response times are commendable, even minor reductions could translate into significant enhancements in patient outcomes, especially in critical situations where every second counts.

By leveraging Machine Learning (ML) algorithms to optimize ambulance placements, this project aims to refine response times further, ensuring swifter medical assistance to those in need. The deployment of advanced algorithms holds the promise of more efficient resource allocation, leading to reductions in mortality rates and enhancements in overall emergency preparedness.

Moreover, the implementation of such innovative solutions aligns perfectly with Qatar's commitment to advanced healthcare solutions and its vision for a modern and responsive healthcare system. By embracing cutting-edge technologies and methodologies, Qatar can solidify its position as a leader in healthcare innovation, setting new standards for EMS not only within the region but on a global scale.

The significance of this thesis transcends its immediate scope, offering a pathway towards a future where emergency response services in Qatar are not just effective but exemplary. Through the synergy of technology, data, and healthcare expertise, improvement in response time will make a tangible difference in the lives of Qatar's residents and visitors, safeguarding their well-being and prosperity for generations to

come.

1.2. How the EMS System Works

The sequence of the steps that the EMS system follows once a request is made is illustrated in Figure 1.2:

1. An incident occurs, prompting a call to the EMS for assistance.
2. A dispatcher assesses the incident's priority level.
3. The nearest ambulance that is available is sent by the dispatcher to the scene of the event.
4. When the ambulance gets there, it either treats the patient there or transports them to a hospital.
    (a) If the patient requires further treatment, the ambulance transports them to the nearest hospital.
5. After completing the patient's treatment and hospital delivery, the ambulance is redeployed to a selected spoke station for its next assignment.

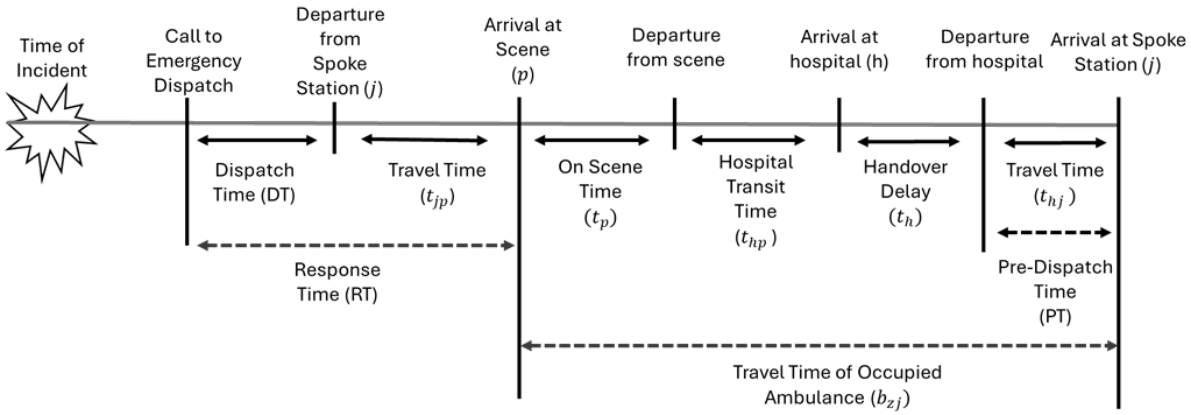


Figure 1.2. An outline of the EMS system's process. It is assumed that every occurrence results in hospitalization.

Throughout the thesis, several definitions will be used extensively. Therefore, to improve the readability of the thesis, we provide a list of definitions used in the EMS system:

1. **Dispatch Time** (DT) is the amount of time that passes between receiving a call and dispatching an ambulance.

2. **Spoke Station** refers to a specific facility within the EMS network where ambulances are stationed or dispatched from. Each spoke station is identified by its index, j.

3. **Patient** A patient refers to a specific patient requesting an EMS. Each patient is identified by a unique index p.

4. **Travel Time** (t_{jp}) is the time it takes to travel from spoke station j to patient location p.

5. **Response Time** (RT) is, as stated in Equation 1.1, the amount of time that passes between when EMS gets a call and when an ambulance arrives at the patient's location.

$$RT = DT + t_{jp} \tag{1.1}$$

6. **Hospital Transit Time** (t_{ph}) is the amount of time needed to get a patient from the scene of the incident to the hospital.

7. **Handover Delay** (t_h) is the delay that occurs when transferring a patient from the ambulance to the hospital staff and paperwork, etc.

8. **Pre-Dispatch Time** (PT) is the duration between when an ambulance becomes available and reaches the spoke station it is redeployed to.

9. **On Scene Time** (t_p) is the duration an ambulance spends at the incident location.

10. **Number of picked-up patients within a given time** (PU) To calculate the total number of patients picked up within a time threshold, τ, we used the following approach: Let PU be the total number of patients picked up within a time threshold by all ambulances.

$$\begin{aligned} PU &= \sum_{p=1}^{P} Pickup_p \\ Pickup_p &= \begin{cases} 1, & \text{if } RT + t_p + t_{hp} \leq \tau \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{1.2}$$

where $Pickup_p$ is an indicator binary variable that equals 1 if any ambulance picks up patient (p) within τ threshold. The indicator binary variable is 1 if the sum of the response time (RT), time at scene (t_p) and travel time from patient to hospital (t_{hp}) is within the threshold τ, and it is 0 otherwise.

11. **Dispatching** is the responsibility of deciding which ambulance to send to an incident.

12. **Redeployment** is the process of reallocating or reassigning available ambulances from one location to a spoke station.

1.3. The Fleet of HMCAS in Qatar

HMC in Qatar operates a cutting-edge ambulance service, featuring a fleet of state-of-the-art vehicles and helicopters, headquartered in Doha. These vehicles are manned by skilled paramedics and emergency care doctors who provide critical first aid to accident victims and individuals facing medical emergencies before swiftly transporting them to the appropriate emergency department [7].

The ambulance fleet is designed for efficient navigation through traffic, ensuring quick access to emergency cases, and is environmentally friendly, contributing to optimized emergency response times. The vehicles boast advanced interiors, equipped with modern technologies, adequate lighting, and smart air-conditioning systems that aid infection control, ensuring a high standard of care for patients.

In recent years, HMC has upgraded its ambulance service, introducing a new fleet, expanding the LifeFlight Service. These strategic steps have significantly enhanced the service's ability to reach patients promptly, regardless of their location.

The HMCAS comprises several specialized units, each serving a unique purpose:

- **Standard Ambulance Unit**: Ambulance for emergency transport, staffed by two ambulance paramedics.

- **Charlie Unit**: 4x4 response vehicle with one Critical Care Paramedic (CCP) and one ambulance paramedic, responsible for advanced medical interventions.

- **Delta Unit**: 4x4 response vehicle with one distribution supervisor, tasked with scene management and operational supervision during emergencies.

- **LifeFlight**: Helicopter service for rapid transit from emergency scenes to HMC hospitals, equipped with advanced life support medical equipment and staffed by two medical crew and two pilots.

Additionally, the HMC ambulance service includes specialized vehicles for major incidents:

- **Major Incident Response Vehicles**: Equipped with rapidly deployable containers for medical, decontamination, and logistics support during major incidents.

- **Medical Vehicle**: Contains resources for creating temporary field treatment areas, staffed by specially trained crews to treat and stabilize patients before transferring them to HMC's Emergency Departments.

- **Decontamination Vehicle**: Enables clinical teams to support other emergency response agencies in hazardous material incidents, aiding decontamination efforts for multiple casualties.

- **Logistics Vehicle**: Stationed at Hamad International Airport, ready for major aircraft or ground emergencies, providing support and surge capacity for emergency departments.

These vehicles and units are crucial elements in HMC's efforts to provide efficient and effective emergency medical services, ensuring timely and high-quality care for patients across Qatar. In this paper we focus on the dynamic redeployment of standard ambulance and Charlie units/vehicles.

1.4. Problems With the Current EMS System

This section identifies key shortcomings in the current EMS framework, including static dispatching methodologies that fail to adapt to dynamic environments and the complex challenges of ambulance redeployment.

1.4.1. Ambulance Dispatching

Existing ambulance dispatching techniques employ a static approach in a dynamic setting. Numerous current approaches adhere to strict guidelines; allocating ambulances exclusively based on the Euclidean distance connecting the site of the occurrence and the available ambulances [8]. This simplistic approach fails to account for the city's road network or traffic conditions, potentially leading to suboptimal dispatch decisions. Despite the strong correlation between Euclidean distance and road network distance, outliers exist, as illustrated in the scatter plot in Figure 1.3 [9].

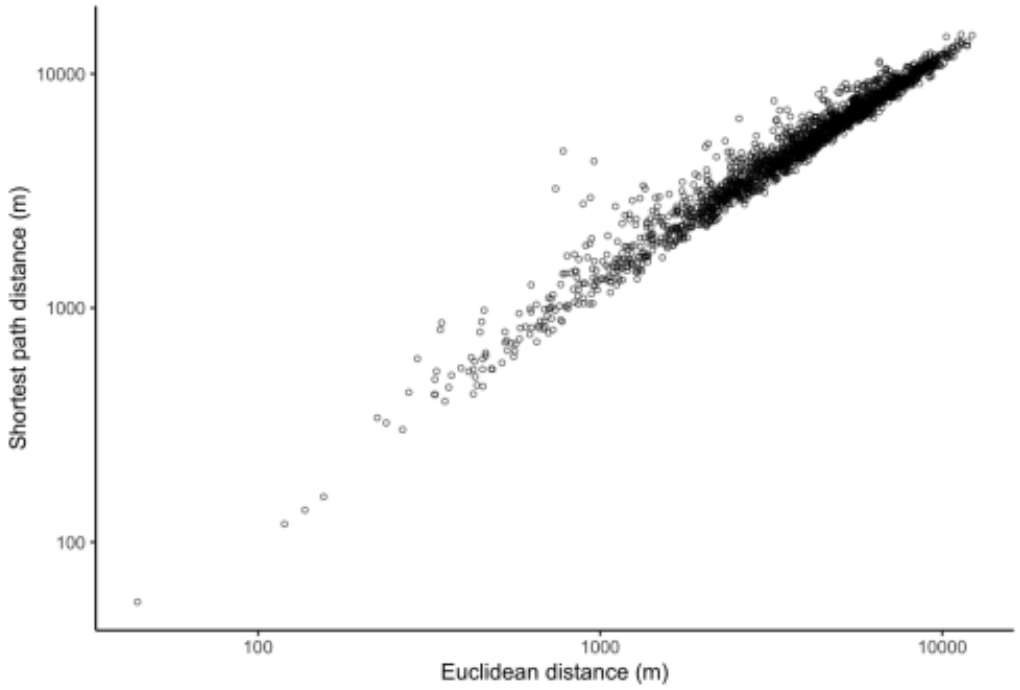


Figure 1.3. Euclidean distance versus road network distance in a scatter plot [9]

The distinction between road network distance and Euclidean distance is seen in Figure 1.4. It draws attention to situations in which the Euclidean distance is less than the real road distance, highlighting the drawbacks of using Euclidean distance alone to determine dispatching [10].

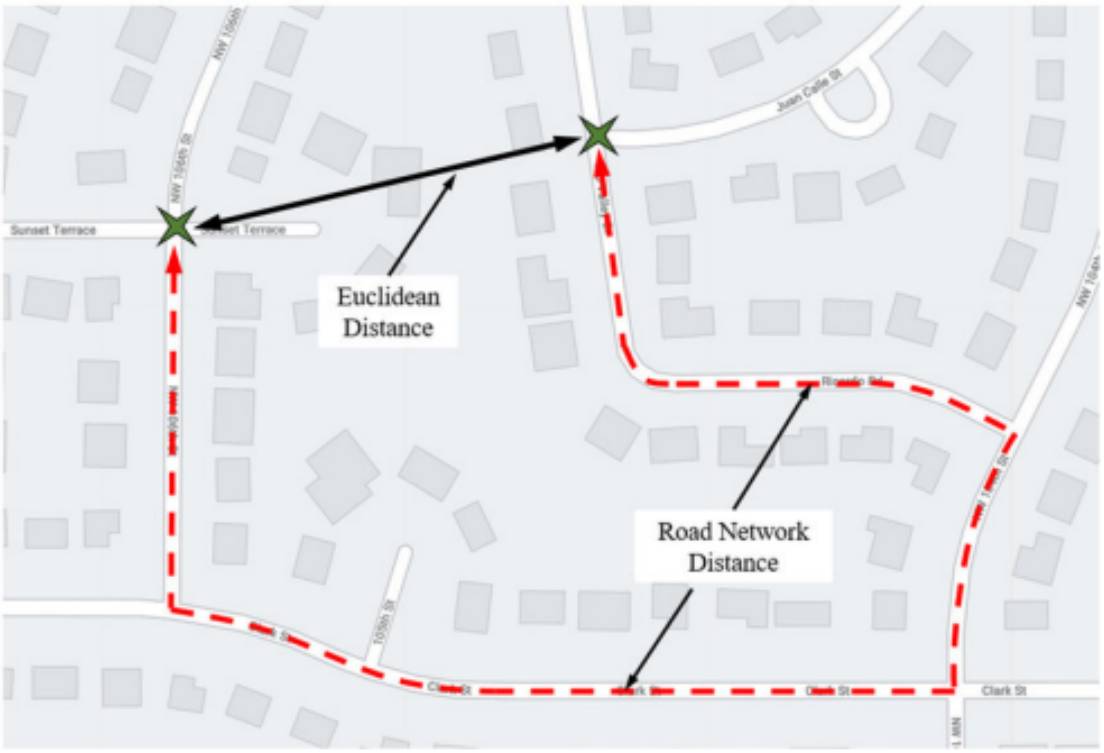


Figure 1.4. There is a difference between the road network distance and the Euclidean distance, as can be seen from the comparison. The Euclidean distance is obviously less than the road network distance [10]

Usually, incidents are handled first-in-first-out (FIFO) from the incident queue. But this strategy might not always be the best one, particularly when taking incident priority into account. Higher-priority incidents further down the list may need to be attended to immediately, which would cause lower-priority occurrences to receive shorter response times.

EMS regulations mandate the dispatch of an ambulance once an incident occurs, even if a closer ambulance is soon to become available. This practice can lead to sub-

optimal response times, as a closer but momentarily unavailable ambulance could have responded faster than the currently dispatched one [11].

Furthermore, ambulances may be lured away from places where they ought to have been placed strategically if the closest ambulance is dispatched without proper consideration to the scene of the occurrence. The frequent deployment of ambulances to one location of concern may cause delays in responding to other regions that may be high-risk [12].

1.4.2. Ambulance Redeployment Problem

Swift and effective ambulance redeployment are crucial in preserving lives during medical emergencies [2]. Precisely forecasting the strategic positioning of ambulances holds the potential to notably decrease response times, thereby guaranteeing prompt medical aid for individuals facing critical situations. Yet, the difficulty lies in identifying the most suitable spoke stations for ambulances, a multifaceted undertaking influenced by various factors such as population density, traffic flow, geographical characteristics, and historical incident data [13].

The ambulance redeployment problem stands as a crucial challenge within EMS optimization. Its primary objective is to ascertain the most effective placement of ambulances within a geographical area at specific spoke stations, aiming to minimize response times and ultimately enhance patient outcomes. This problem involves intricate spatial, temporal, and resource allocation considerations. It demands a delicate balance between factors such as coverage, demand dynamics, and operational constraints.

Tackling the ambulance redeployment problem holds immense significance in the realm of emergency response systems. By addressing this challenge, it becomes

possible to significantly enhance the efficiency of these systems, ensuring timely and effective medical assistance for individuals in need [14]. The optimization of ambulance redeployment not only saves crucial time in emergencies but also contributes significantly to improving overall healthcare outcomes within communities.

1.4.3. Other Challenges in the EMS system

In addition to the challenges discussed earlier, there are several other critical issues affecting the current EMS system:

- **Incident Over-labeling**: Because of the unpredictability of the call, incidents are frequently classified as acute, which causes an overlabeling of episodes in this category. This can result in inefficient resource allocation and potentially delayed response times for truly urgent incidents [15].
- **Lack of Consensus on Standards**: There is no consensus on standards for assessing the accuracy of EMS dispatching, making it difficult to compare different EMS systems and evaluate the effectiveness of new dispatching strategies [16].
- **Limited Resource Optimization**: The current EMS system often relies on static approaches to resource allocation, which may not be optimal for dynamic environments. More flexible and adaptive approaches are needed to optimize resource allocation [17].
- **Lack of Integration with Advanced Technologies**: Many EMS systems lag behind in adopting advanced technologies such as real-time data analytics and machine learning, which could significantly improve efficiency and patient care [18].

Addressing these issues is crucial for improving the overall efficiency and effectiveness of EMS systems, ultimately leading to better patient outcomes.

1.5. Methodology

The research plan was structured into several work packages (WPs) outlined as follows:

WP1: Conducting a literature survey on EMS, focusing on the Ambulance Redeployment Problem and the application of ML in EMS.

WP2: Developing a comprehensive mathematical model for the Ambulance Redeployment Problem, aimed at deriving a clear model to facilitate the formulation of an optimization framework.

WP3: Designing Deep Reinforcement Learning policy gradient algorithms for the dynamic redeployment of ambulances and Charlie vehicles.

WP4: Implementing the proposed redeployment method based on the developed algorithms.

WP5: Conducting data preprocessing and analysis on real-world data collected from HMCAS to prepare the dataset for modeling and evaluation.

WP6: Conducting comparative experiments and analysis between the proposed techniques and state-of-the-art baseline methods.

WP7: Writing up the thesis and preparing publications based on the research findings.

1.6. Thesis Objectives and Contributions

This thesis aims to address the challenges in ambulance and Charlie vehicle redeployment by proposing a novel approach that leverages deep learning and RL techniques. The primary objectives of this research are:

1. **Developing a Deep Score Network:** In order to combine all of a station's and regions dynamic factors into a single score, the thesis suggests using a DNN known as the deep scoring network. The efficient trade-offs between many parameters made possible by this network will result in more effective judgments on the redeployment of ambulances and Charlie vehicles.

2. **Applying Reinforcement Learning:** To learn the deep score network, the research employs a DRL framework that is based on a policy gradient method. With this method, the network may learn dynamically and flexibly, progressively increasing its performance.

3. **Evaluating the Proposed Method:** The thesis aims to evaluate the proposed dynamic ambulance redeployment method using real-world datasets. The evaluation will compare the performance of the proposed method against state-of-the-art baseline methods, demonstrating its effectiveness and efficiency.

4. **Testing Robustness:** In-depth testing will also be done as part of the thesis to see how well the proposed approach holds up under various scenarios, such as shifting traffic patterns and varying EMS system settings. This will guarantee that the approach can be used successfully in a variety of real-world scenarios.

The contributions of this thesis are:

1. **Novel Methodology:** The proposed approach combines deep learning and RL techniques in a novel way to address the complex problem of ambulance and Charlie redeployment. This methodology has the potential to significantly improve the efficiency and effectiveness of EMS.

2. **Practical Applications:** The research findings are expected to have practical applications in real-world EMS systems, leading to more timely and effective responses to emergency calls. This could ultimately lead to improved patient outcomes and reduced mortality rates.

3. **Contribution to the Literature:** The thesis contributes to the existing literature on ambulance redeployment by introducing a new methodology and demonstrating its effectiveness through rigorous evaluation and testing. This could serve as a valuable reference for future research in this area.

1.7. Thesis Outline

This thesis is structured into several chapters, each building upon the previous one to provide a comprehensive analysis and solution to the ambulance redeployment problem. The chapters are as follows:

Chapter 2 offers a thorough analysis of the literature on ambulance redeployment, including a range of topics including RL-based methods. It also discusses forecasting methods, RL, RL terminology, and multi-agent RL. This chapter sets the foundation for the research by summarizing the current state of the art and identifying gaps in existing approaches.

Chapter 3 focuses on the datasets used in the research and the analysis of the data. The chapter includes an in-depth analysis of the incident data, highlighting key metrics such as response time, incidents per location, and incidents over time. This research aids in comprehending the difficulties associated with redeployment and offers insightful information about the characteristics of ambulance requests. It also delves into the simulation aspects of the research, discussing both real-world EMS records and synthetic simulation. The chapter outlines the baseline simulation environment parameters, providing a detailed overview of the simulation setup used in the study.

Chapter 4 presents the dynamic redeployment method and the deep scoring network. It provides an explanation of the ambulance redeployment issue statement and shows how the deep score network combines all of a station's dynamic elements into a single score. The chapter also covers the dynamic redeployment technique and the RL framework that was used to train the deep-scoring network. The foundation for the research approach is laid forth in this chapter.

Chapter 5 presents the evaluation metrics used to assess the proposed redeployment method. It discusses the effectiveness and time efficiency of the method, as well as the convergence of training. The chapter also analyzes the necessity of considering all factors and the influence of patient amount. Additionally, it evaluates the robustness of the proposed redeployment method to various factors such as traffic conditions, the number of ambulances, and human factors.

Chapter 6 discusses the dynamic redeployment of Charlie Vehicles in EMS, which are specialized 4x4 response vehicles. Unlike standard ambulances, Charlie vehicles are deployed dynamically to regions with anticipated high demand for critical care. The chapter explores the challenges and opportunities in managing Charlie Vehicles, and shows how the deep score network combines all of a region's dynamic elements into a single score. The chapter also covers the dynamic redeployment technique and the RL framework that was used to train the deep-scoring network.

Chapter 7 presents the evaluation metrics used to assess the proposed redeployment of Charlie vehicle method. It discusses the effectiveness of the method, as well as the convergence of training. The chapter also analyzes the necessity of considering all factors and the influence of patient amount.

Chapter 8 concludes the thesis by summarizing the key findings and contributions of the research. It also suggests future directions for research, including further exploration of RL, simulation, data analysis, and other optimization techniques.

For improve the readability of the thesis, we give a summary of abbreviations in Table 1.1

Table 1.1. Abbreviations and Full Names

Abbreviation	Full Name
ADP	Approximate Dynamic Programming
AveRT	Average Response Time
CAD	Computer-Aided Dispatch
CCP	Critical Care Paramedic
DMEXCLP	Dynamic Maximum Expected Covering Location Problem
DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
EMT	Emergency Medical Technicians
EMS	Emergency Medical Services
EMXCLP	Maximum Expected Covering Location Problem
ERTM	Expected Response Time Model
FIFO	First-In-First-Out
GA	Genetic Algorithm
HMC	Hamad Medical Corporation
HMCAS	Hamad Medical Corporation Ambulance Services
LSTM	Long short-term memory
LS	Least Ambulances
MCLP	Maximal Covering Location Problem
MAPE	Mean Absolute Percentage Error
MAQR	Multi-agent Q-network with Experience Replay

Continued on next page

Abbreviation	Full Name
MDP	Markov Decision Process
MEXCLP	Maximum Expected Covering Location Problem
MIP	Mixed Integer Programming
ML	Machine Learning
MLP	Multi-layer perceptron
MSE	Mean Squared Error
NN	Neural Network
NS	Nearest Station
OF	Objective Function
PHC	Primary Healthcare Corporation
PT	Pre-Dispatch Time
PU	Pick Up
RA	Random Allocation
RelaRT	Relative Response Time
RF	Random Forest
RL	Reinforcement Learning
RPMP	Reliability P-Median Problem
SMDP	Semi Markov Decision Process
SimPy	Simulation in Python
WGAN	Wasserstein Generative Adversarial Neural Network

CHAPTER 2: LITERATURE REVIEW

The primary areas of EMS research can be categorized into Ambulance dispatching (Section 2.1), EMS forecasting (Section 2.2) and Ambulance Redeployment (Section 2.3) as outlined in Figure 2.1.

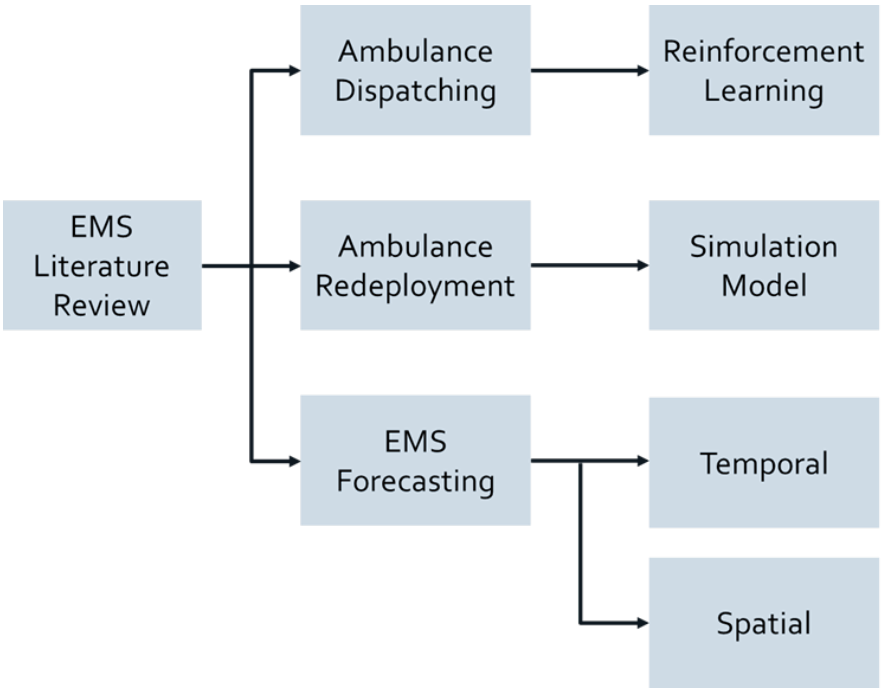


Figure 2.1. A brief overview of the studies covered in this chapter

Current reviews of the ambulance dispatching issue and EMS research are given in detail by Bélanger et al. [19], Neira et al. [20], and Mukhopadhyay et al. [21]. This chapter also discusses the background for RL.

Online and batch learning are distinguished in machine learning. Training a model on a given dataset in batches and then deploying the trained model is known as batch learning. After being deployed, the model is not retrained or adjusted to consider fresh data that was gathered during deployment. Online learning, on the other hand, uses the model's previous predictions to continuously train it during deployment.

2.1. Ambulance Dispatching

Complex decision-making is involved in dispatching ambulances, and the literature has examined a number of algorithmic and Reinforcement Learning (RL) techniques. Considering the large search space of all the scenarios of assigning m ambulances to n zones, the problem presents challenges for algorithmic methods like linear programming [22], mixed integer programming [23], and tabu-search [24].

However, RL provides advantages when it comes to managing dynamic and stochastic situations and adjusting to changing conditions. To optimize response time and coverage in ambulance dispatching scenarios, RL approaches generally use a Markov Decision Process (MDP) model, including versions like Q-learning and Approximate Dynamic Programming (ADP).

In one study, Bandara et al. examined how incident priority can improve patient survivability in EMS dispatching. Equation 2.1 illustrates how they used a survival function $S(t_R)$ to predict the chance of survival.

$$S(t_R) = \max\left[(0.594 - 0.055 \times t_R)\,; 0\right] \tag{2.1}$$

, where t_R denotes the response time [25]

The study looked at 2x2 zones and used commercial optimization software and thorough enumeration to determine the optimum course of action. The study made the assumptions that response times are independent of priority, that ambulances only dispatch from base stations, and that there is a constant arrival rate that follows a Poisson distribution.

The optimum course of action is mostly dependent on balancing demand among

zones, according to the results. It is always best to send the closest ambulance when demand is balanced. Nonetheless, in cases of imbalance, the nearest ambulance is sent to acute occurrences, though not always to urgent ones. An ambulance's availability for urgent incidents and the shortest distance are balanced in the ideal policy. The closest ambulance in a zone with less demand responds to urgent occurrences from any zone, which lengthens the average response time but improves patient survival.

For simple scenarios, this optimal policy makes sense, but for realistic circumstances with more granularity, it becomes more difficult and computationally demanding. A heuristic was created to simulate the best course of action in scenarios with greater granularity in order to address this. This approach, however, is deterministic and ignores the fact that Poisson rates vary between zones.

In order to allocate military medical evacuation helicopters, Keneally et al. used MDP while taking Euclidean distance and incident priority into account [26]. Their research demonstrated the advantages of overreacting when there are significant classification errors, focusing on the effects of patient priority classification errors on dispatching policies.

For effective ambulance dispatching, Liu et al. created the Multi-agent Q-network with Experience Replay (MAQR). Their RL-based approach beat heuristic algorithms like location-based and time-based allocation, and each ambulance was represented as an agent [27].

For responder dispatch, Mukhopadhyay et al. combined incident prediction and dispatch models with a parametric survival model and a Semi Markov Decision Process (SMDP) [28]. Their strategy, which included ambulance response time models and online incident prediction, demonstrated encouraging reductions in response times.

Together, these studies highlight the potential for optimizing ambulance dispatching using algorithmic and RL-based approaches, with RL methods providing flexibility and adaptation in dynamic contexts.

2.2. EMS Forecasting

Forecasting involves predicting both spatial and temporal aspects, which becomes complex, particularly with high granularity. In the EMS context, high temporal granularity refers to intervals of one hour.

Using the same dataset, Hermansen [29] and Van De Weijer and Owren [30] concentrated on prediction in their earlier master's theses.

Hermansen investigated split and complete strategies. Whereas the split approach models the overall volume and spatial distribution separately, the whole approach predicts each place directly. Using online learning as opposed to batch learning, they experimented with Multi-layer Perceptron (MLP) and Long Short-Term Memory (LSTM) models. In the individual assessments of volume and spatial distribution, the split technique outperformed it; nevertheless, in the combined evaluation, the whole strategy had a marginal advantage.

Call volume was found to be influenced by weather [31], with models that did not include weather data exhibiting superior performance in volume evaluation. The best model for combined evaluation, however, included weather data, suggesting that time-dependent aspects of models that did not explicitly include weather data might reflect weather variability.

Using a split strategy similar to Hermansen's, Van De Weijer and Owren used MLP in conjunction with a variety of time series decomposition approaches to estimate

total incident volume over time. They used a genetic algorithm (GA) to forecast volume by optimizing the weights of a Poisson neural network.

For spatial prediction, they used GA for spatial location aggregation and a pre-trained Wasserstein Generative Adversarial Neural Network (WGAN). While the volume forecast was successful, the spatial prediction models faced challenges, with the GA-aggregated MLP slightly outperforming the historical average. Adding covariates and spatial aggregation improved spatial predictions but decreased precision. The study also revealed a consistent number of people requiring ambulances per location over time.

Mannering [32] concentrated on highway accidents, noting the temporal instability of model parameters due to driver behavior. Driver behavior, influenced by factors like age, risk-taking tendencies, and macroeconomic conditions, is temporally unstable, necessitating online learning for optimal parameter adaptation over time. Urban dynamics, such as traffic and population shifts, underscore the importance of online learning in adapting to evolving environments.

2.3. Ambulance Redeployment

Schjolberg and Bekkevold [33] proposed using GA and other evolutionary methods for ambulance redeployment optimization, departing from the traditional approach of Mixed Integer Programming (MIP) due to its higher time complexity. Their study focused on both daytime and nighttime redeployment strategies, which were static and independent of the time dimension. Using a Discrete Event Simulation model in Java, built upon the work of McCormack and Coates [34], they evaluated the effectiveness of these redeployment strategies.

Clustering algorithms were used to accomplish population-proportionate rede-

ployment, which allowed for customization based on the population surrounding base stations. The strategy fared better than others in a one-year lengthy simulation, according to the results. The authors questioned if population-proportionate redeployment is always the best option or if this superiority was the result of overfitting by the GA approach. The optimized redeployments performed better in the short term (less than three months), suggesting that although improved ambulance redeployment works well at first, it needs to be updated on a regular basis to continue working well.

The effects of changing the quantity of ambulances used for optimization and simulation were also examined in the study. Results indicated that fewer ambulances might be dispatched to Akershus and Oslo without appreciably raising the average response time. The scientists did, however, issue a warning, stating that consideration should be given to the non-linear relationship between reaction time and survivability (the likelihood of patient survival). They contended that if Oslo University Hospital (OUH) offered a survival function, redeployments may be further streamlined.

Survival functions are crucial for modeling the probability of patient survival based on response time, particularly in critical incidents like cardiac arrest. These functions, as described in the literature, are influenced by factors such as incident type, patient age and health, and the availability of medical care in ambulances. Examples of such survival functions are illustrated in Figure 2.2.

A recent study addresses the challenge of dynamic redeployment of ambulances to minimize the expected fraction of late arrivals [36]. In dynamic ambulance repositioning, decisions on how to redeploy vehicles need to be made in real time, considering the status of all other vehicles and ongoing accidents. This problem is particularly challenging in urban areas, and traditional solution methods become intractable as the

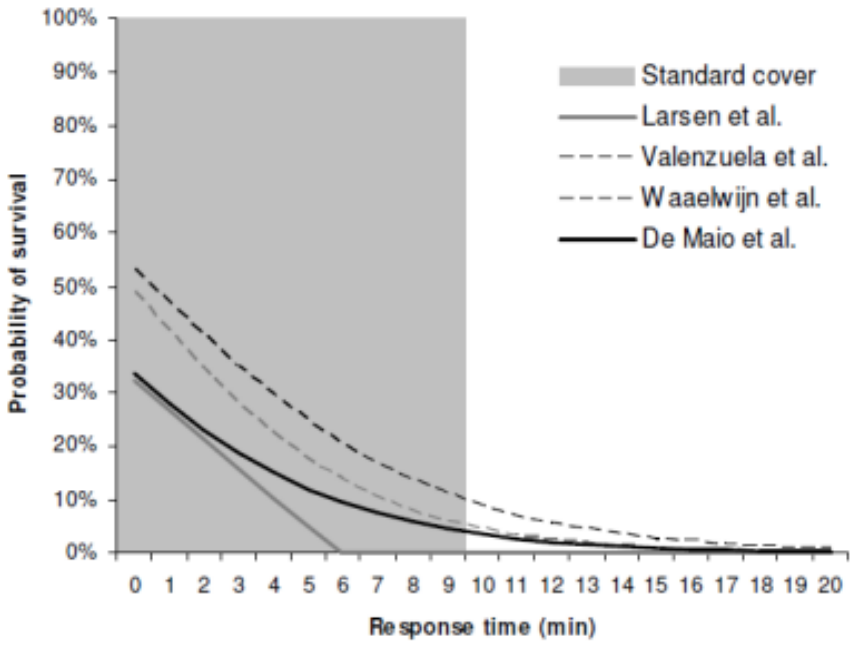


Figure 2.2. A comparison of cardiac arrest survival functions [35]

number of vehicles increases. Therefore, the study aimed to develop a scalable algorithm that performs well in practice.

The proposed approach is a polynomial-time heuristic, a method that makes quick, practical decisions without requiring extensive assumptions on the region or detailed state information. Unlike other solution methods that become impractical with a large number of vehicles, the heuristic is designed to be scalable and efficient for real-time decision-making.

To evaluate the performance of the heuristic, the study uses a simulation model of EMS operations. The performance of the proposed heuristic is compared to static solutions, including a classical scenario where an idle vehicle is always sent to its predefined base location. The results show that the heuristic outperforms the optimal static solution for a tractable problem instance. Additionally, a realistic urban case study demonstrates a 16.8% relative improvement in performance compared to a benchmark static solution.

The study concludes that the proposed algorithm fulfills the need for real-time, simple redeployment policies that significantly outperform static policies in terms of minimizing late arrivals of ambulances. The use of a polynomial-time heuristic allows for efficient decision-making without sacrificing performance, making it a practical solution for dynamic ambulance repositioning in urban areas.

The Maximum Coverage Location Problem (MCLP), introduced by Church and Revelle in 1974 [37], aims to maximize the weighted coverage of demand locations by at least one ambulance. The model uses binary variables to represent whether an ambulance is placed at a specific base location and whether a demand location is covered by at least one ambulance. The binary variable x_j indicates whether an ambulance is placed at base location j, with values of 0 (no ambulance) or 1 (ambulance present). Similarly, the binary variable y_{i1} represents whether demand location i is covered by at least one ambulance, also taking values of 0 (not covered) or 1 (covered).

The objective function of the MCLP is to maximize the sum of the weighted coverage of demand locations by at least one ambulance, as shown in Equation (2.2):

$$\max \sum_{i \in I} d_i y_{i1} \tag{2.2}$$

This objective function calculates the total weighted coverage achieved by the ambulance deployment, where d_i represents the weight assigned to demand location i, indicating its importance.

Constraints (2.3) ensure that if demand location i is covered by at least one ambulance ($y_{i1} = 1$), then there is at least one ambulance located at a base location j that covers i:

$$\sum_{j \in J_i} x_j \geq y_{i1} \quad \forall i \in I \tag{2.3}$$

where J_i denotes the set of base locations that cover demand location i.

Equation (2.4) limits the total number of ambulances that can be placed at the base locations to p, ensuring a constraint on the total number of ambulances deployed:

$$\sum_{j \in J} x_j = p \tag{2.4}$$

Constraints (2.5) and (2.6) specify that the variables x_j and y_{i1} can only take binary values (0 or 1) indicating whether an ambulance is placed at base location j and whether demand location i is covered by at least one ambulance, respectively:

$$x_j \in \{0, 1\}, \forall j \in J \tag{2.5}$$

$$y_{i1} \in \{0, 1\}, \forall i \in I \tag{2.6}$$

The MCLP can be used to determine optimal base locations and the number of bases needed for coverage at varying levels. However, it assumes that ambulances are always available, which may not be the case in practice. This assumption can impact the model's practical feasibility, as the coverage indicated by the model may not be guaranteed in real-world scenarios.

Random Allocation (RA) is a basic method for ambulance redeployment where ambulances are randomly assigned to available stations without considering any specific criteria or optimization strategies. This approach relies solely on chance for station

selection, making it a naive redeployment method.

The Nearest Station (NS) method is a straightforward approach to ambulance redeployment, where ambulances are reassigned to the nearest station when they become available. This method prioritizes minimizing travel time to incidents by placing ambulances in close proximity to their likely destinations.

On the other hand, the Least Ambulances (LS) method focuses on redistributing available ambulances to stations with the fewest number of ambulances currently deployed. The goal of this approach is to achieve a more even distribution of ambulances, ensuring better coverage and response capabilities across different geographical areas.

By deploying ambulances to stations with fewer resources, the LS method aims to address potential imbalances in ambulance availability. This can help improve response times in areas that are currently underserved or have higher demand for emergency medical services.

Similar to the Reliability P-median Problem (RPMP) for facility location difficulties proposed by Snyder and Daskin (2005), the Expected Response Time Model (ERTM) seeks to minimize the expected response time for all demand locations [38]. Even in the event that every ambulance is full, the ERTM nevertheless presumes that every emergency call is handled.

For each demand location di, the ERTM uses binary variables z_{dijk} to find the nearest ambulance, the second nearest ambulance, and so on. When an ambulance at base j is the kth-nearest ambulance for demand location di, these variables are set to 1. It is possible to compute the predicted response time for each demand location using these data.

The probability that demand location di is served by the nearest ambulance is

$1 - q$, where q is the probability that di is served by the second nearest ambulance, and so on. The probability that di is served by the farthest or pth-nearest ambulance is calculated differently to ensure that the probabilities sum up to one.

The ERTM then minimizes the weighted expected response time using the following objective function:

$$\min \sum_{j \in J} \sum_{di \in DI} \sum_{k=1}^{p-1} d_{di} t_{ji} (1-q) q^{k-1} z_{dijk} + \sum_{j \in J} \sum_{di \in I} d_{di} t_{jdi} q^{p-1} z_{dijp} \tag{2.7}$$

subject to the following constraints:

$$\sum_{j \in J} z_{dijk} = 1, \quad \forall di \in DI, k \in \{1, \ldots, p\} \tag{2.8}$$

$$x_j \geq \sum_{k=1}^{p} z_{dijk} \quad \forall di \in DI, \quad \forall j \in J \tag{2.9}$$

$$\sum_{j \in J} x_j \leq p \tag{2.10}$$

$$x_j \in \mathbb{N} \quad \forall j \in J \tag{2.11}$$

$$z_{dijk} \in \{0, 1\} \quad \forall di \in DI, \quad \forall j \in J, k \in \{1, \ldots, p\} \tag{2.12}$$

In these equations (Equations 2.7,2.8,2.9,2.10,2.11,2.12), $d_{di} t_{dij}$ represents the travel time from base j to demand location di, q_k is the probability that di is served by

the kth-nearest ambulance, and x_j is a binary variable indicating whether base j is open.

Among the earliest models to consider the busy fraction q of ambulances is Daskin's (1983) MEXCLP [39]. The model takes into account the likelihood that an ambulance will be available within the goal response time r, and maximizes the weighted predicted coverage of all demand locations. Whether at least k ambulances can cover demand location $i \in I$ is indicated by the binary variable y_{ik}. The likelihood that one of the ambulances is available is calculated in the objective function. The overall weighted projected coverage is shown here.

$$\max \sum_{i \in I} \sum_{k=1}^{p} d_i (1-q)^{k-1} y_{ik} \tag{2.13}$$

subject to,

$$\sum_{j \in J_i} x_j \geq \sum_{k=1}^{p} y_{ik}, \forall i \in I \tag{2.14}$$

$$\sum_{j \in J} x_j \leq p, \tag{2.15}$$

$$x_j \in \mathbb{N}, \quad \forall j \in J \tag{2.16}$$

$$y_{ik} \in \{0,1\}, \quad \forall i \in I, k \in \{1,\ldots,p\} \tag{2.17}$$

Jagtenberg introduced an algorithm [40], in addition to the static MEXCLP, that addresses the dynamic ambulance relocation problem. When an ambulance becomes idle after serving an accident, the DMEXCLP algorithm is activated to determine its next base location. For each potential base location, the algorithm calculates the improvement in coverage that would result from dispatching an ambulance there. The destination chosen is the one that maximizes total coverage. Unlike simply sending an ambulance

to a base location with the fewest ambulances, which may not always lead to better coverage, the DMEXCLP algorithm considers this factor along with others, making it highly effective for dynamic ambulance relocation scenarios. For the specific details of the DMEXCLP algorithm, refer to Appendix B, Algorithm 4.

2.4. Reinforcement Learning

By simulating intelligent agents that interact with their surroundings in order to maximize a cumulative reward, RL is a machine learning paradigm that sets itself apart from both supervised and unsupervised learning [41].

In RL, an environment is defined by its states, possible actions, and rewards. When the environment is in a state s, taking an action a leads to a transition to a new state s' and an associated reward r. The objective is to select actions over time t to maximize the total reward for an episode, which is a sequence of states and actions until a terminal state is reached or a maximum number of iterations is reached (Figure. 2.3). In non-deterministic environments, the transition from state s to s' given action a occurs with a probability $P(s'|s, a)$.

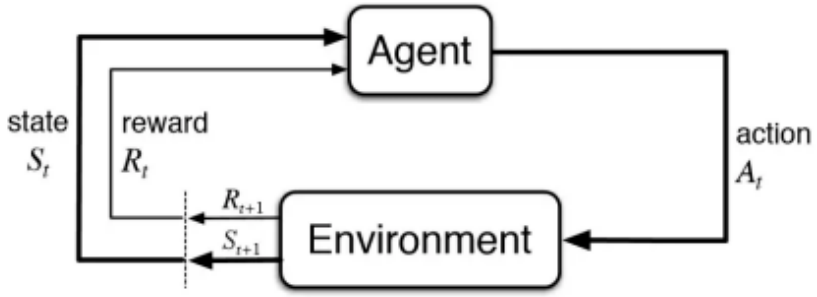


Figure 2.3. The setup of reinforcement learning [42]

The Markov property, which asserts that future states solely depend on the current state and not on previous states, is assumed by most RL theories to apply to

environments. A policy π, which can be based on a state-value function $V(s)$ that represents the desirability of being in a state s (following policy π), serves as the basis for actions. This policy is learned over time. Starting from that state, the state-value function determines the predicted future rewards, which are then discounted by a factor γ across subsequent time steps t. Equation 2.18 defines the state-value function for an infinite time horizon.

$$V^{\pi}(s_{t=0}) = E\left[\sum_{t=0}^{\infty} \gamma^t \cdot r_t\right] \tag{2.18}$$

Alternatively, a policy can be represented by an action-value function, often denoted as Q-values in Q-learning, which estimates the expected future rewards when action a is taken in state s, as in Equation 2.19.

$$Q^{\pi}(s, a) = E\left[\sum_{t=0}^{\infty} \gamma^t \cdot r_t | s, a\right] \tag{2.19}$$

In environments with small state spaces, policies can be represented as dictionaries mapping states and actions to values. For larger state spaces, policies can be implemented as Deep Neural Network (DNN), taking states as inputs and outputting expected future reward distributions (Q-values) among possible actions.

2.4.1. Deep Reinforcement Learning

In DRL, the core idea is to leverage DNN to approximate complex functions that are involved in the RL process. This is particularly useful for handling high-dimensional state spaces, which are prevalent in real-world applications.

One of the key components in RL is the Q-function (or action-value function), which determines the expected return for taking an action in a given state and following

a specific policy thereafter. In DRL, the Q-function is represented by a DNN. The network takes the state as input and outputs the Q-values for each possible action.

Mathematically, the Q-function can be represented as:

$$Q(s, a; \theta) \tag{2.20}$$

where s represents the state of the environment, describing its current situation or configuration as perceived by the agent. a denotes the action taken by the agent in response to the state s, representing the decision made to influence the environment. θ represents the parameters of the neural network used to approximate the policy or value function in reinforcement learning.

The goal of training the Q-network is to minimize the following loss function:

$$L(\theta) = \mathbb{E}\left[(Q(s, a; \theta) - (r + \gamma \max_{a'} Q(s', a'; \theta^{-}))^2\right] \tag{2.21}$$

where: r represents the reward received after taking action a in state s, indicating the immediate feedback from the environment. s' denotes the next state, representing the state that the agent transitions to after taking action a in state s. γ is the discount factor that determines the importance of future rewards, with values closer to 1 indicating a greater emphasis on long-term rewards. θ^{-} represents the parameters of a target network used to stabilize training by providing a more stable estimation of the value function or policy.

Similarly, in policy-based methods, the policy function is represented by a DNN. The network takes the state as input and outputs a probability distribution over actions. The policy is trained using techniques such as stochastic gradient ascent to maximize

the expected cumulative reward.

2.4.2. Policy Gradient

In RL, policy gradient methods are a class of algorithms that optimize the policy function directly without requiring the value function to be estimated explicitly. Policy gradient approaches update the policy parameters in a way that maximizes the expected return, as opposed to changing the Q-values.

The gradient of the expected return with regard to the policy parameters is used to compute the policy gradient. The policy settings are updated using this gradient in a way that maximizes the probability of choosing actions that result in larger rewards. DNN are frequently combined with policy gradient techniques to learn intricate policies in high-dimensional domains.

The policy gradient is a way to update the policy parameters θ in the direction that increases the expected return $J(\theta)$. It is given by the gradient of the expected return with respect to the policy parameters:

$$\nabla_\theta J(\theta) = \mathbb{E}_\tau[\nabla_\theta \log \pi_\theta(\tau) R(\tau)] \tag{2.22}$$

The gradient of the expected return with respect to the policy parameters θ, denoted as $\nabla_\theta J(\theta)$, signifies the direction in which the policy parameters should be adjusted to improve the expected return.

A trajectory, represented by τ, is a sequence of states and actions sampled from the policy. It captures the path taken by an agent in the environment.

The probability of a trajectory τ under a policy π with parameters θ, denoted as $\pi_\theta(\tau)$, indicates the likelihood of observing the trajectory given the policy.

The return of a trajectory τ, denoted as $R(\tau)$, is the sum of rewards obtained by the agent along that trajectory. It is a measure of the cumulative reward achieved by following a specific policy.

The policy parameters are updated in the direction that increases the expected return by taking a step proportional to the policy gradient:

$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta J(\theta) \tag{2.23}$$

The policy parameters at time step t, denoted as θ_t, represent the set of parameters that define the policy at a specific point in time. These parameters are updated iteratively to improve the policy.

The learning rate, denoted as α, is a hyperparameter that controls the size of the update to the policy parameters. It determines how quickly or slowly the policy adapts to new information.

By iteratively updating the policy parameters using the policy gradient, the policy converges towards an optimal policy that maximizes the expected return. DNN are often used to represent the policy function in high-dimensional state spaces, allowing policy gradient methods to learn complex policies.

2.4.3. Multi-agent Reinforcement Learning

Multiple spoke stations are integral to the ambulance redeployment problem, presenting a cooperative multi-agent RL scenario. Here, each agent strives to maximize its reward through collaboration. In contrast, other multi-agent RL scenarios may feature agents in competitive settings.

In competitive scenarios, agents' rewards conflict with each other, leading to

the development of complex policies reminiscent of Game Theory. Conversely, in cooperative setups, rewards are shared among agents, fostering coordination to maximize the system-wide reward.

The collaborative nature of EMS planning offers various planning perspectives. One approach involves formulating a policy for each spoke station, allowing them to act independently based on local observations—referred to as decentralized planning. Alternatively, centralized planning considers all spoke stations and current incidents, with a single policy making decisions based on system-wide observations to redeploy an available ambulance. This thesis focuses on implementing such a centralized policy.

CHAPTER 3: DATASET AND SIMULATION

3.1. Dataset

We evaluate the performance of our dynamic ambulance redeployment strategy using real-world data. The data from Qatar's EMS system, comprises road networks, hospitals, ambulance spoke stations, and records of EMS requests.

EMS Patient Request Records: These records show when patients place 999 calls in Qatar, which are comparable to 911 calls in the United States. Every patient record has a timestamp, latitude, and longitude that identify the patient's location. From January 1 to February 21, 2022, 51 days' worth of EMS request record data were gathered, totaling 45,619 records. An estimated 1205 EMS requests are received per day on average, or around 55 requests every hour.

Ambulance Hubs: Qatar has 7 ambulance hubs with specified geographical locations, including latitudes and longitudes.

Ambulance Spoke Stations: Qatar has 60 ambulance spoke stations, each with geographical coordinates, including latitudes and longitudes.

Hospitals: Patients are transported to 52 hospitals in Qatar by ambulance. The geographic locations of these hospitals are also available to us.

Road Networks: The road network data contains details on Qatari roadways, such as road vertices that are latitude and longitude coordinated.

3.1.1. Data Analysis

This section offers an analysis of the EMS request records dataset, organized into sections that examine various aspects of the dataset.

3.1.1.1. Response Time

The response time is a critical metric in EMS as it directly impacts patient outcomes. In this section, we analyze how response time is affected by the time of day.

The graph in Figure 3.1 shows the average response times throughout the day. By analyzing this graph, we can identify patterns and trends in response times based on the time of day. We can observe longer response times during peak traffic hours and shorter response times during off-peak hours.

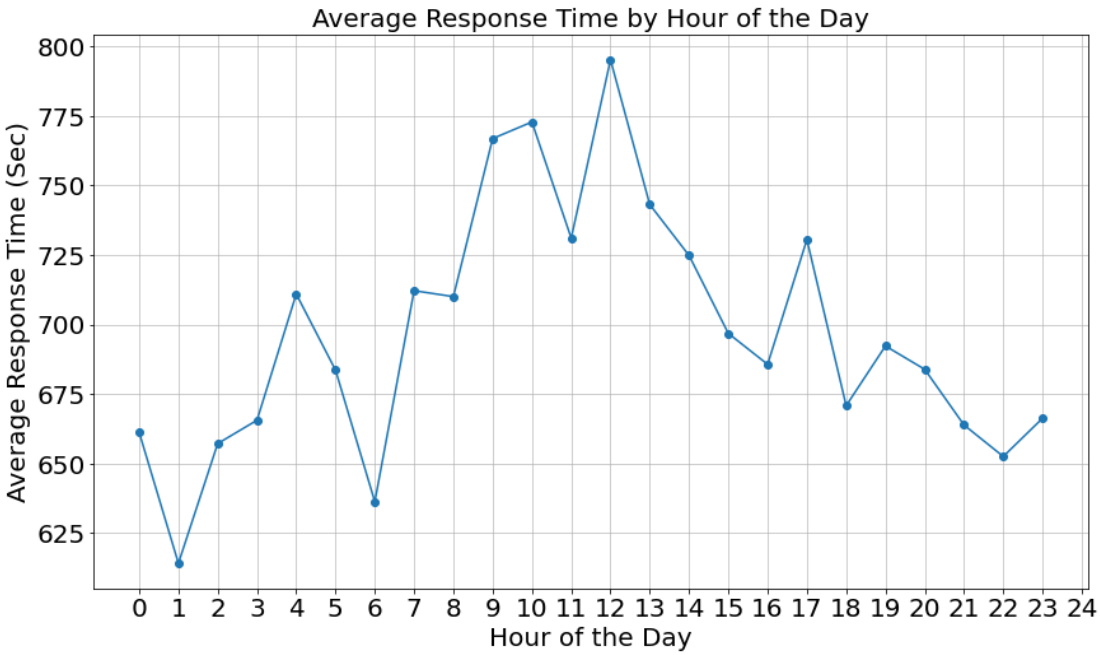


Figure 3.1. Average Response Time Vs Time of Day

Table 3.1 provide valuable insights into the distribution and characteristics of ambulance response times in the dataset. The average response time is approximately 699.97 seconds, indicating the typical duration for an ambulance to reach a call location. The standard deviation of 675.35 suggests a considerable variability around the mean, highlighting the range of response times experienced. The minimum response time recorded is 22.80 seconds, indicating the fastest response observed, while the maximum

response time is 7198.80 seconds, representing the longest response time in the dataset. The quartile analysis reveals that 25% of response times are less than or equal to 372.00 seconds (25th percentile), 50% are less than or equal to 511.80 seconds (50th percentile or median), and 75% are less than or equal to 750.00 seconds (75th percentile), showcasing the distribution of response times across different percentiles.

Statistic	**Value**
Mean	699.97
Standard Deviation	675.35
Minimum	22.80
25th Percentile	372.00
50th Percentile (Median)	511.80
75th Percentile	750.00
Maximum	7198.80

Table 3.1. Summary Statistics of Response Time

3.1.1.2. Patient Requests Per Location

Table 3.2 below lists the top 10 locations in Qatar, with the highest number of incidents recorded. These locations are identified by their latitude and longitude coordinates, and the number of incidents recorded at each location is also provided for a random day.

The location with coordinates (25.267457, 51.609393) recorded the highest number of incidents, with a total of 487 incidents. This suggests that this particular area in Doha experiences a higher frequency of incidents compared to other locations.

Location_Lat	Location_Long	Incident_Count
25.267457	51.609393	487
25.259036	51.614914	259
25.169158	51.399248	165
25.275943	51.528389	143
25.169889	51.398485	118
25.216073	51.530133	79
25.306712	51.499702	76
25.261972	51.613531	69
25.254509	51.534917	37
25.263201	51.543926	35

Table 3.2. Top 10 Locations with the Highest Number of Incidents

Locations such as (25.259036, 51.614914) and (25.169158, 51.399248) also reported relatively high numbers of incidents, with 259 and 165 incidents respectively.

The concentration of incidents in these specific areas could be attributed to various factors, including population density, traffic patterns, and the presence of commercial or residential areas. Understanding these spatial patterns can help emergency services allocate resources more effectively and implement targeted interventions to reduce the incidence of emergencies in these areas.

3.1.1.3. Patient Requests Over Time

Figure 3.2 shows the number of incidents per hour of a random day. The plot illustrates the variation in incident frequency throughout the day, providing insights into

the temporal patterns of incidents.

The highest number of incidents occurred as shown in Figure 3.2 during the evening hours, particularly at 5 PM, 6 PM, 7 PM, and 8 PM, with over 1300 incidents recorded during these hours. This observation suggests a potential correlation between time of day and incident occurrence.

With 902 events reported overall, the first day of February in 2022 was the date with the most incidents. Numerous reasons, including events, the surrounding environment, or other external factors impacting incidence rates, could be responsible for this large number of incidents. For emergency services, operational planning and resource allocation can be greatly impacted by an understanding of these trends.

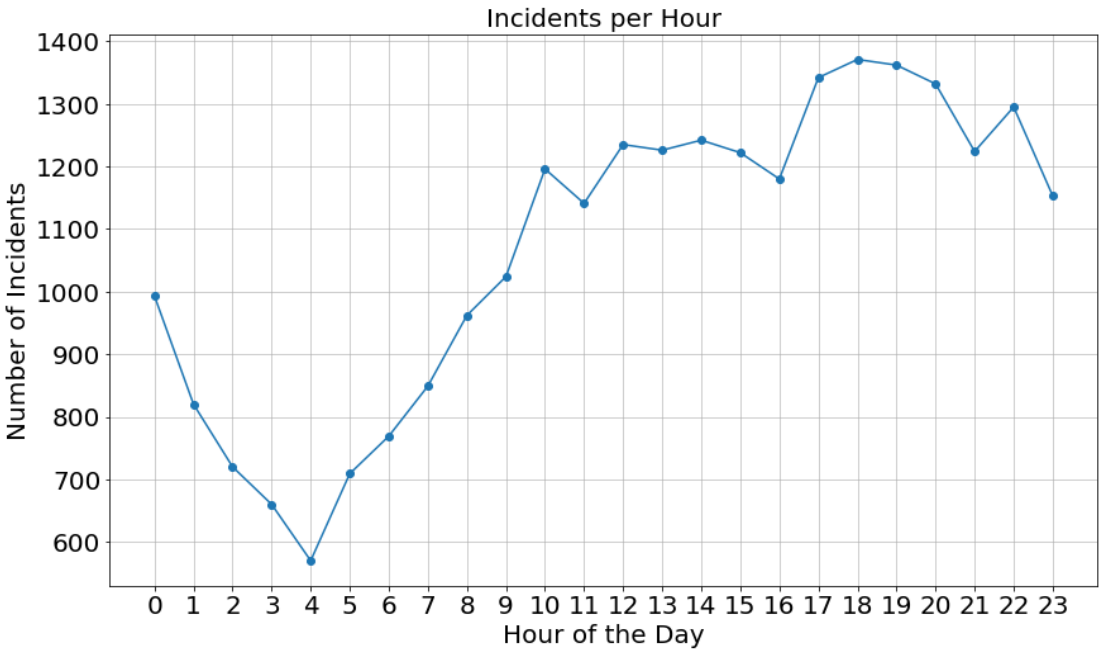


Figure 3.2. Incidents Per Hour

3.2. Simulation

Simulation plays a vital role in evaluating our proposed ambulance and Charlie vehicle redeployment method, enabling us to assess its performance under various

scenarios. By providing a controlled environment, simulation allows us to test and optimize ambulance redeployment strategies, offering insights that can guide real-world implementation. We begin by generating patient requests based on real-world EMS records, which are then used to train and test our deep score network and RL algorithms. Additionally, we utilize simulation with synthetic incidents through the OpenAI Gym and SimPy libraries to further evaluate our method's performance in a controlled setting. This section provides a detailed overview of our simulation setup.

We assess our ambulance and Charlie vehicle redeployment method's efficacy using simulations, which is a typical practice in the area [43]–[46]. In addition to ambulance and Charlie vehicle redeployment, simulation is frequently employed in various real-world scenarios involving decision-making, including express services [47], taxi sharing [48], [49], and more. Our simulation's patient requests are all based on actual EMS request records from Qatar, complete with timestamps and locations.

3.2.1. Synthetic Simulation

To facilitate the development and evaluation of the RL framework, we utilize "Gym", the OpenAI Gym, a standardized environment structure and Python library designed for the development and testing of RL algorithms. Additionally, our model incorporates "SimPy", a Python Discrete Event Simulation library, to manage the simulation processes efficiently. This comprehensive framework allows us to study and optimize ambulance and Charlie vehicle redeployment strategies in a controlled, simulated environment, providing valuable insights for real-world implementation.

The controlled behavior of the items within the simulation environment is made possible by its reduction of the real-world issue. It is explained in more detail in this

section.

3.2.1.1. Model Overview

- Events take place in certain regions of a finite-dimensional world. Throughout the day, the occurrences' geographic pattern could shift.
- Ambulances are deployed from designated spoke station in the event of an incident; the nearest free ambulance is used. On the other hand, Charlie vehicles are deployed from designated regions in the event of a critical incident; the nearest Charlie vehicle is used.
- A patient is picked up by an ambulance and taken to the nearest hospital.
- The agent (hub) then uses the suggested redeployment method to move the ambulance to a spoke station. After arriving at that spoke station, the ambulance is ready to be utilized for future emergencies.
- In the case of Charlie vehicle redeployment, the agent (hub) uses the suggested redeployment method to move the Charlie vehicle to a region. After arriving at that region, the Charlie vehicle is ready to be utilized for future emergencies.

3.2.1.2. Simulation Environment Initiation

For the duration of an agent's training and testing, the simulation environment is only launched once. The following are set up when the simulation environment object starts up:

- Maximum x and y coordinates in the world coordinate system have been reached.

– Incidents occur within predefined areas in a fixed-dimensional world, and their geographic distribution fluctuates during the day.

3.2.1.3. Baseline Simulation Environment Parameters

The following features were included in the configuration of the simulation environment:

– Size of the world is 40 km^2.

– One hospital is located at the centre of the world.

– Ambulances and Charlie vehicles, on average, each respond to eight incidents per day each (a low utilization is used so that call-to-response time is mostly dependent on placement of ambulances/Charlie vehicles, rather than any queuing).

– Ambulances must arrive at a spoke station before being available for incidents. On the other hand, Charlie vehicles must arrive at a region before being available for critical incidents.

– Ambulances and Charlie vehicles travel in straight lines at 80 kph.

– There are 10 spokes stations and regions spaced evenly across the 40 km^2 world.

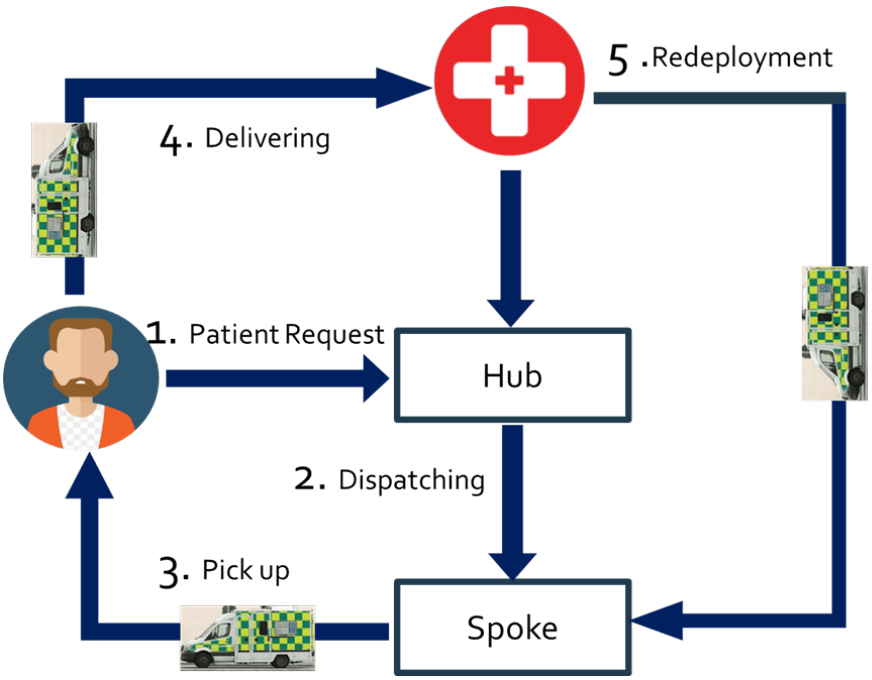
– Incidents occur with a random jitter of ±2 km in x and y around incident location centre.

CHAPTER 4: DEEP SCORE NETWORK AND DYNAMIC AMBULANCE REDEPLOYMENT ALGORITHM

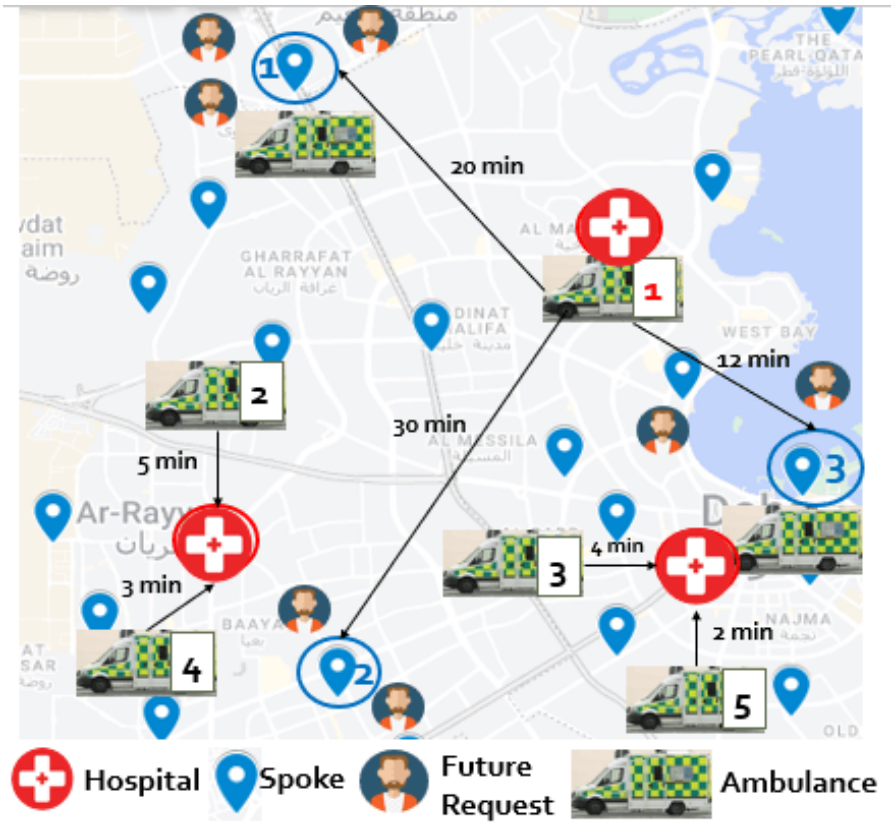
4.1. Problem Definition

Every year, emerging illnesses and catastrophes including heart attacks, cerebral hemorrhages, and traffic accidents put millions of lives in danger in cities across the globe [50]–[52]. For instance, every year more than 100,000 inhabitants of Qatar are impacted by serious illnesses or accidents, necessitating the prompt transportation of these individuals to medical facilities via ambulance services [53]. By quickly deploying ambulances to pick up patients and bring them to hospitals, EMS play a vital role in saving lives. Every second matters for patients in severe condition, and the sooner they receive medical attention, the better their prognosis. For this reason, an EMS system must develop solutions to shorten patient response times.

The process of ambulance service involves several steps, as explained in 1.2 and illustrated in 4.1a. The process in Figure 4.1a starts with the patient request (Step 1). When a patient or someone on behalf of the patient contacts emergency services, they provide information about the patient's condition and location. Based on this information, the dispatcher (at the hub) evaluates the situation and dispatches the closest available ambulance to the scene. Once dispatched at Step 2, the ambulance crew, typically consisting of Emergency Medical Technicians (EMTs) or paramedics, arrives at the location. They assess the patient's condition, provide necessary medical treatment, and stabilize the patient if needed. The crew then pick up the patient (Step 3) and transports the patient to the hospital (Step 4), taking into account the severity of the condition and the nearest appropriate facility for treatment. After delivering the patient, the ambulance is ready to be redeployed at Step 5. Redeployment involves returning to



(a) Ambulance Redeployment



(b) Multiple Dynamic Factors

Figure 4.1. Redeployment of ambulances and the various dynamic aspects should be taken into account

a spoke station and being available for the next call.

Step 5 in Figure 4.1a illustrates how the dynamic redeployment strategy of mobile ambulances affects patient response times. When an ambulance becomes available, this approach selects which spoke station it should be redeployed to. For example, ambulance 1 in Figure 4.1b can be redeployed to one of the ambulance spoke stations in the city once it has taken a patient to a hospital and is now available. Future patient response times are impacted by the repurposing of existing ambulances. For instance, if three patients are anticipated to be close to station 1 in the future, redeploying ambulance 1—which is currently available—to station 1 may enable prompt ambulance dispatching from station 1 to pick up these patients.

Although the response time of patients is also affected by the dispatching of ambulances (step 2 in Figure 4.1a), in practice, the closest ambulance is typically called to pick up a patient [36], [54], [55]. This is due to the fact that the greedy dispatching approach has already attained competitive performance, which makes the development of an improved dispatching method challenging [56], [57]. We thus concentrate on researching the redeployment approach.

However, because several factors need to be taken into account and balanced at the same time, dynamic ambulance redeployment is complex. To be more precise, each spoke station in Figure 4.1b has the following dynamic factors influencing whether ambulance 1 should be redeployed to:

1. The arrival rate near this spoke station in the future is a crucial factor. If more patients are expected to be nearby, redeploying the currently available ambulance to this spoke station would be more beneficial.

2. The current availability of ambulances at this spoke station is crucial. The fewer

ambulances available at a given spoke, the more urgent it becomes for the emergency hub to redeploy an ambulance to the spoke.

3. Another crucial factor to consider is the travel time required for the currently available ambulance to reach this spoke station. If the ambulance is too far away, redeploying it to this spoke station might not be feasible, as it would spend a significant amount of time traveling empty.

4. Another factor is the current and future status of other ambulances that are in service. As an example, ambulances 2 and 4 in Figure 4.1b are currently occupied, but they will become available at a hospital that is closer to station 2 than ambulance 1. Since ambulances 2 and 4 can be redeployed to station 2 in the near future, it becomes less required in this scenario to redeploy the current ambulance to that location.

Therefore, it is a complex optimization problem to choose a spoke station that achieves a good balance between these four factors, as various factors have varying preferences for spoke stations. In Figure 4.1b, for instance, factor 1 chooses to redeploy ambulance 1 to station 1 because it anticipates receiving the greatest number of nearby requests. Nevertheless, since station 2 is devoid of ambulances, ambulance 1 should be redeployed there if just factor 2 is taken into account. In a similar vein, station 3 is the most appropriate station for factor 3 because it is nearest to ambulance 1. The condition of the occupied ambulances 2–5 (factor 4) should also be taken into account in addition to these three variables. It is therefore difficult to quantitatively balance every factor.

Choosing an appropriate spoke station becomes more challenging in real-world EMS systems because there may be several dozen spoke stations and occupied ambulances. Previous techniques of ambulance redeployment typically involved the manual

construction of indicators to integrate these complex aspects into a single factor [37], [43], [58]. These indicators, however, only take into account one or a few of these factors. For example, the most commonly used indicator, which simply takes into account factors 1 and 2, is the coverage of a spoke station. An available ambulance is then redeployed to the spoke station with the least coverage, based on each spoke station's current coverage [43]. Alternatively, certain greedy approaches might immediately reallocate an available ambulance to the closest spoke station (factor 3), or the spoke station with the least amount of ambulances (factor 2).

Furthermore, many static redeployment techniques redeploy an ambulance to its base spoke station straight away, disregarding any dynamic factors specific to each spoke station [38], [46], [58]. Since these intricate factors are nearly impossible for handcrafted rules to balance, it is evident that these manually developed indicators are difficult to optimize for patient response times.

To address this challenge, we introduce a straightforward yet innovative dynamic ambulance redeployment method. This method effectively balances all the complex factors mentioned earlier. In this section, we first explore the Deep Score Network, which consolidates a station's dynamic factors into a single score (Section 4.3). Subsequently, we introduce a RL framework designed to train the score network (Section 4.3.1). Finally, we elaborate on the method for dynamic ambulance redeployment based on the Deep Score Network (Section 4.4).

The notations used are summarized in Table 4.1.

Table 4.1. Description of Parameters for Dynamic Ambulance Redeployment

Parameters	**Description**

j	Spoke station index
J	The number of spoke stations
i	Available ambulance index
I	The number of available ambulances
z	Occupied ambulance index
Z	The number of occupied ambulances
p	Patient requests
P	The number of patient requests
h	Hospital
x_j	Dynamic factors for each spoke station j
$\lambda_{1j}, \ldots, \lambda_{mj}$	The arrival rate of patient requests nearby spoke station j in the future m period
n_j	The number of available ambulances i that spoke station j currently has
l_j	The geographical location of each spoke station j
e_j	The expected travel time between spoke station j and the current available ambulance
$b_{1j}, \ldots, b_{zj}$	The travel time between spoke station j and each occupied ambulance z
t_p	Constant time spent at patient location
t_h	Constant time spent at the hospital
t_{hp}	The travel time to reach hospital h from patient p
t_{hj}	The travel time to reach spoke station j from hospital h
PT_j	Pre-dispatch time to reach spoke station j
RT_j	Response time from spoke station j
DT	Constant dispatch time

PU	Total number of patients picked up within threshold τ minutes by all ambulances from spoke station j
s_t	State of the environment
a_t	Action taken (redeployment order)
r_t	Reward gained
$\alpha_1, \alpha_2, \alpha_3$	Weighing factors

4.2. Deep Score Network

Figure 4.2 shows our deep score network. The network is made up of an output layer, two hidden layers, and inputs. The inputs are represented by x_j, which stands for the current factors of each spoke station j. These dynamic circumstances impact the decision to redeploy the ambulance that is now available. Each hidden layer has a customizable number of neurons; for the sake of this study, we set both to 20. A $\tanh$ activation function is applied after each hidden layer, creating a nonlinear relationship between the input components and the output score. In deep learning, the $\tanh$ activation function is frequently employed [59]. The result is y_j, which is the score of spoke station j. The model learn a scoring function, $y_j = f(x_j; \theta)$. The parameters of the neural network layers are donted as θ.

In Section 4.3, a DRL framework is used to learn the weights θ in the score network, where the weights are shared among all spoke stations.

As illustrated in Figure 4.2, the factors x_j for each station j are $(m+1+1+k)$-dimensional vectors:

$$x_j = (\lambda_{1j}, \ldots, \lambda_{mj}, n_j, e_j, b_{1j}, \ldots, b_{zj}), \tag{4.1}$$

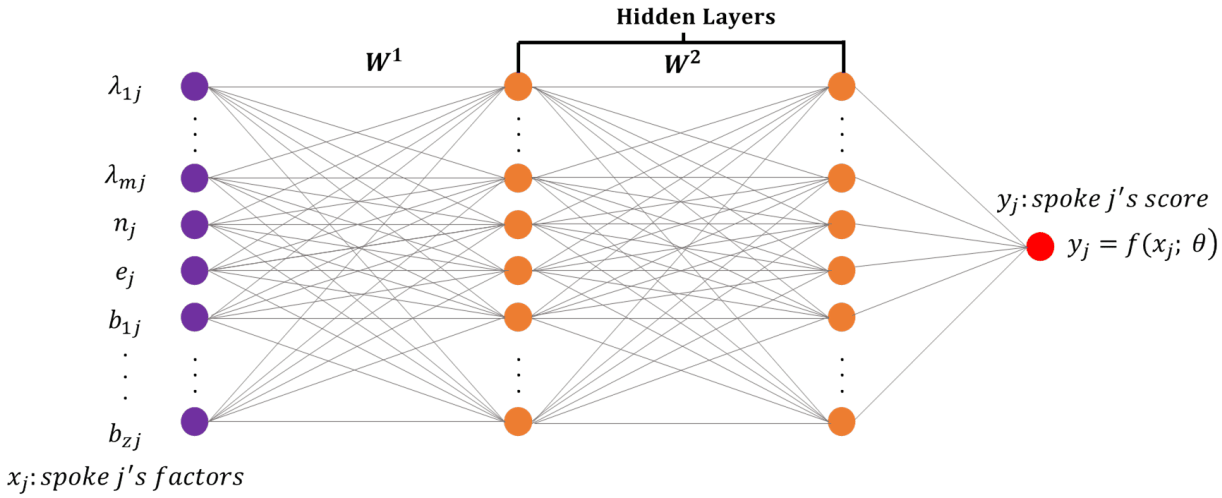


Figure 4.2. Deep Score Network. The current factors x_j of a spoke station are input; the station's score, y_j, is output.

where $(\lambda_{1j}, \ldots, \lambda_{mj})$, n_j, e_j, and $b_{1j}, \ldots, b_{zj}$ correspond to factors 1, 2, 3, and 4, respectively, as discussed in the Problem Definition (Section 4.1). Below, we introduce the detail of each factor in x_j.

4.2.1. Dynamic Factors

Factor 1. $\lambda_{1j}, \ldots, \lambda_{mj}$ is the rate at which patient requests arrive at spoke station j, e.g, the number of patient requests in the upcoming hour and the hour following that. The number of future time periods that are being considered is indicated by m. A half-hour or an hour might be considered a time span. In terms of the time it takes for ambulances to travel on road networks, we define an ambulance station j to be nearby an EMS request if that station is the closest station to the request.

To forecast the values of $\lambda_{1j}, \ldots, \lambda_{mj}$, we used a ML technique called Random Forest (RF) regression. This approach utilizes an ensemble of decision trees to predict the arrival rate. The RF regression model included the date, day of the week, and time of the day as features.

The arrival rate was calculated based on the number of requests received at each spoke station within a specific time window (30 mins). During the training process, historical data, including the arrival rates and corresponding dates, days of the week, and times of the day, was used to train the RF regression model.

Once the RF regression model was trained, it is employed to predict the arrival rate for future time periods. The model takes as input the date, day of the week, and time of the day for the future time and generates the predicted arrival rate as output. The performance of the RF regression model was evaluated using metrics such as mean squared error (MSE), and achieved accuracy of 97.76%.

Figure 4.3 illustrates the predicted arrival rate for each hour of the day. As expected, we can see peak hours have high arrival rates and off-peak hours have small arrival rates.

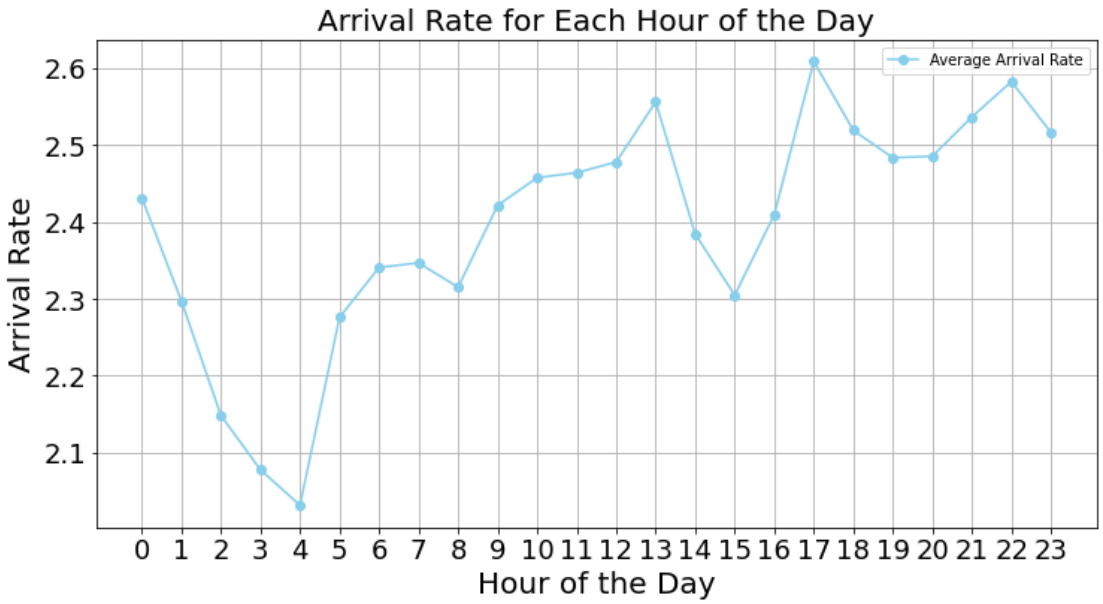


Figure 4.3. Arrival Rate for Each Hour of the Day

Factor 2. n_j represents the number of ambulances currently stationed at spoke station j. This information is readily available from the perspective of the EMS hub in

an EMS system.

Factor 3. e_j denotes the estimated travel time required for the currently available ambulance i to reach spoke station j if redeployed there. This estimate considers factors such as traffic conditions. Various methods, such as those proposed in [60]–[62], can be employed for travel time estimation. In our method, we evaluate the impact of travel time estimation errors on dynamic ambulance redeployment and find that using established travel time estimation methods, such as [63], effectively mitigates these errors.

Factor 4. $b_{1j}, \ldots, b_{zj}$ represent the travel time between spoke station j and each occupied ambulance z. Similar to e_j, these values are also estimates.

We consider only ambulances transporting patients to hospitals, for which we can estimate the travel time from the patient's location to the hospital (t_{hp}) and from the hospital h to spoke station j (t_{hj}).

The time at the patient pickup location t_p and at the hospital t_h are constant (e.g., for patient handover, paperwork, etc.). We are unable to predict the time it will take an ambulance to get at spoke station j since we do not know which hospitals the ambulances heading to patient scenes will be visiting.

The travel time between spoke station j and an occupied ambulance z, denoted as b_{zj} can be expressed in Equation 4.2 and is illustrated in Figure 1.2:

$$b_{zj} = t_p + t_{hp} + t_h + t_{hj} \tag{4.2}$$

4.3. Reinforcement Learning Deep Score Network

In this section, we propose learning the weights θ of the score network using a RL framework [64], [65], which is based on policy gradient [66]. This is because we

need a labeled score y_j given the (dynamic) factors x_j of a spoke station j. Thus, no supervised learning algorithm can be used to learn the θ.

4.3.1. Reinforcement Learning Framework

The dynamic ambulance redeployment problem can be formulated as an RL task, which involves five main concepts: state, action, transition, reward, and policy. These concepts are fundamental in understanding and solving the dynamic ambulance redeployment problem using RL. We present each of these concepts in the context of the dynamic redeployment challenge for ambulances below.

State. As soon as an ambulance becomes available, we can characterize the system status as s_t. This state s_t should contain all the data needed to deploy the ambulance that is currently on hand again. Consequently, the factors of every ambulance spoke station are included in s_t.

$$s_t = (x_1, x_2, \ldots, x_J) \tag{4.3}$$

where J is the number of ambulance spoke stations in the EMS system and x_j refers to the factors of ambulance spoke station j as specified in Equation 4.1.

Action. The action in the ambulance redeployment problem is the same as choosing a spoke station to which the available ambulance will be sent at this time. Consequently, the current operation, indicated by a_t,

$$a_t \in \{1, 2, \ldots, J\} \tag{4.4}$$

where $a_t = j$ denotes the redeployment of the available ambulance to spoke station j.

Transition. Until another ambulance becomes available, no further action is conducted after taking action a_t in the current state s_t. The following state s_{t+1} is entered by the system when a new ambulance is made available. For instance, let's say that the time slot is 7:45 am and the current step is t. At 7:58 a.m., the system will transition to state s_{t+1} if a new ambulance becomes available. There is no set time period between two states.

Reward. In the context of the dynamic ambulance redeployment problem, the reward function plays a crucial role in guiding the learning process of the RL algorithm. The goal of the reward function is to quantify the desirability of different states and actions, providing the agent with feedback on its decisions. In this specific formulation, the reward is designed to achieve several objectives. Firstly, it aims to maximize the number of picked-up patients within a specified time threshold, ensuring efficient utilization of available ambulances. Secondly, it seeks to minimize the response time from spoke station to the patient, enhancing the system's ability to respond promptly to emergencies. Finally, the reward function aims to minimize the pre-dispatch time from available ambulances to spoke stations, reducing delays in providing medical assistance. By adjusting the weighting factors α_1, α_2, and α_3, it allows us to prioritize different aspects of the system based on the requirements. The reward function can be formulated as in Equation 4.5

$$r_t(s_t, a_t) = \alpha_1 \sum_{\tau=1}^{P} Pickup_p - \alpha_2 \sum_{j=1}^{J} RT_j - \alpha_3 \sum_{j=1}^{J} PT_j \tag{4.5}$$

Equation 4.5 defines the reward function for the RL framework in the context of dynamic ambulance redeployment. The equation aims to quantify the desirability of different states and actions, providing feedback to the RL algorithm to improve the

efficiency and effectiveness of ambulance redeployment.

In Equation 4.5, the reward function $r_t(s_t, a_t)$ at time t is composed of three terms weighted by factors α_1, α_2, and α_3. The first term, $\alpha_1 \sum_{\tau=1}^{\tau} Pickup_\tau$, defined in Equation 1.2, encourages maximizing the total number of patients picked up within a specified time frame. This term promotes efficient utilization of available ambulances. The second term, $-\alpha_2 \sum_{j=1}^{J} RT_j$, also defined as in Equation 1.1 ($RT = DT + t_{jp}$), aims to minimize the total response time from spoke stations to patients. It considers the constant dispatch time DT and the travel time of the ambulance from the spoke station to the patient t_{jp}. The third term, $-\alpha_3 \sum_{j=1}^{J} PT_j$, seeks to minimize the total pre-dispatch (PT_j) time from available ambulances to spoke stations.

The weighting factors α_1, α_2, and α_3 allow for adjusting the importance of these objectives in the reward function based on the specific requirements of the ambulance redeployment system.

Policy. In this approach, an action is selected depending on the current states s_t using the policy $\pi_\theta(s_t, a_t)$. The probability of choosing an action a_t in light of the current spoke situation s_t is represented by this policy. For this, a policy network is used, as shown in Figure 4.4. For each spoke station j, the policy network determines the score as follows: $y_j = f(x_j; \theta)$, given the current state $s_t = (x_1, x_2, \ldots, x_J)$.

To carry out this calculation, the policy network combines the score network shown in Figure 4.2. Notably, there is only one set of weights θ in the policy network, meaning that various spoke stations use the same set of parameters θ. Our goal is to learn a single scoring network for all stations instead of different networks for every spoke station, which is reflected in this design decision. A softmax function is used to determine the probability $\pi_\theta(s_t, a_t = j)$ for each potential action j (spoke station j) after

calculating each spoke station j's score y_j. RL frequently uses the softmax function, which may be represented as follows [66], [67]:

$$\pi_\theta(s_t, a_t = j) = \frac{\exp(f(x_j;\theta))}{\sum_{j=1}^{J} \exp(f(x_j;\theta))} \tag{4.6}$$

The likelihood that an ambulance will be redeployed to a spoke station with a better score is higher. The aim of RL is to develop an optimal policy network (i.e., policy $\pi_\theta(s_t, a_t)$) to maximize the rewards received by adhering to the policy, as explained in Section 4.3.2. As a result, our goal is achieved since the learnt policy network uses the score network's weights.

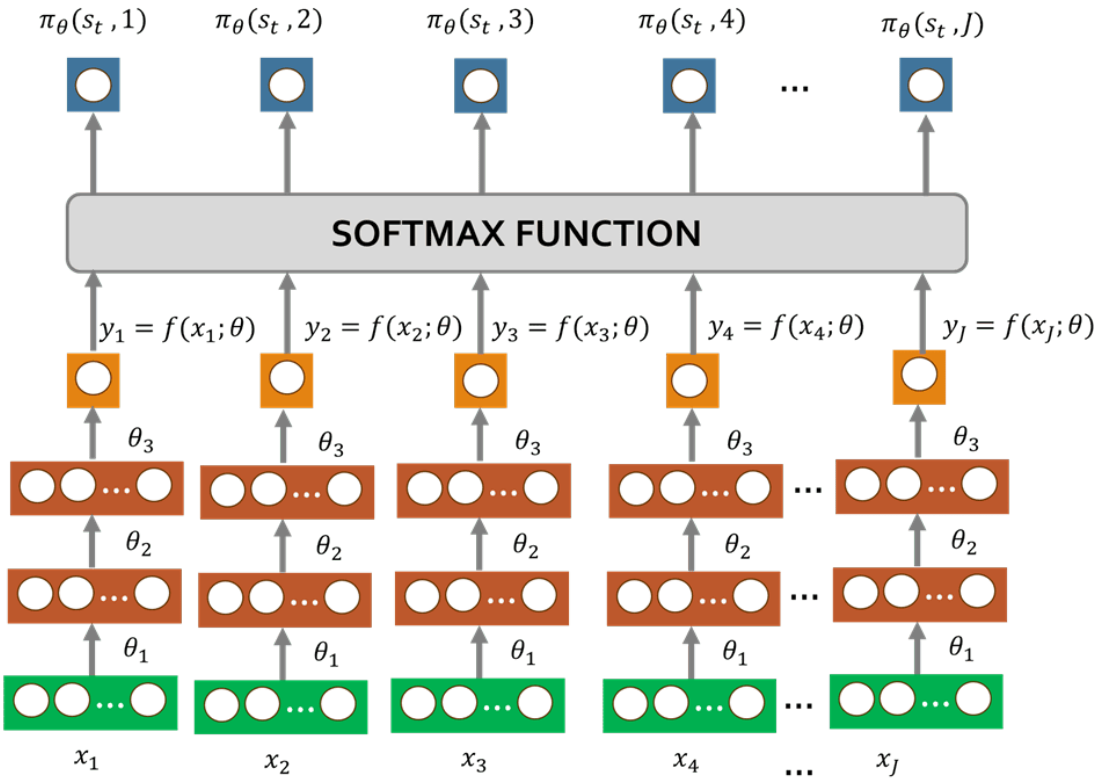


Figure 4.4. Policy Network. For each x_j, $\theta = (\theta_1, \theta_2, \theta_3)$ stays the same, i.e., only one θ shared by all stations.

4.3.2. Learning θ With Policy Gradient

Objective. By training an optimal policy network, represented by the optimal weights θ, the goal of RL is to enable an agent to maximize expected long-term discounted reward by following the policy π_θ, given any state s. This can be stated by the objective function:

$$\max_{\theta} J(\theta) = \mathbb{E}_{s\sim\pi_\theta}[v(s)] \tag{4.7}$$

where $v(s)$ refers to the expected long-term discounted reward starting from state s and following policy π_θ. Moreover, $s \sim \pi_\theta$ means that state s is sampled following policy π_θ, starting from any random state. In addition, the discounted reward $v(s)$ is written as:

$$v(s) = \mathbb{E}[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots | s = s_t], \tag{4.8}$$

where the discounted ratio (e.g., $\gamma = 0.99$) of future rewards is $\gamma \in [0, 1]$. The state-value is another name for $v(s)$ [66]. Furthermore, by applying policy π_θ, represented by $q(s, a)$, we may calculate the expected long-term discounted rewards for all states s and action a.

$$q(s, a) = \mathbb{E}[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots | s = s_t, a = a_t] \tag{4.9}$$

The state-action value is denoted by $q(s, a)$ [66]. The relationship between state-value $v(s)$ and state-action value $q(s, a)$ is as follows, per the definitions [66].

$$v(s) = \sum_{a \in A} \pi_\theta(s, a) q(s, a) \tag{4.10}$$

Figure 4.5 illustrates the relationship between $v(s)$ and $q(s, a)$. Each action is taken with probability $\pi_\theta(s, a)$, connected with state-action value $q(s, a)$, given states, by adhering to policy π_θ. State value $v(s)$, or Equation 4.10, is hence the probability expectation of state-action value $q(s, a)$.

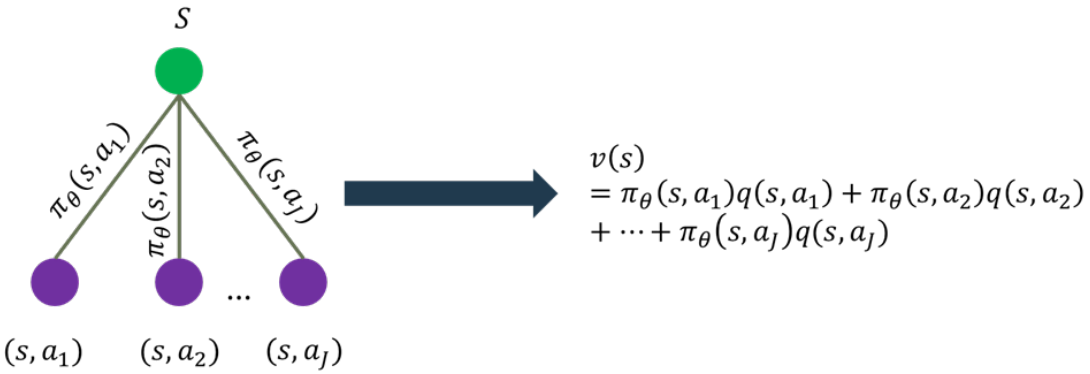


Figure 4.5. The Relation Between $v(s)$ and $q(s, a)$

Gradient. Policy gradient methods can be used to maximize the objective function [65], [66]. The gradient of $J(\theta)$ with respoect to θ can be obtained by combining Equations 4.7 and 4.10 and using the methodology described in [66], [67].

$$\nabla_\theta J(\theta) = \mathbb{E}_{(s,a)\sim\pi_\theta} \left[(\nabla_\theta \log \pi_\theta(s, a)) \cdot q(s, a) \right]. \tag{4.11}$$

where $(s, a) \sim \pi_\theta$ represents the state-action pair (s, a), which is sampled based on any random state and policy π_θ. Appendix A has the derivation. Equation 4.11's gradient allows us to update θ as

$$\theta \leftarrow \theta + \alpha \cdot \nabla_\theta J(\theta), \tag{4.12}$$

where the learning rate is denoted by α (e.g., $\alpha = 0.005$). It's important to note that the goal of this problem is to maximize $J(\theta)$. Consequently, we use $\theta + \alpha \cdot \nabla_\theta J(\theta)$ to update θ rather than $\theta - \alpha \cdot \nabla_\theta J(\theta)$.

4.4. Dynamic Redeployment Algorithm

The policy network functions as an algorithm for ambulance redeployment. Once trained, it can efficiently redeploy available ambulances. However, its use of a softmax function to select actions (stations) introduces a degree of randomness. This randomness aids in exploring more state-action pairs during the score network's learning process, enhancing learning effectiveness.

Our redeployment algorithm assigns ambulances to stations based on their scores. It is activated whenever an ambulance becomes available for redeployment.

Algorithm 1 starts by initializing the weights θ randomly and setting the learning rate α. It then enters a loop for each episode, where it resets the environment to its initial state and samples a state s_t from the environment. Within each episode, it repeats the following steps: the policy π_θ is used to select an action a_t, which is executed in the environment. The resulting next state s_{t+1} and reward r_t are observed, and the state-action value $q(s_t, a_t)$ is calculated. The weights θ are updated using the policy gradient method to maximize the objective function $J(\theta)$, which represents the expected long-term discounted reward. This process continues until the episode ends, at which point the learned weights θ are returned.

Our redeployment method takes three steps to redeploy an available ambulance, as illustrated in Algorithm 2. Equation 4.1 describes how to first extract the current factors x_j for each station j (i.e., the present status of the EMS system). Secondly, we

Algorithm 1 Learning θ

procedure LEARNINGSCORENET
    $\theta \leftarrow$ random initialization
    Set learning rate α
    Set maximum number of episodes or iterations
    for each episode **do**
        Reset environment to initial state
        Sample state s_t from environment
        repeat
            Use policy π_θ to select action a_t
            Execute action a_t, observe next state s_{t+1} and reward r_t
            Calculate state-action value $q(s_t, a_t)$ (Equation 4.9)
            Update weights θ using policy gradient (Equation 4.12)
            Move to next state s_{t+1}
        until episode ends
    end for
    Return learned weights θ
end procedure

Algorithm 2 Dynamic Ambulance Redeployment Algorithm

procedure REDEPLOY
    Acquire every station j's current factors x_j
    Compute every station j's current score $y_j = f(x_j; \theta)$
    Obtain the chosen station $j^* = \arg\max_j \{y_j = f(x_j; \theta)\}$
    return j^*
end procedure

use the scoring network to compute the score y_j of each station j, i.e., $y_j = f(x_j; \theta)$. Thirdly, we redeploy the available ambulance to the station j^* that has the highest score. In other words,

$$j^* = \text{argmax}_j\{y_j = f(x_j; \theta)\}. \tag{4.13}$$

We utilize the DRL algorithm (Algorithm 1) to train the deep scoring network using the EMS requests from the preceding 31 days, from January 1st 2022 to February 1st 2022. The remaining 20 days (February 1–21, 2022) are used as test data to evaluate the redeployment algorithm (Algorithm 2) based on the learned deep score network, as illustrated in Figure 4.6.

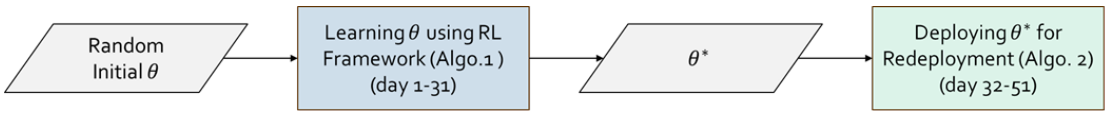


Figure 4.6. Relationship between DRL Algorithm 1 and ambulance redeployment Algorithm 2

CHAPTER 5: EVALUATION OF PROPOSED DYNAMIC AMBULANCE REDEPLOYMENT ALGORITHM

In order to evaluate our dynamic ambulance redeployment approach's performance effectively, we contrast it with several baseline redeployment strategies, explained in the literature review in Section 2.3, including MCLP, RA,NS,LS,ERTM,MEXCLP and DMEXCLP approaches.

To provide a realistic benchmark for evaluating the effectiveness of our dynamic ambulance redeployment approach, we also consider the actual performance derived from real-world data as a baseline. The actual performance represents the response times and coverage achieved by the existing ambulance deployment strategy without any optimization.

This baseline helps us understand the improvement our approach can achieve over the current system in terms of response times, coverage, and overall efficiency. By comparing the actual performance with the results of our dynamic redeployment approach, we can assess the impact and effectiveness of our proposed solution in enhancing EMS.

The actual performance baseline serves as a practical reference point, reflecting the current operational status of the HMCAS. It provides valuable insights into the challenges and limitations of the existing system, highlighting the areas where improvements are most needed.

5.1. Performance Metrics

Two standard metrics are commonly employed in the literature to evaluate the effectiveness of ambulance redeployment techniques. The average response time (AveRT)

in seconds for all patients is the first metric. It is calculated as the average of the response times of individual patient requests, as shown in Equation 5.1, where P is the total number of patient requests and RT_p is the response time of patient request p. A lower AveRT indicates better performance of the redeployment method.

$$\text{AveRT} = \frac{1}{P}\sum_{p=1}^{P} RT_p \tag{5.1}$$

The second metric is the ratio of patients picked up within a specified time threshold, denoted as RelaRT (relative response time). This metric is calculated as the proportion of patient requests with response times less than or equal to the threshold RT_{t*} (In our experiments the threshold used was 10 minutes), as shown in Equation 7.1. A higher RelaRT indicates that more patients are being picked up within the specified time threshold, which is desirable for a redeployment method.

$$\text{RelaRT} = \frac{1}{P}\sum_{p=1}^{P} ||RT_{tp} \leq RT_{t*}|| \tag{5.2}$$

5.2. Effectiveness of Proposed Redeployment Method

This section evaluates the effectiveness of our dynamic ambulance redeployment method compared to various baseline techniques. We conduct extensive simulations for each approach, evaluating AveRT and RelaRT performance metrics. In our EMS system, the number of ambulances (represented by I) is fixed at 60, 70, ..., 100.

Table 5.1 summarizes the comparison results for standard ambulances. Our proposed dynamic redeployment method consistently outperforms the baseline methods and actual performance from real-world data across different scenarios. For example,

at $I = 70$, our method achieves an AveRT of 360.75 sec and a RelaRT of 0.812. In comparison, the best-performing baseline method, ERTM, achieves an AveRT of 412.0 sec and a RelaRT of 0.810 under the same conditions, resulting in a 12.61% reduction in AveRT and a 0.25% increase in RelaRT.

As the number of ambulances increases, our method maintains superior performance. At $I = 80$, our method achieves an AveRT of 340.50 sec and a RelaRT of 0.825, compared to LS, which achieves an AveRT of 470.1 sec and a RelaRT of 0.820, indicating a 27.54% reduction in AveRT. Similarly, at $I = 90$, our method achieves an AveRT of 330.25 sec and a RelaRT of 0.835, compared to LS, which achieves an AveRT of 435.0 sec and a RelaRT of 0.837, resulting in a 24.00% reduction in AveRT.

Comparing our method to actual performance reveals significant improvements in ambulance redeployment strategies. For example, at $I = 70$, our method achieves an AveRT of 360.75 sec and a RelaRT of 0.812, compared to the actual performance of 462.8 sec AveRT and 0.678 RelaRT, indicating a 21.99% reduction in AveRT and a 19.88% increase in RelaRT. This trend continues as the number of ambulances increases, with our method consistently achieving lower AveRT and higher RelaRT values compared to actual performance, demonstrating the effectiveness of our approach in enhancing EMS efficiency and response times.

Additionally, our method outperforms all other baseline methods, including MCLP, RA, NS, ERTM, MEXCLP, and DMEXCLP, across different scenarios. For example, at $I = 80$, our method achieves an AveRT of 340.50 sec and a RelaRT of 0.825, while DMEXCLP achieves an AveRT of 393.0 sec and a RelaRT of 0.798, resulting in a 13.36% reduction in AveRT and a 3.38% increase in RelaRT. This trend continues across all scenarios, highlighting the robustness and effectiveness of our approach in

Table 5.1. Comparisons with Baseline Methods for Dynamic Ambulance Redeployment

Methods	I = 60		I = 70		I = 80		I = 90		I = 100	
	AveRT	RelaRT	AveRT	RelaRT	AveRT	RelaRT	AveRT	RelaRT	AveRT	RelaRT
MCLP	610.7	0.589	571.9	0.619	550.6	0.634	536.1	0.684	524.2	0.659
RA	805.4	0.541	745.9	0.579	717.2	0.596	697.5	0.610	683.4	0.621
NS	770.0	0.586	764.2	0.590	750.2	0.602	744.7	0.607	733.2	0.616
LS	591.6	0.755	520.1	0.796	470.1	0.820	435.0	0.837	413.4	0.838
ERTM	484.4	0.790	412.0	0.810	378.7	0.821	369.3	0.839	363.8	0.843
MEXCLP	488.1	0.782	447.1	0.767	396.0	0.772	379.7	0.811	363.7	0.819
DMEXCLP	500.2	0.779	431.9	0.783	393.0	0.798	370.2	0.820	357.3	0.828
Actual	525.3	0.648	462.8	0.678	438.6	0.691	424.7	0.705	413.2	0.717
Our Proposed Method	400.25	0.798	360.75	0.812	340.50	0.825	330.25	0.835	320.25	0.845

improving ambulance redeployment strategies in emergency medical services.

5.3. Time Efficiency

Our ambulance redeployment method is executed on a computer with an 8 GB RAM and 2.11 GHz Intel(R) CPU using Python. Compared to other DRL jobs that necessitate a large number of GPUs, training the score function only takes slightly less than 10 hours using CPUs. Real-time redeployment of an available ambulance only takes a few milliseconds after learning the score function, which satisfies the need for real-world time efficiency.

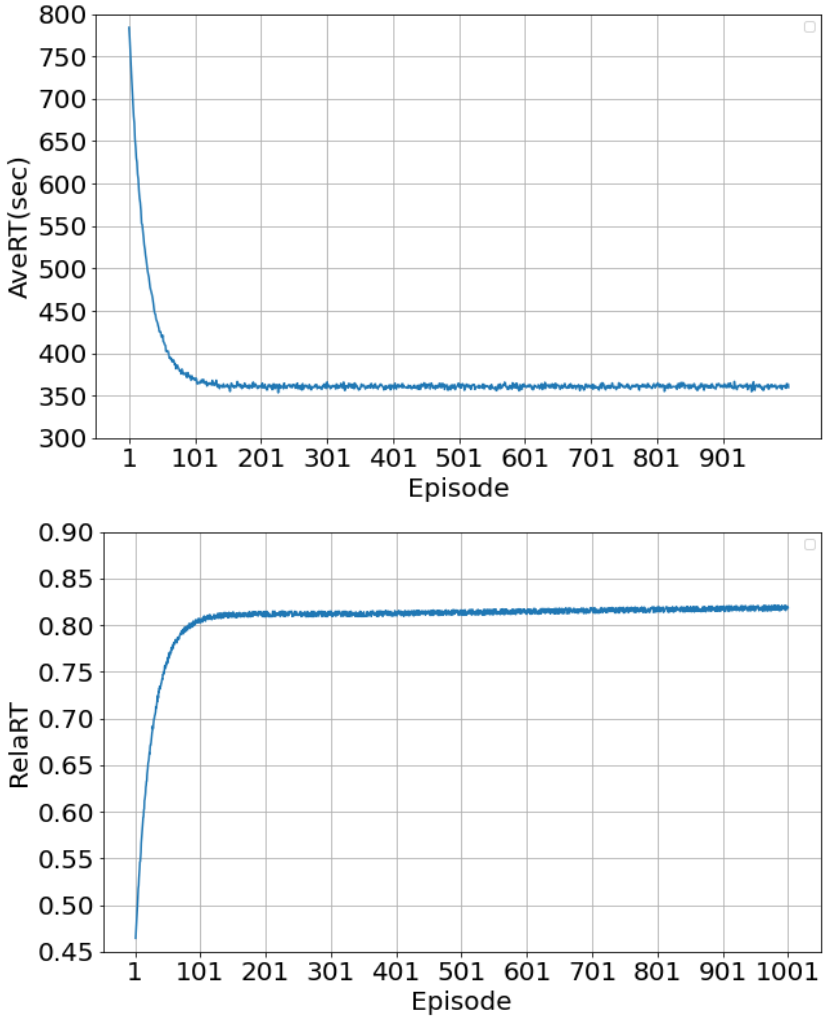


Figure 5.1. Convergence of Training. I=70, m=1

5.4. Convergence of Training

Figure 5.1 displays our scoring function's performance throughout training. It shows that the training is convergent and that the scoring function can quickly converge to a nearly optimal solution.

5.5. Necessity of Considering All Factors

This part examines the necessity of including the dynamic factors of every spoke station into our ambulance redeployment strategy, as delineated in Section 4.3. In order

to achieve this, we look at several combinations of these factors, as shown in Figure 5.2, such as 12, 13, 14, 23, etc.. For example, on the x-axis labeled '12', for each spoke station, only factors 1 and 2 are taken into account, as $x_i = (\lambda_{1j}, \ldots, \lambda_{mj}, n_j)$. After that, we train a new deep scoring network with inputs made up only of factors 1 and 2. Next, we evaluate the scoring network's redeployment performance using only these factors.

Likewise, the '123' axis denotes that factors 1, 2, and 3 are under consideration. We investigate several factor combinations in our large tests using our redeployment approach, and the comparative findings are shown in Figure 5.2. It is clear that optimizing for all four factors results in the best performances, where at 1234, the AveRT was the shortest, and RelaRT was the highest compared to the different combinations of factors.

In addition, we analyze the significance of each factor. We initially define

$$AveRT_f = \frac{1}{|FA_f|} \sum_{fa \in FA_f} AveRT_{fa}$$

as the average response time of each factor f. Here, FA_f represents the set of factors containing factor f, such as $FA_1 = \{12, 13, 14, 123, 124, 134\}$ and $FA_2 = \{12, 23, 24, 123, 124, 234\}$. $AveRT_{fa}$ denotes the average response time of our redeployment method considering factors fa, for instance, $AveRT_{12} = 415.35$, $AveRT_{13} = 556.75$, as shown in the left part of Figure 5.2. Similarly, we can define

$$RelaRT_f = \frac{1}{|FA_f|} \sum_{fa \in FA_f} RelaRT_{fa}$$

for each factor f. Subsequently, $AveRT_f$ and $RelaRT_f$ can be utilized to indicate the importance of each factor f. The more significant the factor f is, the smaller (bigger)

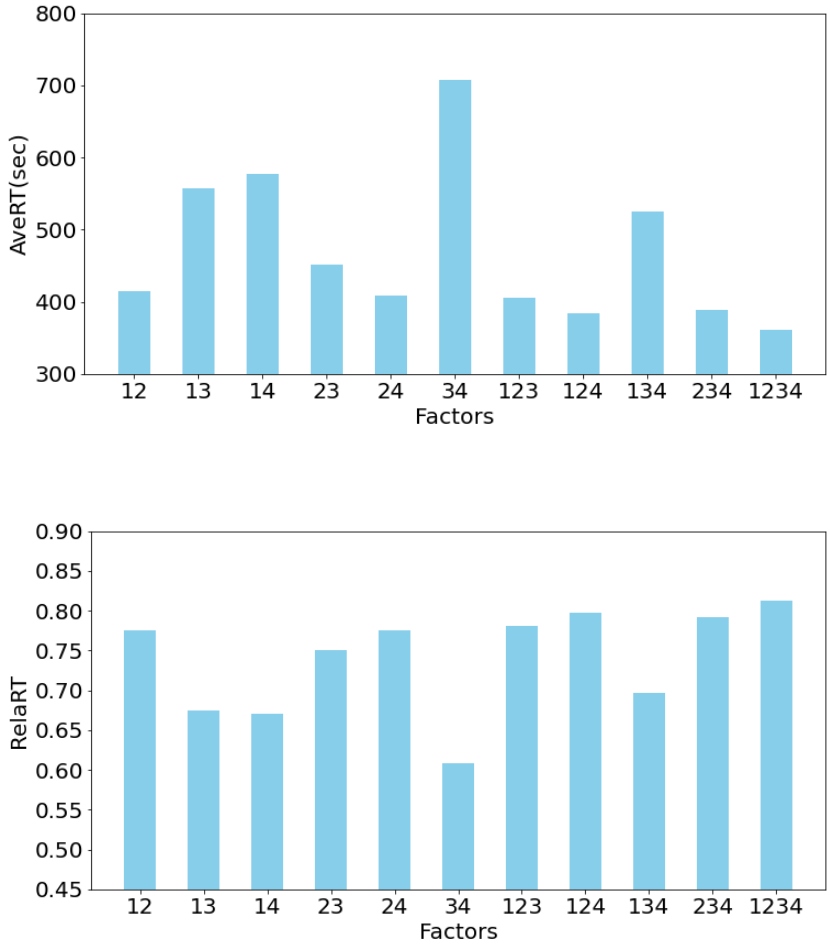


Figure 5.2. Performance of Proposed Method Considering Different Factors. I=70, m=1

the $AveRT_f$ ($RelaRT_f$). As can be seen from Figure 5.3, factor 2 is the most important, followed by factor 1. This is consistent with our intuition that the two most important factors in determining whether an available ambulance should be redeployed to a spoke station are the number of ambulances currently in the spoke station and the arrival rate close to the spoke station.

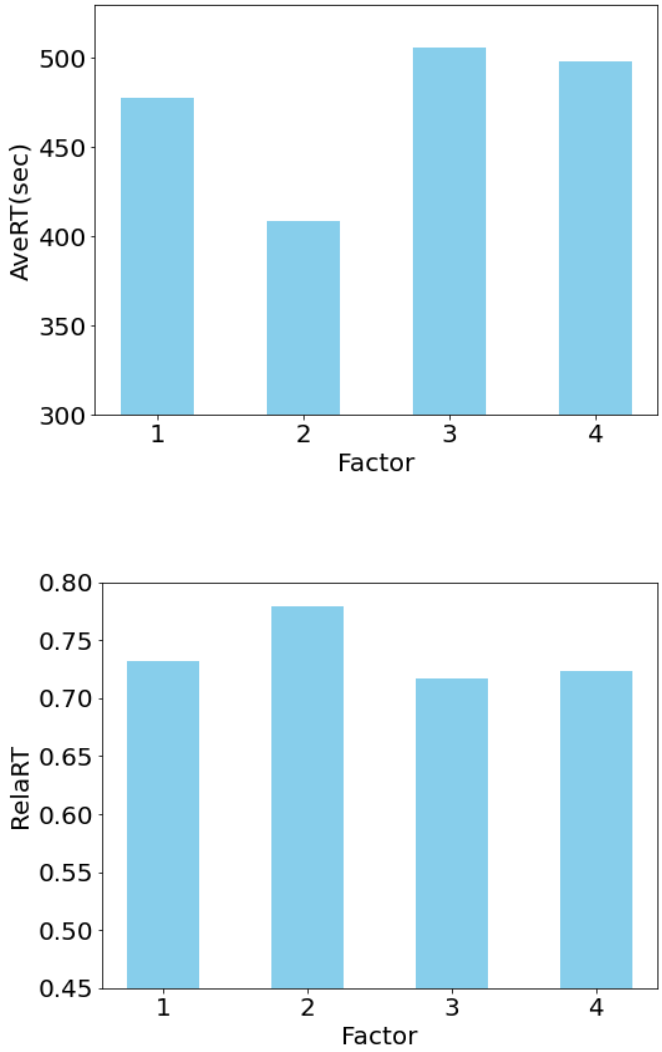


Figure 5.3. Significance of Each Factor. I=70, m=1

5.6. Necessity of Parameter m

We investigate the effect of parameter m, which indicates the number of upcoming timeframes considered for each station, in $\lambda_{1j}, \ldots, \lambda_{mj}$. Figure 5.4 summarizes the experimental results by adjusting m and assessing its effect on AveRT and RelaRT. In Figure 5.4, the x-axis takes into account the various m, and once the scoring network converges and stabilizes, it obtains the AveRT and RelaRT (y-axis). Several $m's$ do not show appreciable variations in our redeployment method's performance, as 5.4 shows. It suggests that our redeployment strategy is robust to various m settings.

5.7. Influence of Number of Patient Requests

We examine the performance of our strategy in terms of the quantity of patient requests at various times of the day. The best course of action for each patient is to minimize response time by having an ambulance dispatched from the closest station (based on journey time). But if there are not any ambulances at the closest station, they have to send one out from another, which takes longer to respond. Thus, we introduce Ratio _AveRT $= \frac{\text{AveRT}}{\text{AveRT\_optimal}}$, which is the ratio of the average response time (AveRT) to the optimal average response time in a time period (e.g., an hour). The average response time of patients is represented here by AveRT_optimal, assuming that ambulances pick up all patients from the closest stations. Similarly, we define Ratio_RelaRT $= \frac{\text{RelaRT}}{\text{RelaRT\_optimal}}$ as the ratio of the relative response time (RelaRT) to the optimal relative response time in each hour. The same assumptions that apply to AveRT_optimal also apply to the computation of RelaRT_optimal. A smaller Ratio_AveRT and a greater Ratio_RelaRT are what we expect. Figure 5.5 shows the relationship between these two ratios and the number of patients in an hour.

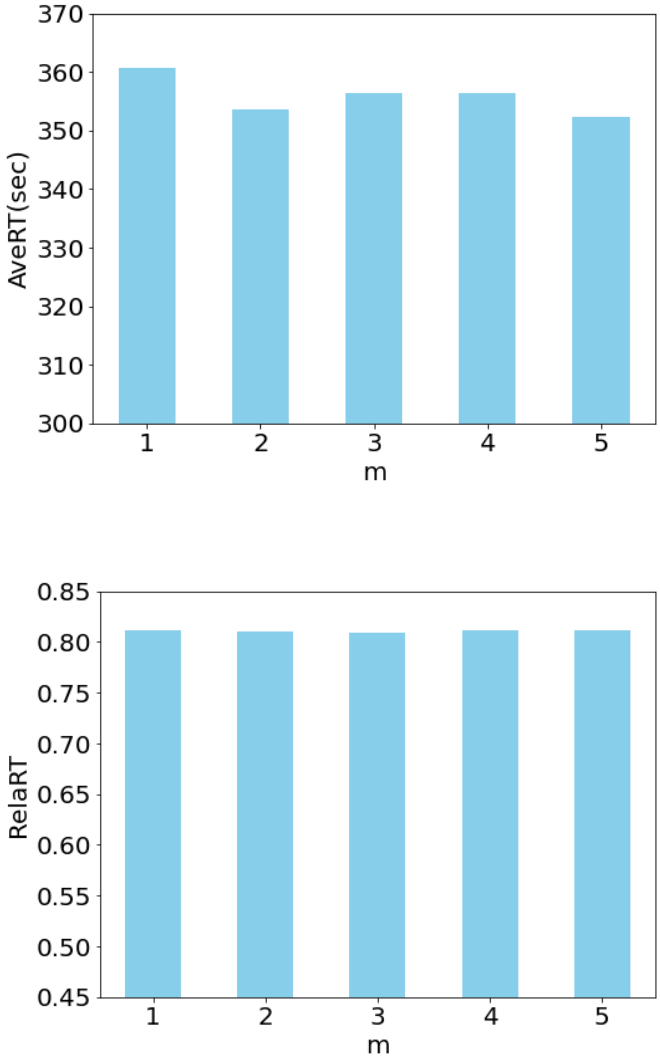


Figure 5.4. Impact of Parameter m to Our Proposed Method. I=70

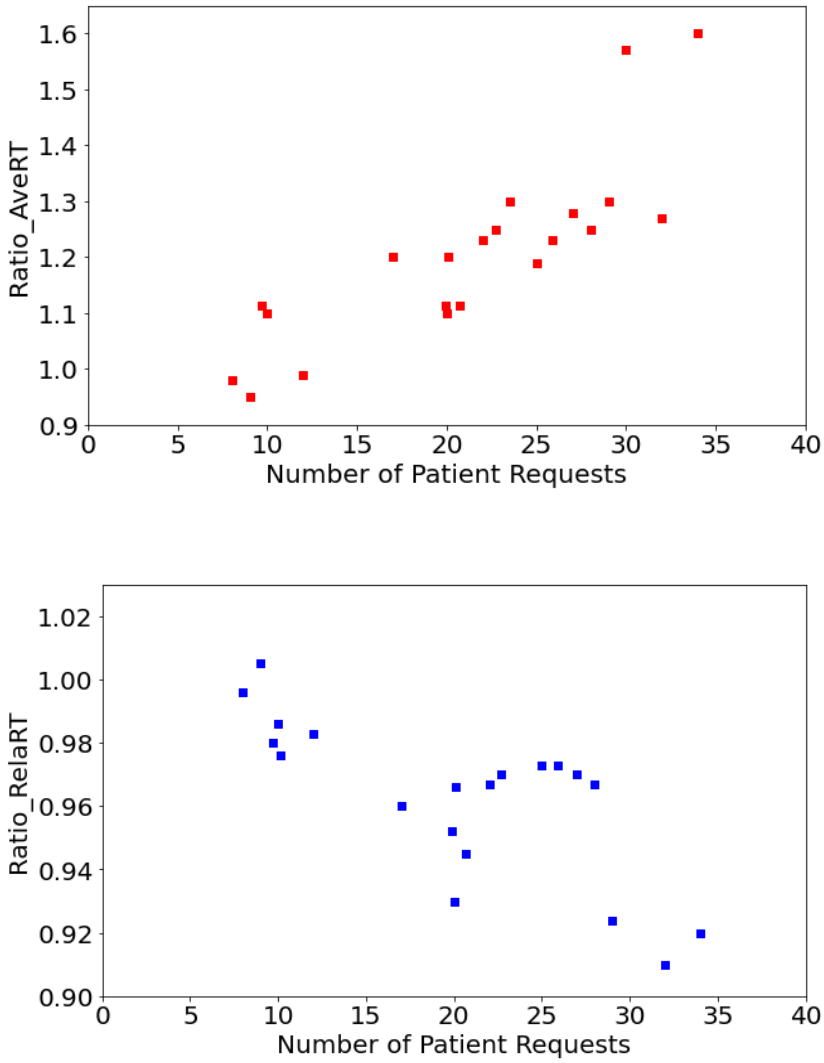


Figure 5.5. Influence of Number of Patient Requests to Our Proposed Method. I=70, m=1

In Figure 5.5, we observe that during time periods with higher patient volumes, the Ratio_AveRT is notably high. This indicates that many patients are not being picked up by ambulances from the nearest stations, resulting in increased response times. Similarly, in time periods with a higher number of patients, the Ratio_RelaRT decreases, indicating that fewer patients are being picked up within 10 minutes. This suggests a greater need for ambulances during these busy periods.

Moreover, we can explore methods to automatically adjust the number of ambulances during different time periods of the day. This could help reduce both patient response times and the operational costs of EMS systems.

5.8. Robustness of Proposed Redeployment Method

This section assesses the dynamic ambulance redeployment method's robustness to different scenarios. We evaluate its robustness against variables including changing traffic circumstances over time, different ambulance capabilities, and human factors. By conducting comprehensive simulations and analyzing the method's performance metrics, we demonstrate its ability to maintain effective redeployment strategies in dynamic and challenging environments. These findings highlight the method's robustness and its potential to enhance emergency medical service efficiency under real-world conditions.

5.8.1. Robust to Traffic Circumstances

Travel time estimation can be inaccurate due to time-varying traffic circumstances, which can also inject noise into factors 3 and 4 (e_j and $b_{1j}, \ldots, b_{zj}$). It is crucial to evaluate how resilient our redeployment strategy is to these traffic conditions (errors in trip time estimation). The travel time estimate e_j is assumed to follow a Gaussian

distribution (same for $b_{1j}, \ldots, b_{zj}$); that is, $e_j \sim N(\tilde{e_j}, (\epsilon\tilde{e_j})^2)$, where N is a Gaussian distribution, $\tilde{e_j}$ denotes the actual travel time, and $\epsilon \geq 0$ is the error rate in the travel time estimation. A lower error rate ϵ for a journey time estimation technique denotes more precise estimation outcomes. According to current travel time estimation methods, the error rate ϵ has been reduced to less than 0.15. Specifically, due to $e_j \sim N(\tilde{e_j}, (\epsilon\tilde{e_j})^2)$, $|e_j - \tilde{e_j}| = \epsilon\sqrt{\frac{2}{\pi}}$. According to recent studies, Mean Absolute Percentage Error (MAPE) $= \mathbb{E}[|\frac{e_j - \tilde{e_j}}{e_j}|]$ with expectation $\mathbb{E}[|\frac{e_j - \tilde{e_j}}{\tilde{e_j}}|]$ MAPE $= \epsilon\sqrt{\frac{\pi}{\sqrt{2}}} < 0.12$, resulting in $\epsilon < 0.15$. $|$ is a half-normal distribution. Thus, we can conclude that $\mathbb{E}[|\frac{e_j - \tilde{e_j}}{\tilde{e_j}}|] < 0.12$.

The performance of our redeployment technique under errors in trip time estimation is depicted in Figure 5.6. This figure demonstrates the system's robustness to errors in trip time estimation due to traffic. The error rate, denoted by ϵ, leads to a gradual increase or decrease in AveRT (RelaRT). Using the literature travel time estimation method [61] with $\epsilon < 0.15$, the influence of estimation errors can be ignored.

5.8.2. Robust to Number of Ambulances "I"

This section looks at how our approach works in different EMS systems with differing ambulance capacities (I). In particular, we study the applicability of the score network learned under one ambulance capacity setting to other EMS systems with varying capacities. We use data from an EMS system with I=60 to train a scoring network, and we test the network's performance on EMS systems with various I values, e.g., 60, 70, . . . , 100.

Figure 5.7 presents the outcomes of the experiment. 'I=60' scores indicate how well the scoring network trained especially for I=60 performed. Conversely, as shown in Table 5.1, the findings labeled 'I=60-100' show how well scoring networks trained

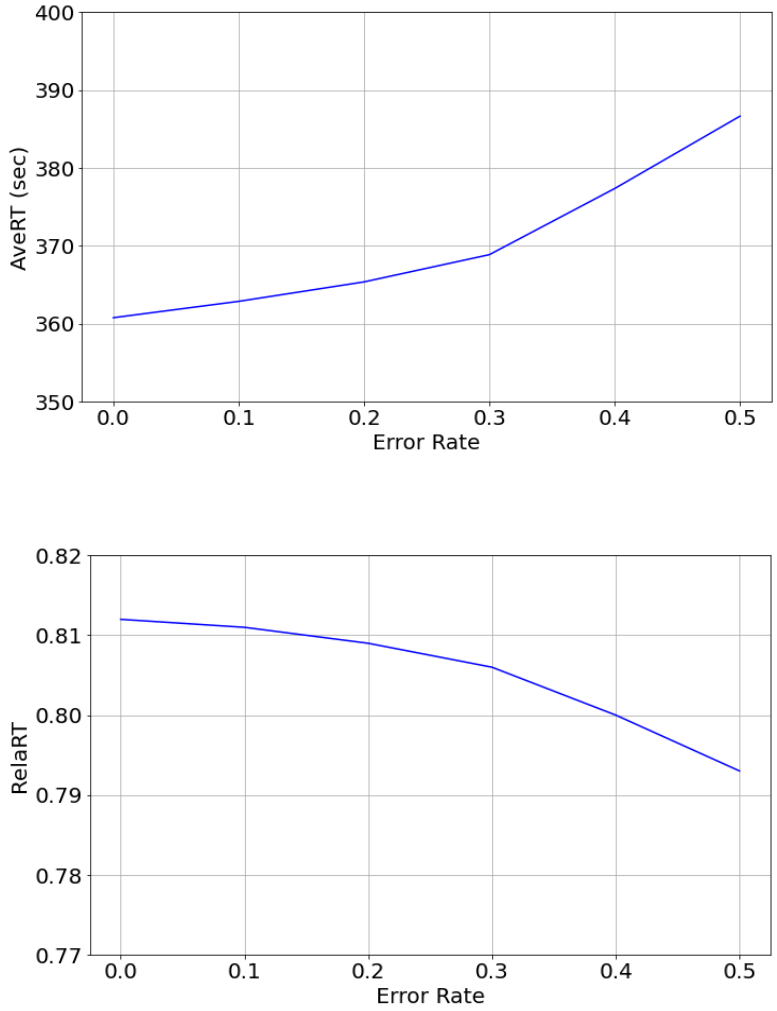


Figure 5.6. Robust to Traffic Conditions. I=70, m=1

for the respective I values performed.

From these findings, we observe that the score network trained for I=60 performs equally well when applied to EMS systems with different capacities. This suggests that a single score network trained under one setting can be effectively used for various EMS systems, even across different cities. This demonstrates the network's excellent transferability and its potential for broad applications in real-world scenarios.

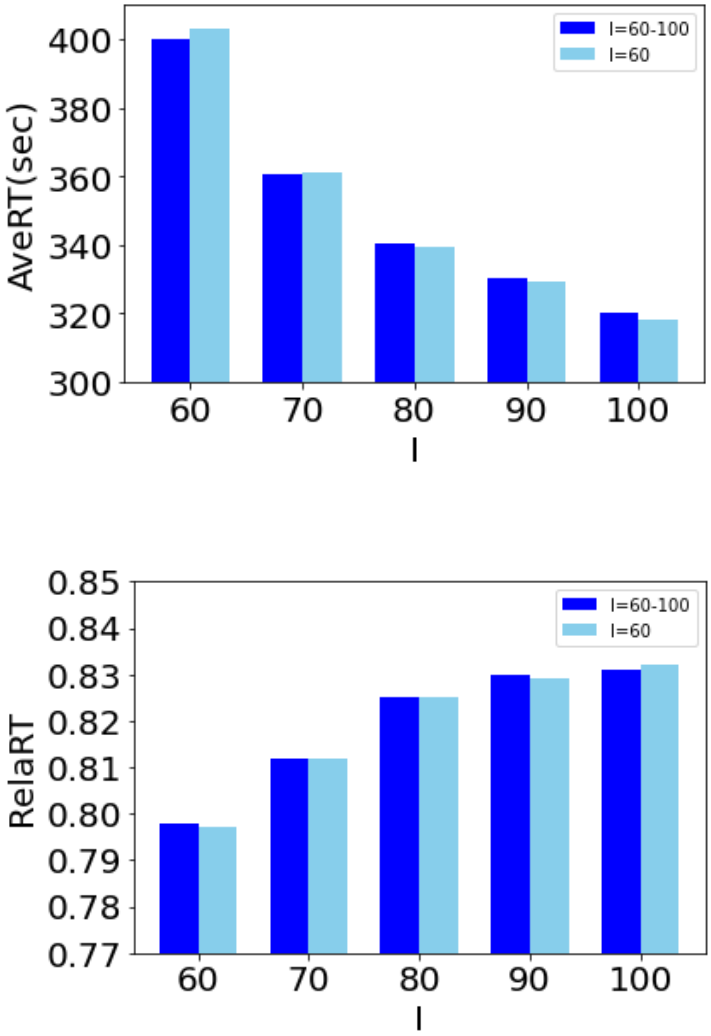


Figure 5.7. Robust to I's. The performance of the score network trained under 'I=60' is indicated by 'I=60', whereas the performance of the score networks learned under matching I's is indicated by 'I=60-100', where m=1.

5.8.3. Robust to Human Factors

We analyze cases in which the redeployment decisions made by our technique are ignored or delayed by ambulance crews. The disobedience probability $\text{prob}_{\text{disobedience}}$ is the chance that an available ambulance will be redeployed to a random station instead of the location recommended by our algorithm. Similarly, the possibility that an available ambulance is delayed before being redeployed to the station recommended by our technique is known as the delay probability $\text{prob}_{\text{delay}}$. Within 30 minutes, we assume that the duration of a delay is uniformly distributed.

Results from the experiment are displayed in Figure 5.8. Because ambulance workers are usually highly trained in emergency medical services, disobedience is rare, with a likelihood of less than 1/50. Under such conditions, as shown in Figure 5.8, the effect of sporadic disobedience on our protocol is negligible. Even when delays happen a little more frequently, their impact increases gradually as the likelihood of a delay rises. As a result, our approach shows robustness to human factors.

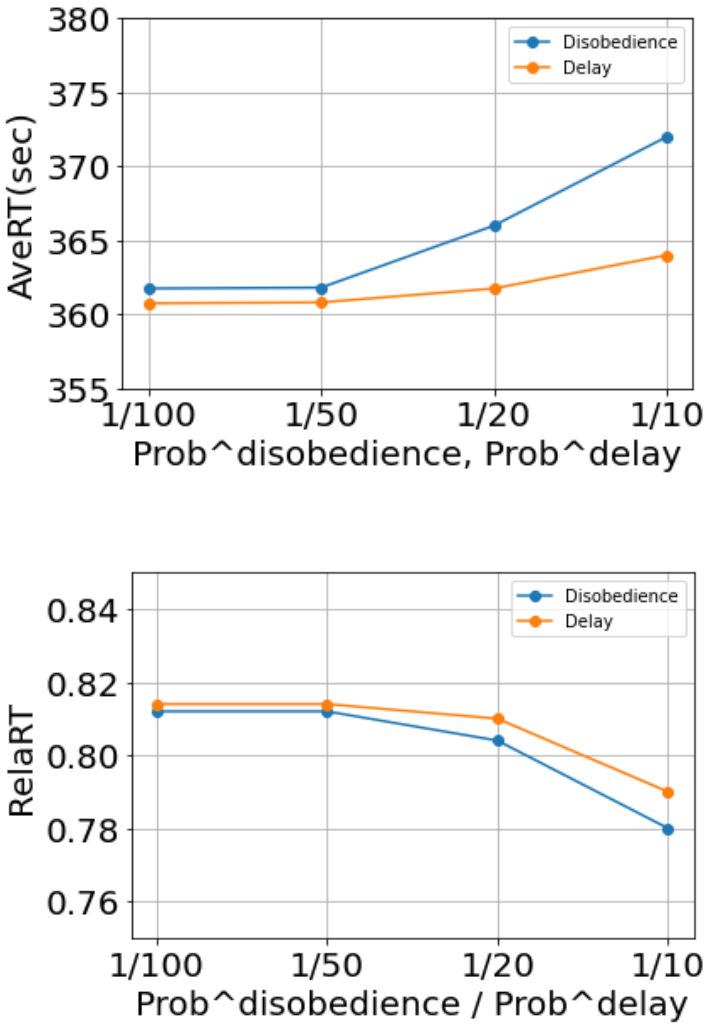


Figure 5.8. Robust to Human Factors. I=70, m=1

CHAPTER 6: DEEP SCORE NETWORK AND DYNAMIC CHARLIE VEHICLES REDEPLOYMENT ALGORITHM

In the realm of EMS, the efficient allocation of resources is paramount to ensuring timely and effective emergency response. While much attention has been rightfully devoted to the optimal redeployment of ambulances to spoke stations, a critical yet distinct challenge emerges in the form of managing Charlie vehicles. These specialized 4x4 response vehicles play a pivotal role in providing critical care to patients in challenging environments and scenarios.

Unlike ambulances stationed at fixed locations, the deployment of Charlie Vehicles is characterized by their dynamic nature, necessitating a nuanced approach to their management. The redeployment of Charlie Vehicles to regions with anticipated high demand presents a unique set of challenges and opportunities, distinct from traditional ambulance redeployment strategies.

This chapter delves into the complexities of dynamic redeployment for Charlie vehicles management, exploring this problem, while complementary to ambulance redeployment, requires tailored strategies to optimize response times and minimize idle driving costs. By addressing this challenge, EMS can enhance its capacity to respond effectively to critical emergencies, particularly in scenarios where the circumstances demand the specialized capabilities of Charlie vehicles.

6.1. Problem Definition

The deployment of Charlie vehicles is limited in Qatar due to their specialized nature and strategic positioning. While standard ambulances fulfill the crucial task of transporting patients, Charlie vehicles stand out as a specialized force, providing

advanced medical care where it is most urgently needed.

Our goal is to optimally direct a fleet of available Charlie vehicles to different regions in the country to minimize the response time to patients who require critical care and vehicles' idle driving costs.

Figure 6.1 illustrates the process of redeployment for Charlie vehicles in EMS. In Step 1, a patient calls 999, and the EMS hub determines that this is a critical care case requiring the dispatch of a Charlie vehicle. In Step 2, the hub dispatches a Charlie Vehicle from the nearest region to the patient's location. Step 3 involves the ambulance traveling to the patient's location, where paramedics provide advanced medical care. After treating the patient, in Step 4, the Charlie vehicle is redeployed to another region based on future demand and availability predictions.

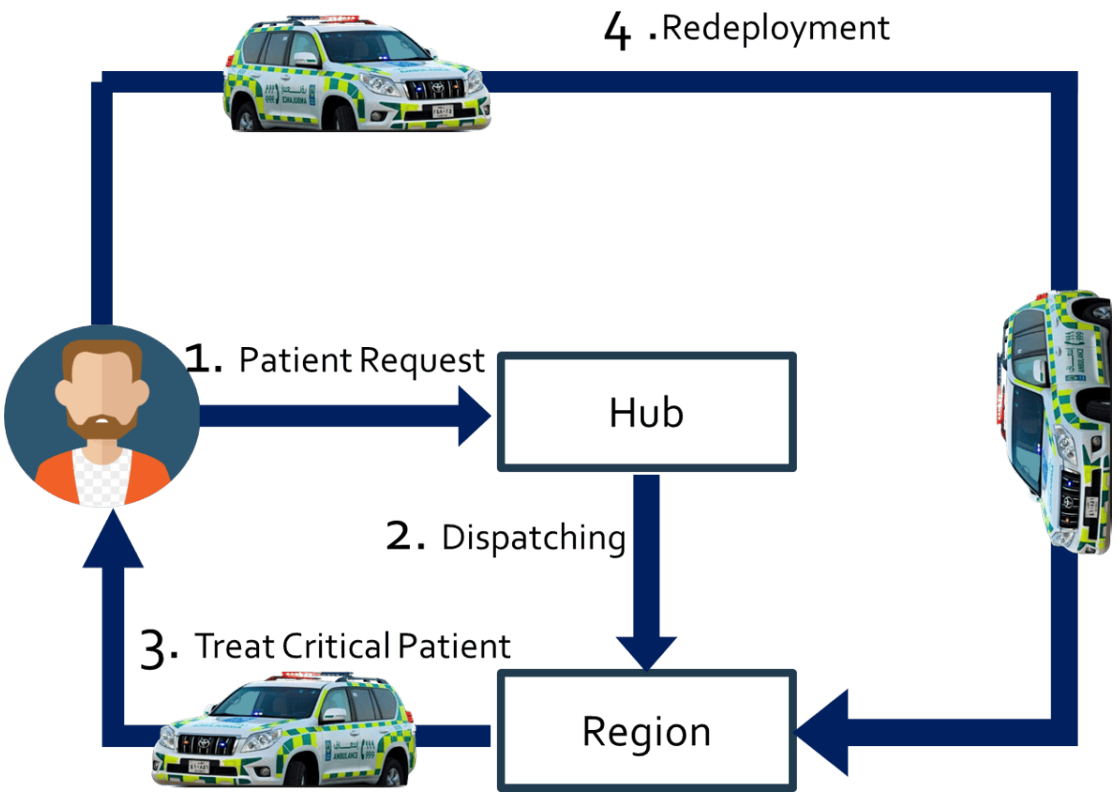


Figure 6.1. Charlie Vehicle Redeployment

We assume that the EMS hub tracks each Charlie vehicle's location (lat,long) and

availability status and all critical patient requests. We use this information to proactively redeploy available Charlie vehicles ($q^{'}$) to regions where it predicts the future demand and to match Charlie vehicles to incoming critical patient requests that require critical care.

The geographical service area is divided into M regions and considers K time slots of length Δk indexed by $k = k_0+, ..., k_0 + K$, where k_0 is the current time slot. The number of critical care patient requests at the $c - th$ region within time slot k is then denoted by v_{kc} and the number of available Charlie vehicles in this region at the beginning of time slot k is denoted by q_{kc}. The index of an individual available Charlie vehicle is denoted by $q^{'}$. We also define $q_{kk'c}$ as the number of vehicles that are occupied at time k but will become idle in the $c - th$ region in time slot k'.

To predict the future $q_{kk'c}$ given a set of redeployment actions, we use $F_k = (f_{k1}, ..., f_{kN})$, where N is the total number of Charlie vehicles, to denote the current location, occupied/idle status and destination of each Charlie vehicle at time k. By combining this data, we can predict $Q_{k:k+K} = (q_{k1}, ..., q_{(k+K)M})$, a matrix that gives the number of Charlie vehicles available at each region c from time k to time $k + K$, given the redeployment actions. Similarly, we define the future demand $V_{k:k+K} = (\overline{v}_{k1}, ..., \overline{v}_{(k+K)M})$.

The grid bellow (Table 6.1) illustrates an example of the state of idle Charlie vehicles in different regions over three time slots (e.g. k=1, k=2, k=3). Each row represents a region (e.g. A, B, C, D), and each column represents a time slot. The values in the grid represent the number of idle Charlie vehicles in each region at each time slot (q_{kc}). This grid helps visualize the dynamics of idle vehicles across regions and time, which is crucial for making decisions on redeployment actions to minimize

Table 6.1. Grid represent the number of idle Charlie vehicles in each region at each time slot (q_{kc})

c	k=1	k=2	k=3
A	$q_{1,A}$	$q_{2,A}$	$q_{3,A}$
B	$q_{1,B}$	$q_{2,B}$	$q_{3,B}$
C	$q_{1,C}$	$q_{2,C}$	$q_{3,C}$
D	$q_{1,D}$	$q_{2,D}$	$q_{3,D}$

response times and idle driving costs.

We assume the EMS consists of a EMS hub (dispatch center), a number of geographically distributed Charlie vehicles, and critical patients. Figure 6.2 illustrates this framework.

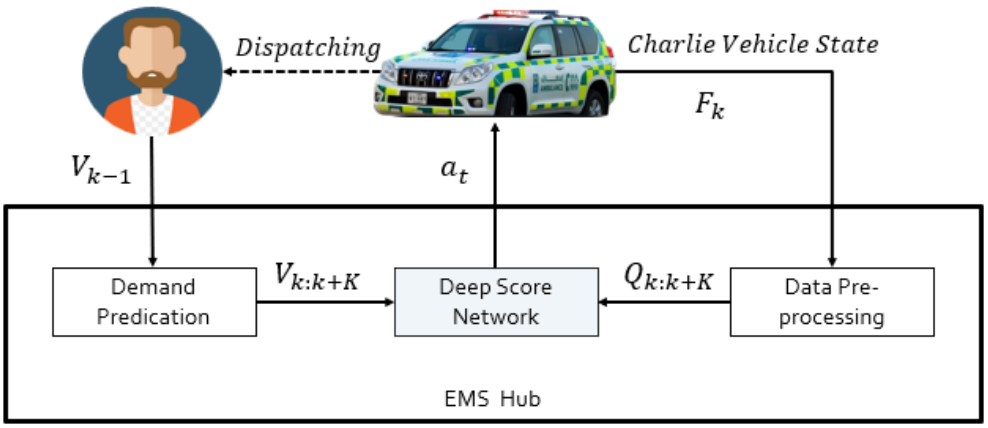


Figure 6.2. Interaction of Charlie Vehicles and Critical Patients with the EMS Hub

The notations used in this section are summarized in Table 6.2.

Table 6.2. Description of Parameters for Dynamic Charlie Vehicle Redeployment

Parameters	**Description**
N	Number of Charlie vehicles
M	Number of regions
c	Index of a region
$\gamma \in (0, 1]$	Time discount rate
Δk	Step size
K	Maximum time steps
f_{kn}	n-th Charlie vehicles state at the beginning of time slot k
$q^{'}$	Index of an idle Charlie vehicle
q_k	Number of idle Charlie vehicles in each region at time slot k
$q_{kk'}$	Number of occupied Charlie vehicles at time k that become idle at time k'
v_k	Number of critical requests in each region at time slot k
$\overline{v}_k$	Number of predicted critical requests in each region at time slot k
u_k	Number of Charlie vehicles to be redeployed between regions at time slot k
τ_k	Expected travel time between the regions at time slot k
$\beta_1, \beta_2, \beta_3$	Weighing factors

Similar to our dynamic score network explained in Section 4.2, Figure 6.3 illustrates the deep score network for Charlie vehicle redeployment. The inputs are represented by e_c, which stands for the current factors for each region c. These dynamic factors impact the decision to redeploy the available Charlie vehicle. The result is reg_c, which is the score of the region c. The model learns a scoring function, $reg_c = f(e_c; \theta)$. The parameters of the neural network layers are donated as θ.

In Section 6.3, a DRL framework is used to learn the weights θ in the score network, where the weights are shared among all spoke stations.

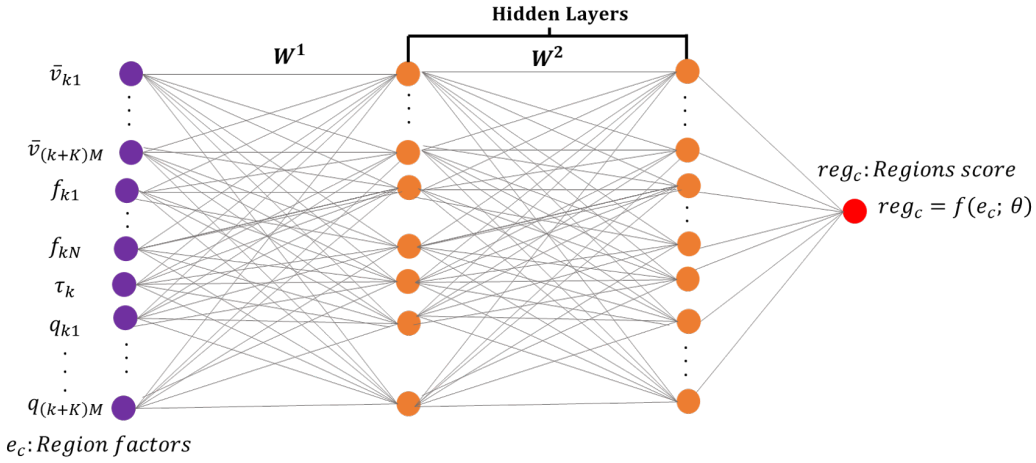


Figure 6.3. Deep Score Network.

As illustrated in Figure 6.3, the factors e_c are denoted as:

$$e_c = (V_{k:k+K}, F_k, \tau_k, Q_{k:k+K}) \tag{6.1}$$

6.2.1. Dynamic Factors

In the context of dynamic redeployment for Charlie vehicles management, several key dynamic factors influence the decision-making process. These factors are crucial for

optimizing the deployment of Charlie vehicles to minimize response times for critical care incidents and reduce idle driving costs. The dynamic factors considered in this framework are:

Factor 1. $V_{k:k+K} = (\overline{v}_{k1}, ..., \overline{v}_{(k+K)M})$ represents the demand at which critical patient requests arrive in the future at each region c. This factor represents the anticipated number of critical care incidents expected to occur at each region c in the upcoming time period. It is essential for predicting the demand for critical care services and optimizing the deployment of Charlie vehicles to respond effectively to emergencies in each region. This was found using RF regression as done before in Factor 1. Section 4.2.1.

Factor 2. $F_k = (f_{k1}, ..., f_{kN})$ denotes the state of each Charlie vehicle, including its current location, idle or occupied status, and destination, is another important dynamic factor. where N is the number of Charlie vehicles. This information, provides real-time data on the availability and positioning of Charlie vehicles. It helps in determining the optimal redeployment actions to efficiently utilize the vehicles' resources.

Factor 3. τ_k represents the expected travel time between two regions at each time slot k,which is a critical factor influencing the decision to redeploy Charlie vehicles. This factor considers various factors such as road conditions, traffic congestion, and distance between regions. It helps in estimating the time required for a Charlie vehicle to reach its destination, enabling proactive redeployment to regions with anticipated high demand.

Factor 4. $Q_{k:k+K} = (q_{k1}, ..., q_{(k+K)M})$ denotes the predicted availability of vehicles to help in forecasting future availability and optimizing redeployment actions to meet anticipated demand efficiently. Predicting the future availability of Charlie vehicles at each region c is essential for proactive redeployment. This prediction is

based on current vehicle availability, redeployment actions, and expected travel times, ensuring that vehicles are strategically positioned to respond to critical incidents in a timely manner.

6.3. Reinforcement Learning Deep Score Network

The dynamic Charlie vehicle redeployment problem can be formulated as an RL task, which involves five main concepts: state, action, transition, reward, and policy. These concepts are fundamental in understanding and solving the Charlie vehicle redeployment problem using RL. We present each of these concepts in the context of the dynamic redeployment challenge for Charlie vehicles in this section.

At each time step k, the agent receives some representation of the environments state s_t and reward r_t. It then takes action a_t to redeploy Charlie vehicles to the different regions so as to maximize the expected future reward:

$$\sum_{k'=k}^{\infty} \gamma^{k'-k} r_{k'}(a_t, s_t) \tag{6.2}$$

where $\gamma < 1$ represents a time discount rate.

To define r_t, we wish to minimize two performance criteria: the response time and the idle driving costs.

6.3.1. Reinforcement Learning Framework

State. At a new time step, we can characterize the system status as s_t. This state s_t should contain all the data needed to deploy the Charlie vehicle that is currently on hand again. Consequently, the factors of every region are included in s_t.

$$s_t = (e_1, e_2, \ldots, e_M) \tag{6.3}$$

where M denotes the maximum number of regions and e_c refers to the factors of the region c.

Action. The action in the Charlie vehicle redeployment problem is the same as choosing a region to which the available Charlie vehicle $q^{'}$ will be sent at this time. Consequently, the current operation, indicated by a_t,

$$a_t \in \{1, 2, \ldots, M\} \tag{6.4}$$

where M is the maximum number of regions.

Transition. Until the next time step, no further action is conducted after taking action a_t in the current station s_t. The following state s_{t+1} is entered by the system when the next time step starts.

Reward. In the context of the dynamic Charlie vehicle redeployment problem, the reward function plays a crucial role in guiding the learning process of the RL algorithm. The goal of the reward function is to quantify the desirability of different states and actions, providing the agent with feedback on its decisions. In this specific formulation, the reward is designed to achieve several objectives.

$$r_t(s_t, a_t) = -\beta_1 \sum_{c=1}^{M} \text{RT}_c - \beta_2 \sum_{c,w=1}^{M} \tau_{k,cw} \cdot u_{k,cw} - \beta_3 \sum_{c=1}^{M} (q_{kc} - \overline{v}_{kc})^2 \tag{6.5}$$

The first part of the reward function in Equation 6.5, $-\beta_1 \sum_{c=1}^{M} \text{RT}_c$ represents the total response time of critical patients in all regions. The sum is taken over all

regions c from 1 to M (the total number of regions). RT_c is the response time of critical patients in region c for an available Charlie vehicle q'. The negative sign indicates that shorter response times result in a higher reward, aligning with the goal of minimizing response times.

The second part of the reward function, $-\beta_2 \sum_{c,w=1}^{M} \tau_{k,cw} \cdot u_{k,cw}$, represents the cost associated with redeploying Charlie vehicles between regions. The sum is taken over all pairs of regions c and w from 1 to M. $\tau_{k,cw}$ is the expected travel time from region $c-th$ to region $w-th$. $u_{k,cw}$ is the number of Charlie vehicles redeployed from region c to region w at time step k. The term is weighted by β_2, which allows to balance the importance of minimizing critical response times versus minimizing idle driving costs.

In optimizing the deployment of Charlie vehicles in EMS, a key consideration is balancing the availability of these specialized vehicles with the demand for critical care services. To address this, we propose enhancing the reward function with a dynamic component that adjusts based on the current availability and expected demand for critical care.

This dynamic component penalizes the presence of idle vehicles when the demand for critical care services is high and rewards having idle vehicles when the demand is low. It is defined as the third part, $-\beta_3 \sum_{c=1}^{M} (q_{kc} - \overline{v}_{kc})^2$, where β_3 is a tuning parameter that controls the impact of this component on the overall reward. This term penalizes the squared difference between the number of idle Charlie vehicles and the expected number of critical care requests in each region. If there are more idle vehicles than expected requests, this term decreases the reward, discouraging idle vehicles. Conversely, if there are fewer idle vehicles than expected requests, this term increases the

reward, encouraging idle vehicles.

By incorporating this dynamic component into the reward function, the optimization process becomes more adaptive to the current demand for critical care, ensuring that Charlie Vehicles are deployed efficiently in response to changing conditions. Adjusting the parameter β_3 allows for fine-tuning the balance between supply and demand for critical care services.

To find $(q_{kc} - \overline{v}_{kc})$, we find the future number of available charlie vehicles: The number of idle charlie vehicles in each time slot is:

$$q_{k+1,c} = \max(q_{kc} - \overline{v}_{k+1,c}) - \sum_{w=1}^{M}(u_{k,cw} - u_{k,wc}) + q_{kk'c} \tag{6.6}$$

Here the first term corresponds to "leftover" Charlie vehicles from time slot k, and the second term to the net number of idle Charlie vehicles redeployed to region c at time k, i.e., right before the start of time slot $k+1$.By subtracting the vehicles arriving in region c from the vehicles leaving region c, we get the net change in the number of idle vehicles in region c.

The last term represents the Charlie vehicles that come into region c at time k: the term $q_{kk'c}$ corresponds to occupied Charlie vehicles at time k that will be available in time slot k'.

Policy. In the context of Charlie vehicle redeployment, the policy remains similar to the ambulance redeployment policy described above in Section 4.3. Here, the action of redeploying a Charlie vehicle is determined based on the current states of the system, represented by s_t, using the policy $\pi_\theta(s_t, a_t)$. The policy network, similar to Figure 4.4, is utilized to select the optimal action for each Charlie vehicle deployment location.

6.3.2. Learning θ With Policy Gradient

In the context of learning the optimal policy for Charlie vehicle redeployment, the process follows a similar approach as described above for ambulance redeployment in Section 4.3.2. The objective remains to maximize the expected long-term discounted reward by training the policy network represented by the optimal weights θ. The objective function is defined as:

$$\max_{\theta} J(\theta) = \mathbb{E}_{s\sim\pi_\theta}[v(s)] \tag{6.7}$$

where $v(s)$ represents the expected long-term discounted reward starting from state s and following policy π_θ. The discounted reward $v(s)$ is calculated similarly to Equation 4.8.

The gradient of $J(\theta)$ with respect to θ is obtained using policy gradient methods, similar to Equation 4.11, and is used to update θ as shown in Equation 4.12. The goal of this process is to learn an optimal policy for Charlie vehicle redeployment that minimizes response times and idle costs by updating the policy network's weights θ.

6.4. Dynamic Redeployment Algorithm

For the dynamic redeployment algorithm for Charlie vehicles, we can adapt the same proposed approach used for ambulances in Section 4.4. Similar to the ambulance redeployment algorithm, the goal is to efficiently redeploy available Charlie vehicles, $q^{'}$ based on their scores, minimizing response times and idle costs. The algorithm is activated whenever a new time step starts.

Algorithm 3, the dynamic redeployment algorithm for Charlie vehicles aims to

efficiently assign each available Charlie vehicle to a deployment region, c, based on its score, reg_c, computed using the scoring network. The algorithm is activated at the start of each time step.

Firstly, the algorithm acquires the current factors, e_c, for each potential deployment region. Next, it computes the score, reg_c, for each deployment region using the scoring network: $reg_c = f(e_c; \theta)$. The algorithm then sorts the deployment regions in descending order of their scores: $c_1, c_2, \ldots, c_M$, where M is the total number of deployment regions.

For each available Charlie vehicle $q^{'}$, the algorithm iterates through the sorted list of deployment regions. If a region c_j has not yet been assigned an ambulance, it assigns the closest available ambulance q' to region c_j, marks region c_j as assigned, and moves to the next available Charlie vehicle.

This process ensures that each Charlie vehicle is redeployed to a deployment region that has not yet been assigned an ambulance, based on the scores of the regions, thus minimizing response times and idle costs for the fleet of available ambulances.

This algorithm is designed to efficiently redeploy a fleet of available Charlie vehicles based on the learned policy network, ensuring that each vehicle is sent to the deployment location that offers the greatest potential impact in reducing response times and improving patient outcomes.

We utilize the DRL algorithm (Algorithm 3) to train the deep scoring network using the EMS requests from the preceding 31 days, from January 1st 2022 to February 1st 2022. The remaining 20 days (February 1–21, 2022) are used as test data to evaluate the charlie vehicle redeployment algorithm (Algorithm 2) based on the learned deep score network, as illustrated in Figure 6.4.

Algorithm 3 Dynamic Charlie Vehicle Redeployment Algorithm

procedure MATCHCHARLIE
    Acquire current factors e_c for each deployment region c
    Compute score $reg_c = f(e_c; \theta)$
    Sort deployment regions in descending order of scores: $c_1, c_2, \ldots, c_M$
    $assigned \leftarrow 0$ ▷ Number of assigned ambulances
    for q' from 1 to q **do** ▷ For each available Charlie vehicle
        Find the closest deployment region c_j from $c_1, c_2, \ldots, c_M$
        Assign ambulance q' to region c_j
        Mark region c_j as assigned
        $assigned \leftarrow assigned + 1$
        if $assigned = q$ **then**
            break ▷ All available ambulances are assigned, exit loop
        end if
    end for
end procedure

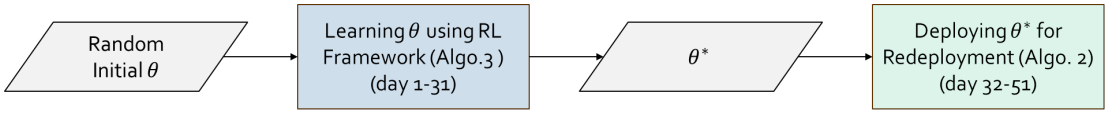


Figure 6.4. Relationship between DRL Algorithm 3 and Charlie vehicle redeployment Algorithm 2

CHAPTER 7: EVALUATION OF PROPOSED DYNAMIC CHARLIE VEHICLES REDEPLOYMENT ALGORITHM

Similar to Chapter 5, we evaluate our dynamic Charlie vehicle redeployment approach by comparing the performance with several baseline redeployment strategies, explained in the literature review in Section 2.3, including MCLP, RA,NS,LS,ERTM,MEXCLP and DMEXCLP approaches.

7.1. Performance Metrics

In evaluating the effectiveness of the proposed dynamic Charlie vehicle redeployment approach compared to the baseline methods, two standard performance metrics are used. The first metric is the average response time (AveRT) in seconds for all critical patients. As explained in Section 5.1 it is calculated as the average of the response times of individual critical patient requests, given by Equation 5.1. A lower AveRT indicates better performance of the redeployment method.

The second metric is the ratio of patients treated within a specified time threshold, denoted as RelaRT (relative response time). Unlike traditional ambulance redeployment, where patients are picked up, the Charlie vehicle approach focuses on treating patients on-site. Therefore, RelaRT measures the proportion of critical patient requests with response times less than or equal to the threshold RT_{t*} (In our experiments, the threshold used was 10 minutes), as shown in Equation 7.1. A higher RelaRT indicates that more patients are being treated within the specified time threshold, which is desirable for the Charlie vehicle redeployment approach.

$$\text{RelaRT} = \frac{1}{P}\sum_{p=1}^{P} ||RT_{tp} \leq RT_{t*}|| \tag{7.1}$$

Table 7.1 compares the performance of various baseline methods and our proposed dynamic redeployment approach in terms of AveRT and RelaRT at different Charlie vehicle numbers (N) in an EMS system. Our method consistently outperforms all baseline methods across different scenarios.

For example, at $N = 20$, our method achieves an AveRT of 470.10 sec and a RelaRT of 0.350, while the best-performing baseline method, ERTM, achieves an AveRT of 481.96 sec and a RelaRT of 0.347. This results in a 2.46% reduction in AveRT and a 0.86% increase in RelaRT for our method compared to ERTM. Similar trends are observed for other baseline methods, with our approach consistently achieving lower AveRT and slightly higher RelaRT values, indicating its effectiveness in improving emergency response times and number of patients treated.

Furthermore, comparing our method to the actual performance observed in real-world data reveals significant improvements. At $N = 20$, our method achieves a notable 21.11% reduction in AveRT and a 20.21% increase in RelaRT compared to the actual performance. This trend persists across all scenarios, demonstrating the robustness and effectiveness of our approach in enhancing emergency medical service efficiency and response times.

Overall, our proposed dynamic redeployment of Charlie vehicles shows significant improvements in AveRT and RelaRT compared to baseline methods and actual performance, highlighting its effectiveness in optimizing Charlie vehicle deployment strategies in EMS. The consistent superiority of our approach across different scenarios underscores its potential to enhance EMS and improve patient outcomes in real-world EMS operations.

Table 7.1. Comparisons with Baseline Methods for Dynamic Charlie Vehicle Redeployment

Methods	N = 10		N = 20		N = 30		N = 40		N = 50	
	AveRT	RelaRT	AveRT	RelaRT	AveRT	RelaRT	AveRT	RelaRT	AveRT	RelaRT
MCLP	625.24	0.238	588.31	0.252	566.79	0.265	541.42	0.279	526.80	0.306
RA	809.17	0.244	749.43	0.268	726.84	0.281	709.25	0.294	694.62	0.311
NS	778.72	0.231	774.49	0.275	758.13	0.283	752.71	0.281	739.83	0.301
LS	615.31	0.253	541.92	0.283	491.69	0.289	454.30	0.297	429.79	0.317
ERTM	495.60	0.339	481.96	0.347	477.22	0.348	476.89	0.331	471.23	0.483
MEXCLP	498.75	0.330	486.87	0.305	486.31	0.298	477.56	0.309	474.90	0.378
DMEXLP	511.42	0.312	493.70	0.318	492.48	0.325	479.13	0.327	466.31	0.380
Actual	614.34	0.268	597.21	0.269	580.67	0.273	564.72	0.279	549.32	0.320
Our Proposed Method	484.95	0.345	470.10	0.350	455.87	0.346	442.22	0.344	429.14	0.497

7.3. Convergence of Training

Figure 7.1 shows our scoring function's performance throughout training. It display's that the training is convergent and that the scoring function can quickly converge to a nearly optimal solution, similar to the results in Section 5.4

7.4. Necessity of Considering All Factors

This section examines the necessity of including the dynamic factors of every region into our Charlie vehicle redeployment method. This was achieved using the same approach as in Section 5.5, however using the dynamic region factors listed in Section 6.2.1.

As we can see in Figure 7.2, using all four factors results in the best performance, where AveRT is the shortest and RelaRT is the highest compared to the other

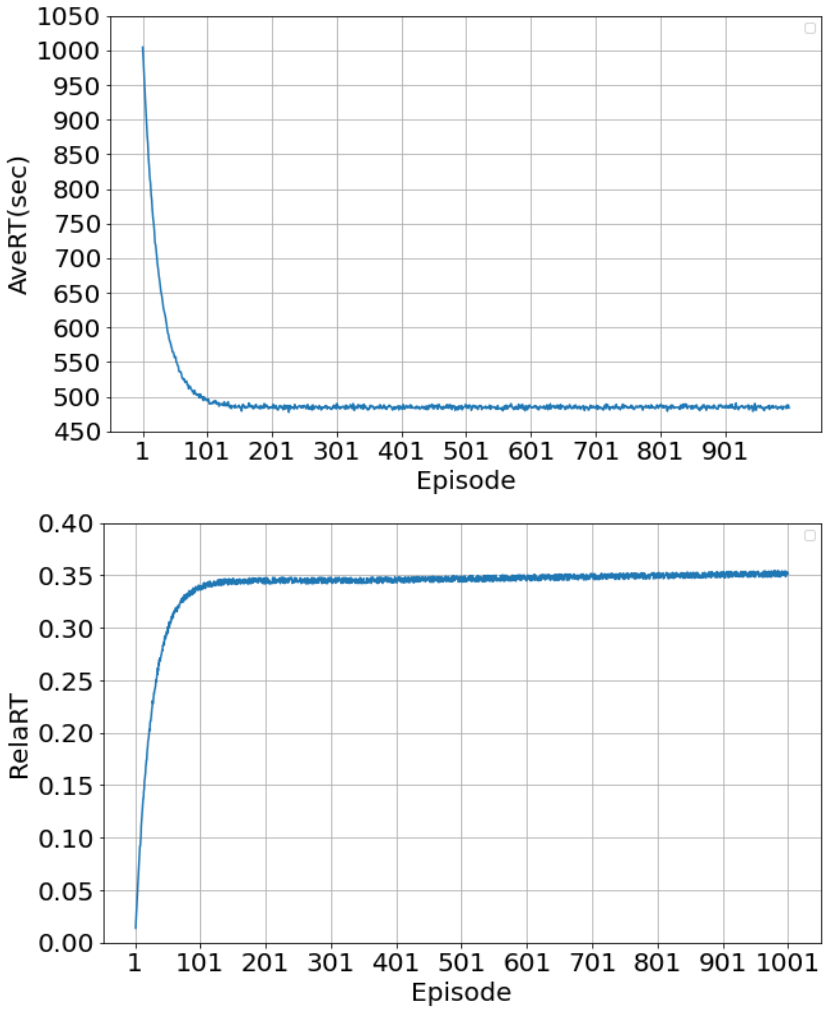


Figure 7.1. Convergence of Training. N=10

combinations of factors.

In addition, we analyze the significance of each factor in Figure 7.3. As illustrated in the figure, factor 4 is the most important, followed by factor 1. This is consistent with our initiation that the two most important factors in determining whether an available Charlie vehicle should be redeployed to a region are the number of available Charlie vehicles in the region and the future demand at the region.

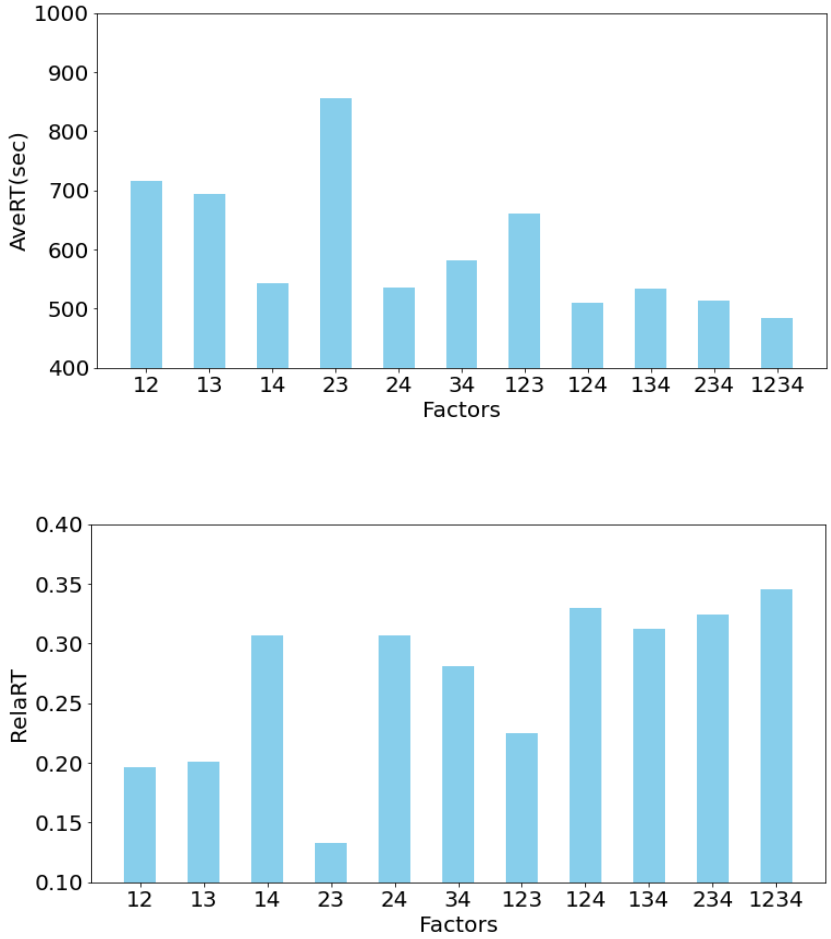


Figure 7.2. Performance of Proposed Method Considering Different Factors. N=10

7.5. Influence of Number of Critical Patient Requests

Similar to Section 5.7, we examine the performance of our proposed approach in terms of the number of critical patient requests.

In Figure 7.4, we can see that during periods of high critical patient volume, the Ratio_AveRT is elevated, suggesting that many patients are experiencing longer response times due to Charlie vehicles not being dispatched from the nearest region. Similarly, during periods of increased critical patient numbers, the Ratio_RelaRT decreases, indicating that fewer patients are being treated within the critical 10-minute window. This

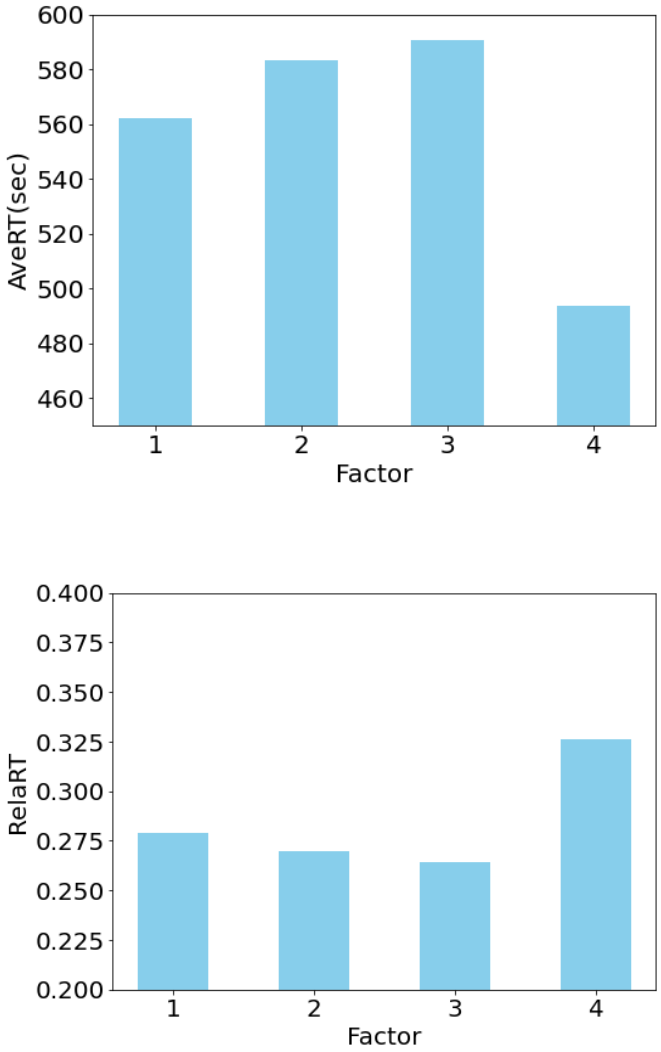


Figure 7.3. Significance of Each Factor. N=10

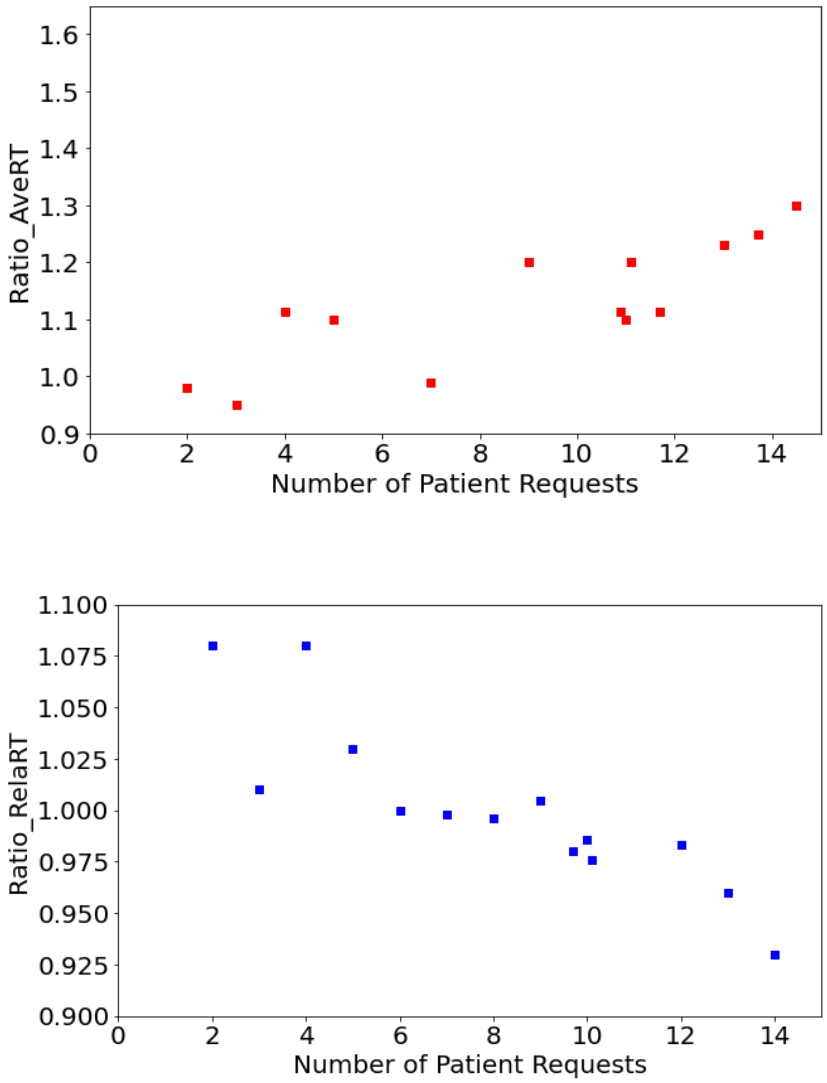


Figure 7.4. Influence of Number of Patient Requests to Our Proposed Method. N=10

highlights a pressing need for additional Charlie vehicles during these peak periods.

CHAPTER 8: CONCLUSION AND FUTURE WORK

8.1. Conclusion

This thesis introduces a novel approach to dynamically redeploy ambulances. The method involves four main steps: proposing a deep score network, training it with a DRL algorithm, implementing an effective dynamic ambulance redeployment algorithm and finally implementing a dynamic Charlie vehicle redeployment algorithm. This approach significantly reduces patient response times. Compared to standard methods, our approach reduces average response times by approximately 20% (100 seconds) and increases the ratio of patients being picked up within 10 minutes from 0.648 to 0.798. As for Charlie vehicle redeployment, our proposed approach reduced average response times by around 13.3% (125 seconds) and increased the percentage of critical patients treated within 10 minutes by approximately 11.08%. This demonstrates that our method can enhance the efficiency of EMS systems, thereby improving their ability to save lives during emergencies.

8.2. Future Work

We want to use the deep score network to solve additional sequential decision-making issues in the future, like express carrier, food carrier, and taxi dispatching. We think that the idea of deep score network learning can also be useful in these kinds of situations. Furthermore, although we have focused on the redeployment of ambulances, we intend to broaden the scope of our research to encompass the combined dispatching and redeployment of ambulances through the use of deep scoring networks. This cooperative method can be seen as a RL issue with several tasks. We will need to think about a few important issues in order to answer this. First, two scoring networks—one

for the redeployment task and another for the dispatching task—will be needed. Second, in order to allow both scoring networks to be learned simultaneously, we will need to use multi-task RL techniques. In order to ensure optimal performance, the elements influencing the dispatching task will need to be carefully considered.

REFERENCES

[1] G. N. Berlin and J. C. Liebman, "Mathematical analysis of emergency ambulance location," *Socio-Economic Planning Sciences*, vol. 8, no. 6, pp. 323–328, 1974, ISSN: 0038-0121. DOI: https://doi.org/10.1016/0038-0121(74)90036-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0038012174900366.

[2] P. T. Pons, J. S. Haukoos, W. Bludworth, T. Cribley, K. A. Pons, and V. J. Markovchick, "Paramedic response time: Does it affect patient survival?" *Academic Emergency Medicine*, vol. 12, no. 7, pp. 594–600, 2005. DOI: https://doi.org/10.1197/j.aem.2005.02.013. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1197/j.aem.2005.02.013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1197/j.aem.2005.02.013.

[3] T. Blackwell and J. Kaufman, "Response time effectiveness:comparison of response time and survival in an urban emergency medical services system," *Academic emergency medicine : official journal of the Society for Academic Emergency Medicine*, vol. 9, pp. 288–95, May 2002. DOI: 10.1111/j.1553-2712.2002.tb01321.x.

[4] "Qatar population," *Worldmeter*, 2022.

[5] F. Gotting, "Healing hands of qatar," 2006.

[6] P. Wilson, G. Alinier, T. Reimann, and B. Morris, "Influential factors on urban and rural response times for emergency ambulances in qatar," *Mediterranean Journal of Emergency Medicine*, 2017.

[7] iloveqatar.net, "A guide to hmc's ambulance service in qatar," Feb. 2020, Posted On: 25 February 2020 04:45 pm, Updated On: 12 November 2020 10:18 am.

[8] V. Schmid, "Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming," *European Journal of Operational Research*, vol. 219, no. 3, pp. 611–621, Jun. 2012, ISSN: 0377-2217. DOI: 10.1016/j.ejor.2011.10.043.

[9] W. Steenbeek, *Geographic distance and road distance are highly correlated (in the four largest municipalities of the netherlands)*, https://www.woutersteenbeek.nl/post/geometric-road-distance-g4, Sep. 2020.

[10] A. H. Wong and T. J. Kwon, "Advances in regression kriging-based methods for estimating statewide winter weather collisions: An empirical investigation," *Future Transportation*, vol. 1, no. 3, pp. 570–589, Oct. 2021, ISSN: 2673-7590. DOI: 10.3390/futuretransp1030030.

[11] A. A. Nasrollahzadeh, A. Khademi, and M. E. Mayorga, "Real-time ambulance dispatching and relocation," *Manufacturing & Service Operations Management*, Apr. 2018. [Online]. Available: https://pubsonline.informs.org/doi/epdf/10.1287/msom.2017.0649.

[12] A. Mukhopadhyay, G. Pettet, C. Samal, A. Dubey, and Y. Vorobeychik, "An online decision-theoretic pipeline for responder dispatch," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, ACM, Apr. 2019. DOI: 10.1145/3302509.3311055.

[13] S. S. Mohri and H. Haghshenas, "An ambulance location problem for covering inherently rare and random road crashes," *Computers & Industrial Engineering*, vol. 151, p. 106 937, 2021.

[14] C. Jagtenberg and A. Mason, "Fairness in the ambulance location problem: Maximizing the bernoulli-nash social welfare," *Available at SSRN 3536707*, 2020.

[15] A. Ismail, "Making sense of a barrier: Us news discourses on israel's dividing wall," *Journal of Communication Inquiry*, vol. 34, no. 1, pp. 85–108, 2010.

[16] N. J. Vianen, I. M. Maissan, D. den Hartog, *et al.*, "Opportunities and barriers for prehospital emergency medical services research in the netherlands; results of a mixed-methods consensus study," *European Journal of Trauma and Emergency Surgery*, pp. 1–12, 2023.

[17] L. Aboueljinane and Y. Frichi, "A simulation optimization approach to investigate resource planning and coordination mechanisms in emergency systems," *Simulation Modelling Practice and Theory*, vol. 119, p. 102 586, 2022.

[18] N. Pulsiri, R. Vatananan-Thesenvitz, T. Sirisamutr, and P. Wachiradilok, "Save lives: A review of ambulance technologies in pre-hospital emergency medical services," in *2019 Portland International Conference on Management of Engineering and Technology (PICMET)*, IEEE, 2019, pp. 1–10.

[19] V. Bélanger, A. Ruiz, and P. Soriano, "Recent optimization models and trends in location, relocation, and dispatching of emergency medical vehicles," *European Journal of Operational Research*, vol. 272, no. 1, pp. 1–23, Jan. 2019, ISSN: 0377-2217. DOI: 10.1016/j.ejor.2018.02.055.

[20] D. Neira, J. Escobar, and S. McClean, "Ambulances deployment problems: Categorization, evolution and dynamic problems review," *International Journal of Geo-Information*, vol. 11, no. 2, pp. 1–37, Feb. 2022, ISSN: 2220-9964. DOI: 10.3390/ijgi11020109.

[21] A. Mukhopadhyay, G. Pettet, S. Vazirizade, *et al.*, "A review of incident prediction, resource allocation, and dispatch models for emergency management," *arXiv*, Jun. 2020. DOI: 10.48550/arXiv.2006.04200. eprint: 2006.04200.

[22] L. A. McLay and M. E. Mayorga, "A dispatching model for server-to-customer systems that balances efficiency and equity," *Manufacturing & Service Operations Management*, Dec. 2012. [Online]. Available: https://pubsonline.informs.org/doi/abs/10.1287/msom.1120.0411.

[23] L. A. Albert, "A mixed-integer programming model for identifying intuitive ambulance dispatching policies," *J. Oper. Res. Soc.*, pp. 1–12, Nov. 2022, ISSN: 0160-5682. DOI: 10.1080/01605682.2022.2139646.

[24] X. Li and C. Saydam, "Balancing ambulance crew workloads via a tiered dispatch policy," *Pesqui. Oper.*, vol. 36, pp. 399–419, Sep. 2016, ISSN: 0101-7438. DOI: 10.1590/0101-7438.2016.036.03.0399.

[25] D. Bandara, M. E. Mayorga, and L. A. McLay, "Optimal dispatching strategies for emergency vehicles to increase patient survivability," *Int. J. of Operational Research*, vol. 15, no. 2, pp. 195–214, Aug. 2012, ISSN: 1745-7645. DOI: 10.1504/IJOR.2012.048867.

[26] S. K. Keneally, M. J. Robbins, and B. J. Lunday, "A markov decision process model for the optimal dispatch of military medical evacuation assets," *Health*

Care Manag. Sci., vol. 19, no. 2, pp. 111–129, Jun. 2016, ISSN: 1572-9389. DOI: 10.1007/s10729-014-9297-8.

[27] K. Liu, X. Li, C. C. Zou, H. Huang, and Y. Fu, "Ambulance dispatch via deep reinforcement learning," *ResearchGate*, pp. 123–126, Nov. 2020. DOI: 10.1145/3397536.3422204.

[28] A. Mukhopadhyay, G. Pettet, C. Samal, A. Dubey, and Y. Vorobeychik, "An online decision-theoretic pipeline for responder dispatch," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, ACM, Apr. 2019. DOI: 10.1145/3302509.3311055. [Online]. Available: https://doi.org/10.1145%5C%2F3302509.3311055.

[29] A. H. Hermansen, "Machine learning for spatio-temporal forecasting of ambulance demand: A norwegian case study," M.S. thesis, NTNU, 2021. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2833870?locale-attribute=no.

[30] E. Van De Weijer and O. A. Owren, "Forecasting ambulance demand in oslo and akershus," M.S. thesis, NTNU, 2022. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3030248.

[31] M. A. Mahmood, J. E. Thornes, F. D. Pope, P. A. Fisher, and S. Vardoulakis, "Impact of air temperature on london ambulance call-out incidents and response times," *Climate*, vol. 5, no. 3, p. 61, Aug. 2017, ISSN: 2225-1154. DOI: 10.3390/cli5030061.

[32] F. Mannering, "Temporal instability and the analysis of highway accident data," *Analytic Methods in Accident Research*, vol. 17, pp. 1–13, Mar. 2018, ISSN: 2213-6657. DOI: 10.1016/j.amar.2017.10.002.

[33] M. E. Schjølberg and N. I. P. Bekkevold, "Simulation and optimization of emergency medical services in oslo and akershus," NTNU, 2022. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3019906.

[34] R. J. McCormack and G. Coates, "A simulation model to enable the optimization of ambulance fleet allocation and base station location for increased patient survival," *Elsevier*, May 2015. [Online]. Available: https://dro.dur.ac.uk/15555.

[35] E. Erkut, A. Ingolfsson, and G. Erdogan, "Ambulance deployment for maximum survival," *ResearchGate*, Oct. 2010. [Online]. Available: https://www.researchgate.net/publication/255579100_Ambulance_Deployment_for_Maximum_Survival.

[36] C. J. Jagtenberg, S. Bhulai, and R. D. van der Mei, "An efficient heuristic for real-time ambulance redeployment," *Operations Research for Health Care*, vol. 4, pp. 27–35, 2015.

[37] R. Church and C. ReVelle, "The maximal covering location problem," in *Papers of the regional science association*, Springer-Verlag Berlin/Heidelberg, vol. 32, 1974, pp. 101–118.

[38] L. Snyder and M. Daskin, "Reliability models for facility location: The expected failure cost case," *Transportation Science*, vol. 39, pp. 400–416, Aug. 2005. DOI: 10.1287/trsc.1040.0107.

[39] M. Daskin, "A maximum expected covering location model: Formulation, properties and heuristic solution," *Transportation Science*, vol. 17, pp. 48–70, Feb. 1983. DOI: 10.1287/trsc.17.1.48.

[40] C. Jagtenberg, S. Bhulai, and R. van der Mei, "An efficient heuristic for real-time ambulance redeployment," *Operations Research for Health Care*, vol. 4, pp. 27–35, 2015, ISSN: 2211-6923. DOI: https://doi.org/10.1016/j.orhc.2015.01.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2211692314200075.

[41] R. S. Sutton and A. G. Barto, *Reinforcement Learning, second edition: An Introduction (Adaptive Computation and Machine Learning series)*. Bradford Books, Nov. 2018, ISBN: 978-0-26203924-6.

[42] S. Bhatt, "Reinforcement learning 101 - towards data science," *Medium*, Apr. 2019, ISSN: 2450-1292. [Online]. Available: https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292.

[43] C. Jagtenberg, S. Bhulai, and R. van der Mei, "An efficient heuristic for real-time ambulance redeployment," *Operations Research for Health Care*, vol. 4, pp. 27–35, 2015, ISSN: 2211-6923. DOI: https://doi.org/10.1016/j.orhc.2015.01.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2211692314200075.

[44] M. Maxwell, M. Restrepo, S. Henderson, and H. Topaloglu, "Approximate dynamic programming for ambulance redeployment," *INFORMS Journal on Computing*, vol. 22, pp. 266–281, May 2010. DOI: 10.1287/ijoc.1090.0345.

[45] V. Schmid, "Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming," *European Journal of Operational Research*, vol. 219, no. 3, pp. 611–621, 2012, Feature Clusters, ISSN: 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2011.10.043. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221711009830.

[46] P. L. V. D. Berg and J. T. V. Essen, "Comparison of static ambulance location models," *International Journal of Logistics Systems and Management*, vol. 32, no. 3-4, pp. 292–321, 2019.

[47] S. Zhang, L. Qin, Y. Zheng, and H. Cheng, "Effective and efficient: Large-scale dynamic city express," in *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2015, pp. 1–4.

[48] S. Ma, Y. Zheng, and O. Wolfson, "Real-time city-scale taxi ridesharing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 7, pp. 1782–1795, 2014.

[49] M. Qu, H. Zhu, J. Liu, G. Liu, and H. Xiong, "A cost-effective recommender system for taxi drivers," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 45–54.

[50] Association for Safe International Road Travel, *Road crash statistics*, Accessed: 2022-03-09, 2018. [Online]. Available: http://asirt.org/Initiatives/Informing-Road-Users/Road-Safety-Facts/Road-Crash-Statistics.

[51] Cleveland Clinic, *Sudden cardiac statistics*, Accessed: 2022-03-09, 2018. [Online]. Available: https://my.clevelandclinic.org/health/diseases/17522-sudden-cardiac-death-sudden-cardiac-arrest.

[52] Heart Foundation, *Sudden cardiac statistics*, Accessed: 2022-03-09, 2018. [Online]. Available: https://www.heartfoundation.org.au/your-heart/sudden-cardiac-death.

[53] World Health Organization, *Qatar: WHO statistical profile*, https://data.who.int/countries/634, Accessed: 2024.

[54] S. Saisubramanian, P. Varakantham, and H. C. Lau, "Risk-based optimization for improving emergency medical systems," in *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2015, pp. 702–708.

[55] P. L. van den Berg, J. T. van Essen, and E. J. Harderwijk, "Comparison of static ambulance location models," in *Proceedings of the 3rd International Conference on Logistics Operations Management*, 2016, pp. 1040–1051.

[56] V. Schmid, "Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming," *European Journal of Operational Research*, vol. 219, no. 3, pp. 611–621, Apr. 2012.

[57] C. Jagtenberg, S. Bhulai, and R. van der Mei, "Optimal ambulance dispatching," *Operations Research for Healthcare*, vol. 24, pp. 269–291, Mar. 2017.

[58] M. S. Daskin, "A maximum expected covering location model: Formulation, properties and heuristic solution," *Transportation Science*, vol. 17, no. 1, pp. 48–70, 1983.

[59] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[60] Z. Zhou and D. S. Matteson, "Predicting ambulance demand: A spatio-temporal kernel approach," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 2297–2303.

[61] D. Wang, J. Zhang, W. Cao, J. Li, and Y. Zheng, "When will you arrive? estimating travel time based on deep neural networks," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[62] B. S. Westgate, D. B. Woodard, D. S. Matteson, and S. G. Henderson, "Large-network travel time distribution estimation for ambulances," *European Journal of Operational Research*, vol. 252, pp. 322–333, 2016.

[63] Y. Wang, Y. Zheng, and Y. Xue, "Travel time estimation of a path using sparse trajectories," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 25–34.

[64] J. Yuan, Y. Zheng, C. Zhang, *et al.*, "T-drive: Driving directions based on taxi trajectories," in *Proceedings of the 18th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems*, 2010, pp. 99–108.

[65] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.

[66] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 2017, Second Edition, in progress.

[67] R. S. Sutton and A. G. Barto, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, pp. 229–256, May 1992.

APPENDIX A: A CALCULATION OF THE OBJECTIVE FUNCTION GRADIENT

In this appendix, we demonstrate the process of obtaining Equation 4.11, which is the gradient $\nabla_\theta J(\theta)$ of the objective function $J(\theta)$. Using Equations 4.7 and 4.10 as a basis, we obtain

$$\begin{aligned} J(\theta) &= \mathbb{E}_{s\sim\pi_\theta}[v(s)] = \sum_{s\in S} p(s)v(s) = \sum_{s\in S} p(s) \sum_{a\in A} \pi_\theta(s,a)q(s,a) \\ &= \sum_{s\in S}\sum_{a\in A} \pi_\theta(s,a)q(s,a) = \mathbb{E}_{(s,a)\sim\pi_\theta}[q(s,a)]. \end{aligned} \tag{A.1}$$

In this case, $s \sim \pi_\theta$ (or $(s,a) \sim \pi_\theta$) denotes that random initial state and policy π_θ are used to sample states (or state-action pairs (s,a)). The probability of states (or state-action pairings (s,a)) by adopting policy π_θ and beginning from a random initial state is thus denoted by $p(s)$ (or $p(s)\pi_\theta(s,a)$). $J(\theta) = \nabla_\theta$.

$$\nabla_\theta J(\theta) = \sum_{s\in S}\sum_{a\in A} p(s)\nabla_\theta \pi_\theta(s,a)q(s,a) \tag{A.2}$$

$$= \sum_{s\in S}\sum_{a\in A} p(s)\pi_\theta(s,a)\nabla_\theta \log \pi_\theta(s,a)q(s,a) \tag{A.3}$$

$$= \mathbb{E}_{(s,a)\sim\pi_\theta}[(\nabla_\theta \log \pi_\theta(s,a))q(s,a)]. \tag{A.4}$$

There is a mathematical transformation from Equation A.2 to Equation A.3.

$$\nabla_\theta \log \pi_\theta(s,a) = \frac{\nabla_\theta \pi_\theta(s,a)}{\pi_\theta(s,a)} \tag{A.5}$$

Equation A.5 is thus valid. There is further literature that contains the derivation [66].

Algorithm 4 Dynamic MEXCLP

Require: Demand d_i of each demand location $i \in V$, base locations $W \subseteq V$, busy fraction $q \in (0, 1)$, current destinations dest(a) for all $a \in$ IdleAmbulances $\subseteq A$, travel times τ_{ij} between any $i, j \in V$, time threshold T to reach an emergency call.

Output: New destination for the ambulance that is about to become idle. This ambulance should not be counted as an idle ambulance yet.

BestImprovement $= 0$.
BestLocation $=$ NULL.
for each $j \in W$ **do**
    CoverageImprovement $= 0$.
    for each $i \in V$ **do**
        $k = 0$.
        if $\tau_{ji} \leq T$ **then**
            $k++$.
            for each $a \in$ IdleAmbulances **do**
                if $\tau_{\text{dest}(a)i} \leq T$ **then**
                    $k++$.
                end if
            end for
            CoverageImprovement$+= d_i(1-q)q^{k-1}$.
        end if
    end for
    if CoverageImprovement $>$ BestImprovement **then**
        BestLocation $= j$.
        BestImprovement $=$ CoverageImprovement.
    end if