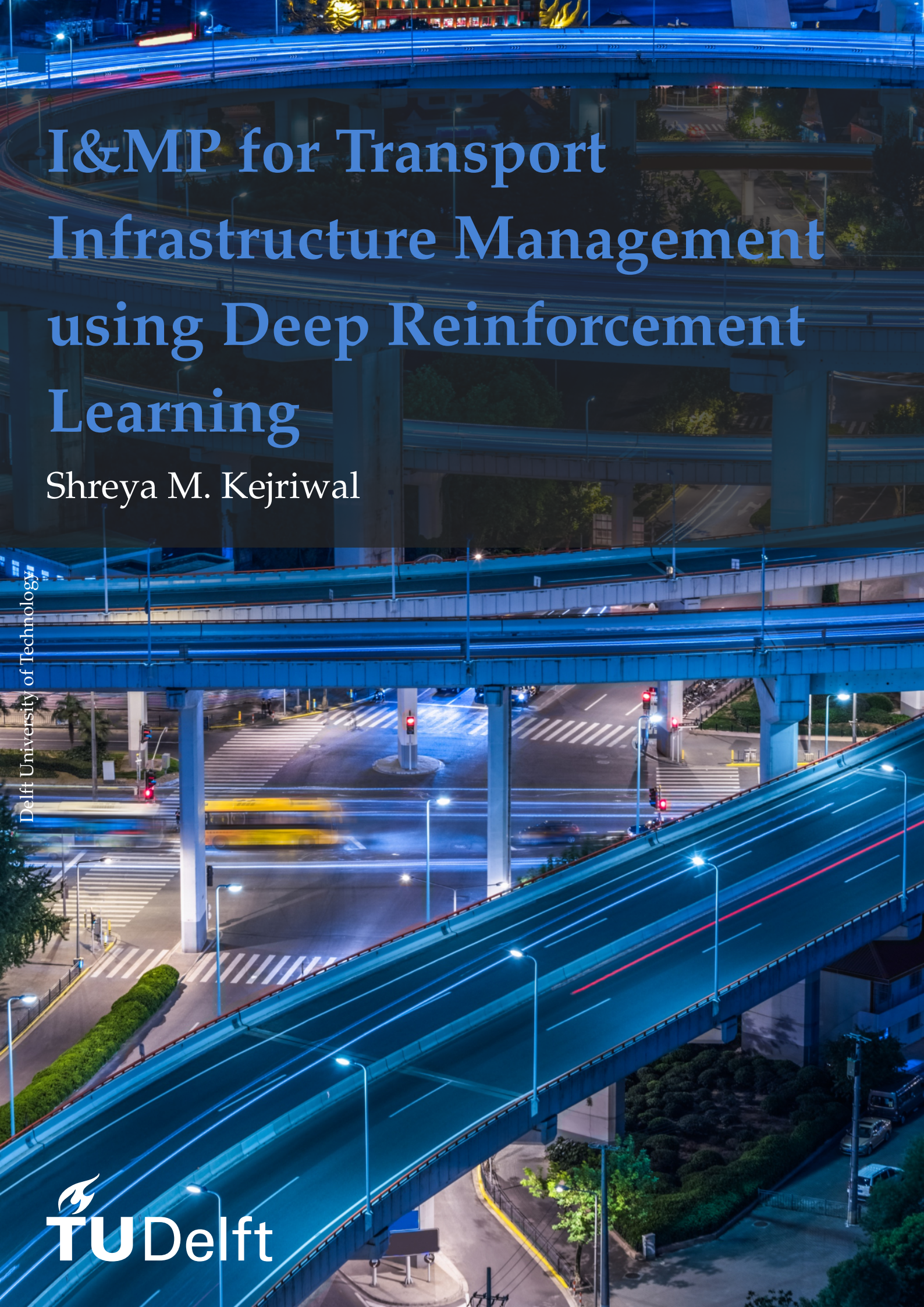
I&MP for Transport Infrastructure Management using Deep Reinforcement Learning

Shreya M. Kejriwal

Delft University of Technology



TU Delft

I&MP for Transport Infrastructure Management using Deep Reinforcement Learning

by

Shreya M. Kejriwal

Student Name

Shreya Kejriwal

Main mentor:	Dr. Charalampos Andriotis
Second mentor:	Dr. Mauro Overend
PhD Advisor:	Prateek Bhustali
External examiner:	Dr. Olindo Caso
Project Duration:	Novemeber, 2023 - July, 2024
Faculty:	Faculty of Architecture and Built Environment, Delft

Cover: Image by fanjianhua on Freepik

Style: TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Acknowledgements

This thesis has brought many challenges, and I am deeply grateful to those who have supported me throughout this journey.

First and foremost, I would like to thank my parents and my brother for their unwavering belief in me. Their love and support have been my rock, guiding me through the toughest situations. My parents have always been my biggest cheerleaders, encouraging me to pursue my dreams and never give up, no matter the obstacles. My brother, with his constant motivation and practical advice, has been an indispensable source of support. He has always been there to offer a listening ear. I am also deeply grateful to my grandparents, 'Nani' and 'Nana' for their love and steady presence in my life.

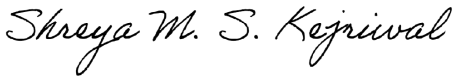
I extend my heartfelt thanks to my friends both in the Netherlands and in India. Thank you for listening to my endless ramblings about my topic and for boosting my confidence to face all challenges head-on. A special thank you to the 'Minions' group for their constant discussions and support throughout this thesis. I am particularly grateful to Prateek for his invaluable help. I could not have completed this thesis without your constant guidance and assistance. You have always made time to explain me everything in simpler terms and continue explore new ideas.

I would especially like to thank Shubham, who has always supported me, in both good or bad days. Your unwavering support and faith in me has helped me through this thesis.

My gratitude also goes to my mentors, Professor Charalampos Andriotis and Professor Mauro Overend. Thank you for listening to my concerns about the project and motivating me to push further. I deeply appreciate your support. I am also thankful to my external supervisor, Professor Olindo Caso, for helping me stay calm before presentations and for supporting my project throughout. Additionally, I would like to thank Ellen for her counselling and support during tough times.

When I began this thesis, I was overwhelmed by the vastness of the project and my inexperience. However, this project has taught me to tackle problems one at a time and has instilled in me the confidence to overcome obstacles with determination and willpower.

I look forward to what the future holds for me.



Shreya M. Kejriwal
Delft, July 2024

Abstract

The administration of transportation infrastructure entails addressing a multitude of obstacles arising from the intricate, fast changing and dynamic nature of the environment. This thesis focuses on improving infrastructure maintenance planning through the application of deep reinforcement learning. The study specifically models the transport environment by utilising a Markov Decision Process (MDP) framework and employing sophisticated DRL algorithms, such as Double Deep Q-Network (DDQN) and Deep Centralized Multi-agent Actor Critic (DCMAC).

The research begins with a thorough examination of current approaches in transportation network simulation, revealing the omissions in current strategies for infrastructure upkeep. This is followed by the development of a simulation environment that simulates real-world conditions and integrates pavement and bridge condition models to evaluate various maintenance strategies.

The proposed framework is tested on a simulated transportation network in the United States, which incorporates a traffic model to account for dynamic changes. The DRL algorithms are then used to make maintenance policies for this environment. These policies are evaluated through extensive simulations, with a focus on reducing maintenance expenses and improving the overall condition of the infrastructure. Results suggest that DRL can significantly improve decision-making processes in infrastructure maintenance, offering potential cost savings and better preservation of transport infrastructure quality.

Contents

Acknowledgements **i**

Abstract **iii**

Nomenclature **xii**

1 Introduction **1**

1.1 Motivation 1

1.1.1 Proposed Approach 3

1.2 Research Question 4

1.2.1 Research Sub Question(s) 4

2 Literature Review **5**

2.1 Literature review overview 5

2.2 Transportation Network Modelling 6

2.2.1 Infrastructure Maintenance Overview 6

2.2.2 Dynamic and Fast Changing Environments 7

2.2.3 System Interactions 8

2.2.4 Traffic impacts 8

2.3 Computation Background 9

2.3.1 Markov Decision Process (MDP) 10

2.3.2 Partially Observable MDPs (POMDPs) 11

2.3.3 Deep Reinforcement Learning (DRL) 12

2.4 Research Gap 13

3 Methodology **15**

3.1 General Framework 15

3.2 Environment Modelling . . . 16
3.3 Reinforcement Learning . . . 17
3.3.1 Double Deep Q-Network (DDQN) . . . 17
3.3.2 Deep Centralized Multi-agent Actor Critic (DCMAC) . . . 19
3.4 Experiments Setup . . . 21

4 Environment Modelling **23**
4.1 Overview . . . 23
4.2 Transportation Network . . . 24
4.2.1 Graph Representation . . . 24
4.2.2 Problem Formulation . . . 25
4.3 Pavement Modelling . . . 27
4.3.1 Critical Condition Index . . . 27
4.3.2 International Roughness Index . . . 28
4.4 Bridge Modelling . . . 28
4.4.1 Deck Condition Rating . . . 29
4.5 Action Space . . . 30
4.6 Transition Probabilities . . . 32
4.6.1 Impact of Actions on the Modelling Metrics . . . 33
4.7 Observation Probabilities . . . 35
4.8 Traffic Model . . . 35
4.8.1 Traffic Assignment Problem . . . 37
4.9 Evaluation Metrics Criteria . . . 41
4.9.1 Agency Cost Metric . . . 43
4.9.2 User Cost Metric . . . 46
4.9.3 Environmental Metric for CO_2 Emissions . . . 48
4.9.4 Safety Metric for Component and Network Safety . . . 51
4.10 Multi-attribute Utility Model . . . 52
4.10.1 Decision-makers Risk Attitude . . . 52
4.10.2 Utility functions . . . 53
4.11 Reward function . . . 54

5 Results and Evaluation **55**
5.1 Scenario Setup . . . 55
5.2 Scenario 1 . . . 56
5.2.1 Baseline for the Scenario . . . 57
5.2.2 Model Parameters . . . 58
5.2.3 Experiment Run 1 . . . 58
5.2.4 Experiment Run 2 . . . 60
5.2.5 Initial Reflection . . . 62
5.3 Scenario 2 . . . 63
5.3.1 Baseline for the Scenario . . . 65
5.3.2 Model Parameters . . . 65
5.3.3 Experiment Run 1 . . . 66
5.3.4 Experiment Run 2 . . . 68
5.4 Scenario 3 . . . 70
5.4.1 Model Parameters . . . 71
5.4.2 Experiment Run 1 . . . 71
5.4.3 Reflections . . . 74

6 Discussion and Conclusion **76**
6.1 Discussion . . . 76
6.1.1 Environment Model . . . 76
6.1.2 Reinforcement Learning Experiments . . . 78
6.1.3 Extension to the Built Environment . . . 79
6.2 Limitations . . . 81
6.3 Future Work . . . 81

A Environment Dynamics **90**
A.1 Pavement dynamics . . . 90
A.1.1 State-Action Transition Probability for Pavement features . . . 90
A.1.2 Observation Probability for Pavement features . . . 91
A.2 Bridge dynamics . . . 92
A.2.1 State-Action Transition Probability for Deck features . . . 92

A.2.2 Observation Probability for Deck features 93

B Algorithms used in the experiments **95**

B.1 Pseudo-code for algorithms 95

B.1.1 Double Q-Learning (DDQN) 95

B.1.2 Deep Centralized Multi-agent Actor Critic (DCMAC) 96

C Additional Experiment Runs **97**

C.1 Scenario 1 Runs 97

C.2 Scenario 2 Runs 98

C.2.1 Entire Policy for Run 2 in Scenario 2 98

C.3 Scenario 3 Runs 99

C.3.1 Policy for Run 0 in Scenario 3 99

List of Figures

3.1 Flowchart depicting the entire process for this thesis starting from data collection to bench marking the results . . . 16
3.2 Flowchart showing the approach used to model the transport environment as a Markov Design Process . . . 17
3.3 Double Deep Q-Network (DDQN) architecture defining both the networks (Firdous et al., 2023) . . . 18
3.4 Deep Centralized Multi-agent Actor Critic (DCMAC) architecture defining the actor and critic network (Andriotis and Papakonstantinou, 2019) . . . 20

4.1 Stylised pavement network presented by Medury and Madanat (2013) . . . 25
4.2 The transport network used for the experiments in Scenario 3 . . . 26
4.3 Mean Critical Condition Index (CCI) for different levels of traffic over the time horizon (Saifullah et al., 2024) . . . 32
4.4 Mean Deck rating for different levels of traffic over the time horizon . . . 34
4.5 Acceleration and deceleration vehicle velocity trajectory for different Levels of service (Margiotta and Washburn, 2017) . . . 39
4.6 Relationship between carbon emissions and vehicle speed (Barth and Boriboonsomsin, 2008) . . . 50

5.1 Policy generated with a non-aggressive deterioration model leading to only 'Do nothing' or 'Inspection' actions . . . 56
5.2 Policy realisation for scenario 1 with no constraints and only one objective to minimise i.e. the total agency cost . . . 59
5.3 Interaction between various costs . . . 60
5.4 Policy realisation for scenario 1 with budget constraints (70% budget allocated) and only one objective to minimise i.e. the total agency cost . . . 61

5.5 (a) Change in direct, indirect and mobilising cost w.r.t. change in budget allocated for the maintenance ; (b) Distribution of each cost in the total agency cost at different budget levels 63
5.6 Proposed transport network for scenario 2 which is part of the larger network studied on this thesis 64
5.7 Cost histogram for scenario 2 with no constraints and only one objective to minimise i.e. the total agency cost 67
5.8 Policy realisation for scenario 2 with with two objective to minimise i.e. the total agency cost and user cost 69

C.1 Run in scenario 1 when inspection action was suggested at every time-step when repair or maintenance. 97
C.2 Run in scenario 2 for all the components when minimising 2 objectives agency and user. 98
C.3 Run in scenario 3 for first 6 components with all the objectives 99
C.4 Run in scenario 3 for next 6 components with all the objectives 100

List of Tables

4.1 Adjacency matrix for the network 25
4.2 Attributes for every link of the network 26
4.3 Classification of all the possible states based on Critical Condition Index (CCI) values (Virginia Department of Transportation, 2018) 27
4.4 Classification of all the possible states based on International Roughness Index (IRI) (m/km) values 28
4.5 Classification of all the possible states based on Condition ratings for bridge component 29
4.6 Deck details for the assumed initial deck rating and deck age 30
4.7 Description of all the tasks performed in the maintenance actions for pavements and bridges 31
4.8 Impact of various actions on segment age and condition state 34
4.9 Matrix of trips between each node pair (vehicles/day) 36
4.10 Overview of the vehicles travelling in the network, their PCE values and growth rate till 2049 36
4.11 Duration of maintenance actions and the reduction in capacity as a result . . 38
4.12 Overview of Network Level of Service at 80kmph (Margiotta and Washburn, 2017) 40
4.13 Type of data needed for calculating different aspects of the metric 42
4.14 Overview of the impact of every action on different rewards 43
4.15 Deck failure probability given the current deck state 44
4.16 % of infrastructure cost expected as mobilising cost 45
4.17 Overweight cost for vehicle type: Truck and XL Truck 46
4.18 IRI based VOC factor parameters 47
4.19 CO_2 emissions for different inspection and maintenance action in gm/m_2 . . 49

4.20 CO_2 emissions per kilometre for vehicles considered in the network in g/km . 49

5.1 Baseline values for Scenario 1 for failure-replacement and time-based maintenance policy considering there is no hard constraint in the environment . . . 58
5.2 Model hyperparameters for experiment 1 listing the parameters setup for DDQN . . . 58
5.3 Exploration of the impact of various budget allocations on different costs impacting the agencies . . . 62
5.4 Attributes for every link of the network for scenario 2 . . . 64
5.5 Matrix of trips between each node pair (vehicles/day) for scenario 2 . . . 64
5.6 Baseline values for Scenario 2 for failure-replacement and time-based maintenance policy considering both agency and user costs . . . 65
5.7 Model Parameters for experiment 2 listing the parameters setup for DCMAC 66
5.8 Comparison of various costs derived in this run with the baseline and the 1st run . . . 70
5.9 Model Parameters for experiment 3 listing the parameters setup for DCMAC 71
5.10 List of scenarios with different metric weights . . . 72
5.11 Results when compared with the first experiment and baselines . . . 73

A.1 Observation Probability $p(o_t|s_t)$ for different inspection actions if true IRI state is s_t . . . 91

Nomenclature

Abbreviations

Abbreviation	Definition
CCI	Critical Condition Index
DCMAC	Deep Centralised Multi-agent Actor Critic
DDQN	Double Deep Q-Network
DRL	Deep Reinforcement Learning
DQN	Deep Q-Network
I&MP	Inspection and Maintenance Planning
IRI	International Roughness Index
MDP	Markov Decision Processes
NBI	National Bridge Inventory
PCE	Passenger Car Equivalent
POMDP	Partially Observable Markov Decision Processes
RL	Reinforcement Learning
TBM	Time based maintenance

1

Introduction

1.1. Motivation

A healthy infrastructure of roads, rail, inland waterways, ports, and airports is the backbone for sound economic development in the EU. Failing infrastructure immediately affects many other economic activities and often has dire consequences. In the past, many infrastructural works were designed without much thought about maintenance. However, infrastructure failure due to lack of maintenance is often very costly. The Dutch Ministry of Infrastructure and Water Management uses a rule of thumb that in case an asset fails, the cost to producers and consumers is ten times that of the cost of repair (Kerkhof et al., 2018). Furthermore, different infrastructures are closely connected and dependent on each other; failure in one affects the performance of the others (**?**).

Bridges and roads are essential components of transportation systems that enable the mobility of people and goods. However, they are also exposed to various sources of deterioration and damage over their long service life, such as traffic loads, environmental erosion, and extreme events (Yang, 2022). These factors can compromise the safety, functionality, and performance of the network, and pose significant risks to asset managers, governments, and users (Orcesi and Frangopol, 2011). Therefore, transportation agencies need to allocate

their limited resources efficiently and effectively to preserve the condition and service level of transport networks, and to manage the long-term risks associated with them (Federal Highway Administration, 2021).

Physical systems like bridges, viaducts, and tunnels in roads and railways are in place for a very long time—often 80 years or more. During such a period, many technological and other developments alter our ideas of what services the systems could/should deliver. Not keeping the infrastructure up to date through a lack of maintenance, renewal, and enhancement puts the entire economy at risk (**?**).

In addition to this, the environment and context around the network are constantly changing, making it difficult for maintenance agencies to plan their repairs. Factors such as traffic variations, weather conditions, policy changes, and economic fluctuations contribute to the dynamic nature of transportation networks. These fluctuations can significantly impact the scheduling and prioritisation of maintenance activities, leading to challenges in ensuring timely and effective interventions. The complexity is further compounded by the diverse types of infrastructure within a network, the varying needs and preferences of users, and the interdependencies between different system components. Maintenance plans must therefore be adaptable and flexible, capable of responding to both predictable and unforeseen changes in the operational context.

In addition to this, the environment and context around the network are constantly changing, making it difficult for maintenance agencies to plan their repairs. Factors such as traffic variations, weather conditions, policy changes, and economic fluctuations contribute to the dynamic nature of transportation networks. These fluctuations can significantly impact the scheduling and prioritisation of maintenance activities, leading to challenges in ensuring timely and effective interventions. The complexity is further compounded by the diverse types of infrastructure within a network, the varying needs and preferences of users, and the interdependencies between different system components. Maintenance plans must therefore be adaptable and flexible, capable of responding to both predictable and unforeseen changes in the operational context.

To address this challenge, this research proposes to use a deep reinforcement learning framework to optimise the lifecycle management of a transport network. Deep reinforcement

learning is a combination of reinforcement learning and deep learning, which is a type of neural network that can learn from complex and high-dimensional data. Deep reinforcement learning has been successfully applied to various optimisation problems in transportation, such as traffic signal control (Chu et al., 2022), vehicle routing (Joe and Lau, 2020), and transit network design (Haydari and Yılmaz, 2022). However, the application of deep reinforcement learning to the lifecycle management of transportation infrastructure assets, especially transport networks, is still limited and the interest in the field is growing.

Maintenance constitutes an inevitable, albeit often invisible, part of countries' transport policies. Infrastructures are increasingly connected to each other, which means that failure in one infrastructure asset will affect the performance of the whole system. In addition, the demand for transport infrastructure has increased as more and more people use it. This accelerates the ageing of infrastructure and leads to an increased need for maintenance. However, strains on public budgets have often resulted in reductions in maintenance budgets. The situation is further aggravated by the effects of climate change, bringing additional pressures on these infrastructures. One of the key challenges in lifecycle management is to optimise the maintenance and inspection plans. Bridges and roads are critical elements of transportation networks, but they are also vulnerable to various deterioration mechanisms, such as corrosion, fatigue, and cracking (Mendoza Lugo et al., 2024) (Calvert et al., 2020).

1.1.1. Proposed Approach

Fast-changing environments are a major obstacle to transport inspection and maintenance planning, which has multiple ripple effects on various aspects of society. It is therefore necessary that the plans created are adaptable and consider the dynamism of the context. They should therefore be adaptable and flexible. It is difficult to find a policy that will work with a diverse transport system with multiple types of infrastructure, various users, and multiple preferences. Our study aims to address this challenge by developing a transport modelling framework with deep reinforcement learning to overcome this scheduling problem for efficient transport infrastructure management.

It is imperative to develop a transport network environment that can capture the dynamism of the system, context, and needs before defining the reinforcement learning model and

testing different policies. Once the environment has captured the real-world transition of the system, various reinforcement learning algorithms are used to generate policies that optimise the set objectives. The study's objectives encompass the economic cost of maintenance on both transport agencies and users, as well as minimising the environmental impact of the measures taken to maintain a functional transportation system.

This study contributes to the larger context of infrastructure maintenance planning using deep reinforcement learning by giving a deeper understanding of changing environments, the factors that affect them, and how they can be modelled. This helps to create more informed and effective policies and improvements that will benefit society by providing an economically viable, sustainable, and highly functional transportation system.

1.2. Research Question

The research question for this thesis is:

How can a multi-asset transportation network be effectively modelled to account for dynamic influences such as traffic variations, policy changes, and system interdependencies to optimise the transportation network management?

1.2.1. Research Sub Question(s)

1. How does the alterations in traffic flows over time impact the agencies, users and societal factors/ costs?
2. What methods can be employed to incorporate stakeholder objectives, in the form of preference shifts, into the reinforcement learning model?
3. How effective is Deep Reinforcement Learning for developing I&M policies in such fast changing environments?

2

Literature Review

2.1. Literature review overview

This section provides a brief overview of the literature reviewed in the next section. It examines literature related to methods of transport environment modelling, including how the systems within the network interact and how the model considers the uncertainty and dynamism from the real-world scenarios. This is followed by a review of the literature related to computational aspects, including Markov design processes, reinforcement learning, and various algorithms. The following sections are based on the aspects that led to the research gap.

The research process began with the selection of terminologies that were used to search and index various articles from journals like Nature, Association for Computing Machinery, Journal of Machine Learning Research, International Journal of Engineering Business Management, and Public Works Management & Policy, to name a few. A short list of terminologies used included:'sequential decision-making', 'infrastructure management', 'lifecycle management', 'lifecycle risk assessment', 'asset management','multi-asset management', and 'traffic impact on infrastructure'.

2.2. Transportation Network Modelling

2.2.1. Infrastructure Maintenance Overview

Maintenance management is important for ensuring the smooth operation and longevity of assets, equipment, and infrastructure across diverse industries. Effective maintenance management is imperative for reducing costs and maximising productivity, regardless of whether it pertains to manufacturing plants, transportation networks, healthcare facilities, or residential buildings. In industries like manufacturing and transportation, unexpected breakdowns or failures almost always lead to big problems, lost money, and compromised safety (Saifullah et al., 2022). The types of maintenance categories available are:

- **Reactive Maintenance**: In this approach, repairs and maintenance activities are performed when equipment breaks or fails. It involves addressing issues as they arise without a predefined maintenance schedule or proactive measures to prevent failures. With reactive maintenance, the focus should be on restoring functionality rather than preventing failures.
- **Preventive Maintenance**: This is a proactive approach to upkeep that involves regular checks, routine tasks, and a systematic care for equipment and assets. The primary objective of preventive maintenance is to prevent equipment failures and reduce the likelihood of unexpected breakdowns by identifying and addressing potential issues before they occur. Maintenance helps organisations extend the life of equipment, optimise performance, and reduce downtime.
- **Predictive Maintenance (PdM)**: This is an advanced approach to maintenance management that utilises data analysis, condition monitoring, and predictive modelling to anticipate and prevent equipment failures. Rather than relying on fixed schedules or reactive responses, predictive maintenance leverages real-time data and analytics to identify anomalies, patterns, or trends that indicate imminent failures. This allows organisations to schedule maintenance tasks precisely when they need, optimising resource allocation and minimising downtime.
- **Reliability-Centred Maintenance (RCM)**: This approach focuses on identifying and prioritising maintenance tasks based on the criticality and risk associated with equip-

ment failures. RCM is aimed to optimise maintenance strategies by determining the most effective and efficient maintenance actions to ensure the reliability and performance of assets while minimising costs. RCM goes beyond traditional time-based or condition-based maintenance approaches by considering the consequences of failures, the likelihood of occurrence, and the overall impact on operations.

2.2.2. Dynamic and Fast Changing Environments

Dynamic and fast-changing environments are characterised by continuous fluctuations in conditions, driven by various internal or external factors. In the context of a transport network maintenance problem, these factors might include dynamic changes in the number of users or shifts in decision-maker preferences when faced with multiple objectives. Learning in such environments poses significant challenges for an agent, as it necessitates up-to-date information about the state of all influencing factors at each time step.

One method proposed in the literature to address this challenge is state augmentation. This technique involves incorporating certain elements of these fluctuating factors into the state space for the agent to observe. According to Wiering (2001), this approach can enhance the agent's ability to adapt to changing conditions. However, state augmentation can significantly expand the state space, leading to potential issues with convergence and the attainment of an optimal policy. Despite these drawbacks, the technique is often employed due to its relative ease of implementation.

While state augmentation offers a practical solution, it is essential to recognise its limitations and seek further advancements in this area. The primary concern is the exponential growth of the state space, which can complicate the learning process and hinder the development of efficient policies. Future research should aim to explore more sophisticated methods that balance the complexity of the state space with the need for accurate and timely information. Potential avenues include dimensionality reduction techniques, advanced modelling approaches, and hybrid methods that combine state augmentation with other learning strategies.

In summary, dynamic and fast-changing environments present substantial learning challenges for agents due to the constant variability in influential factors. While state augmen-

tation is a useful technique for addressing these challenges, its implementation must be carefully managed to avoid the pitfalls of an overly expansive state space. By advancing our understanding and developing more refined methods, we can improve the efficiency and effectiveness of agents operating in such complex environments.

2.2.3. System Interactions

System interdependencies pertain to the interactions and influences among various components of the transportation network, such as diverse bridges and road networks. These factors can have a significant impact on the structural performance and also determine the functional obsolescence of a network (Bektas and Albughdadi, 2021).

The need for a structural review of the current bridge infrastructure to deal with ageing infrastructure is highlighted by failures in past structural design. Highway agencies are interested in improving budget efficiency and reducing environmental impact by managing network-level bridges across their entire life cycle. Similar environmental conditions and traffic loads between adjacent bridges may result in correlated degradation or failures. The spatial proximity between bridges has the greatest impact on environmental correlation effects, but traffic volume correlation effects are predominantly driven by traffic volumes (Lei et al., 2023).

Bridges with similar structural ratings patterns over a long period of time may be highly correlated. Deterministic and stochastic deterioration models have been developed for various bridge components, such as deck, superstructure and substructure, and average daily traffic. For example, if one particular cluster is highly enriched by structurally deficient (SD) bridges, then we can identify other parameters that are similar to these bridges and therefore, we can control them. If this structural deficiency is attributed to the deterioration of deck rating, we can recommend that the bridge authorities implement deck-related rehabilitation measures.

2.2.4. Traffic impacts

One of the most critical assets in transportation infrastructure is the stock of bridge structures. Improper use or inadequate maintenance of these bridges can disrupt the traffic

network, causing perturbations that increase travel expenses and potentially lead to significant economic losses. The initial step in mitigating these issues involves determining the load effects resulting from current traffic conditions (Mendoza Lugo et al., 2024).

For instance, increased traffic demand and loading on a bridge can arise due to population growth and heightened economic activity in the area. Conversely, maintenance budgets may shrink due to fiscal constraints faced by transportation authorities. Additionally, the performance of one bridge can influence others within the network due to their interdependence. These dynamic factors necessitate periodic adjustments to maintenance and inspection plans.

Extensive research has been conducted on the impact of load on bridge structures, as documented in studies such as Sjaarda et al. (2020), Maljaars (2020), and Demir and Dicleli (2023). These studies typically assume the full availability of traffic data from Weigh-In-Motion (WIM) systems and comprehensive information on bridge geometry and material properties. However, reliance on WIM-based systems presents challenges: (i) WIM sensors are not installed at all locations, and (ii) detailed structural information for all bridges is not always accessible.

Analysing traffic load effects on bridges, particularly the Extreme Load Effects (ELEs), is essential for assessing the impact of heavy traffic. ELEs represent the maximum forces, such as bending moments and shear forces, experienced by a bridge throughout its lifespan. This analysis is crucial for ensuring the safety and reliability of infrastructure design and maintenance.

2.3. Computation Background

Sequential decision-making involves a series of decisions made over time by an agent. This procedure finds widespread application in the field of engineering design, wherein the agent gathers data on various alternatives, evaluates them, and ultimately selects the most favourable option. Decisions are made sequentially, navigate through a decision space, select information sources or evaluation methods, and determine whether to continue gathering information. In such situations, the agent observes, acts, observes again, and repeats the cycle. Future actions are frequently influenced by previous observations, and the rationale

behind each action typically serves to inform subsequent decisions.

In uncertain domains, techniques derived from control theory and operations research are frequently employed to address a sequential decision problem. These methodologies aid in identifying optimal policies, as well as in assessing the value of information and control. The control problem encompasses the broader framework of reinforcement learning, in which the agent not only learns to predict the values of encountered environmental states, but also utilises these predictions to optimise actions, ultimately maximising rewards.

2.3.1. Markov Decision Process (MDP)

Markov Decision Processes (MDPs) provide a mathematical framework for modelling problems that require sequential decision making, a crucial component of reinforcement learning. Within an MDP, an agent consistently engages with its environment by choosing actions at each time step and receiving rewards based on these actions. The state of the environment changes according to the agent's current state and chosen action. The primary goal of the agent is to maximise its cumulative reward over a specific time horizon (Sutton and Barto, 2018).

In probability theory, the Markov property refers to the memory-less property of a stochastic process. The probability distribution of future states of the process depends solely on the present state if this property holds true. In simpler terms, predicting the next word in a sentence depends solely on the current word, and not on the words that came before it. The Markov property is upheld in a model if the values in any state are influenced only by the values of the immediately preceding state or a small number of immediately preceding states (Gudivada et al., 2015).

MDPs are usually represented by a tuple $\langle S, A, P, R, \gamma \rangle$, where:

- **S** is the set of possible states of the system.
- **A** is the set of possible actions for the agent.
- **P** is the probability of transitioning from state s to state s' after taking action a. It is represented as $P(s'|s, a)$

- **R** is the reward received by the agent for taking action a in state s. It is represented as $R(s, a)$
- $\gamma \in [0, 1]$ is the discount factor that determines how much the agent values future rewards over immediate rewards.

A solution to a problem in MDP is a policy, which is a rule that tells the decision maker what action to take in each state. There are different methods to find the optimal policy, such as value iteration, policy iteration, or Q-learning. The goal of the agent in a Markov Decision Process (MDP) is to find the best action for each state that maximises the total rewards it can get in the future with a discount factor. The agent selects actions a_t that optimise the expected discounted return G_t, over a time horizon T:

$$G_t = R_{t+1} + \gamma R_{t+2} + ... = \sum_{k=0}^{T} \gamma^k R_{t+k+1} \tag{2.1}$$

The action-value function $Q_\pi(s, a)$, with $s \in S$ and $a \in A$, is defined as the maximum expected return achievable by following a particular policy $\pi : S \rightarrow A$, after observing some state s and then taking some action a:

$$Q_\pi(s_t, a_t) = \mathbb{E}\left[G_t \mid s_t = s, a_t = a, \pi\right] \tag{2.2}$$

Nonetheless, the assumption made in MDPs may not always be accurate, primarily due to the inherent nature of the issue in real-world situations and the stochastic transitions. Therefore, it is not possible to always have complete information about the system. To overcome this disadvantage, the MDP is extended and the POMDPs are discussed in the next subsection.

2.3.2. Partially Observable MDPs (POMDPs)

When dealing with real-world situations, such as machines and systems, it is often the case that the agent or decision maker does not have a complete understanding of the environment. In such cases, an extension to MDP is used, known as POMDP. Instead of directly observing the state, the agent receives only a partial or noisy signal at each time step. This creates

ambiguity in the decision-making process, as the agent must ascertain the true state of the environment based on its observations. A POMDP with a finite horizon can be defined by the tuple $(S, A, P, \Omega, O, R, \gamma)$, where S, A, P, and R are the same as in an MDP. In addition:

- Ω is the set of possible observations
- O is the observation model that gives the probability of seeing an observation o_0 given that the agent was in state s_0 and took action a in the previous time step. $O(o_0, s_0, a) := P(o_0|s_0, a)$.

A belief state b is a probabilistic representation of the agent's estimation of the current state within the state space, which helps with partial observability (Krachtopoulos, 2023). The belief state is also called the belief in the POMDP literature. The conditional probability of a belief state at any given moment is determined by the history of actions and observations. b_0 is the initial belief state before any actions or observations. The current belief state, the action a, and the new observation o' can be used to update the belief state using the Bayesian rule as b':

$$b'(s') = (P(o' \mid s', a)\Sigma_s P(s' \mid a, s)b(s)) \div (\eta(o' \mid b, a)) \tag{2.3}$$

2.3.3. Deep Reinforcement Learning (DRL)

Using a partially observable space with multiple components can often create a large state space. Assuming a simple system comprising 12 components with 6 possible states, the resulting state space will be approximately equivalent to $6 * *12$, i.e. approximately 2 billion. Traditional methods, such as Q-Learning or Tabular methods, are not feasible in such cases due to the exponential growth in state-action pairs. To overcome this DRL was introduced in Mnih et al. (2015). Deep neural networks, which are global function approximators, are used to substitute the value and action-value functions.

This narrows the focus area to certain classes/ methods of reinforcement learning for efficient training. In this research 2 methods are focused on Double Deep Q-Network (DDQN) and Deep Centralised Multi-agent Actor Critic (DCMAC) algorithm. DDQN is a extension to Q-learning and DQN that uses 2 networks to formulate the policy. The actor-critic method is

one of these methods that combines aspects of both value-based and policy-based methods. Actor critic methods use two neural networks for the decision-making process. There are two components to the decision-making process: the actor decides which action to take, and the critic evaluates the actor's actions. This separation allows for more nuanced and stable learning, as the actors can improve their policy based on feedback from the critic, which estimates the value function. Both the methods are discussed in the following sections.

2.4. Research Gap

An extensive review of the literature has been conducted to identify and present the various state-of-the-art methodologies used to evaluate and develop maintenance policies. Additionally, this review summaries the types of infrastructure that have been studied. Despite the recognised importance and complexity of transport network management, there remains a significant gap in research addressing the dynamic influences of spatial environments, policy alterations, and system inter-dependencies on the lifecycle management of bridge networks.

Most existing studies focus on individual bridges or isolated aspects of bridge networks, such as structural health monitoring, deterioration modelling, or maintenance optimisation. These studies often fail to account for the uncertainties and variability's of real-world scenarios, such as changes in traffic flows, weather conditions, budget fluctuations, regulatory changes, and stakeholder preferences. Furthermore, current research lacks efficient methods to model multi-asset problems with multiple objectives, a crucial aspect for comprehensive bridge network management.

Moreover, these studies do not adequately consider the interactions and feedback mechanisms within the system that impact the performance, safety, and sustainability of the entire network. Consequently, there is a pressing need for research that optimises the lifecycle management of bridge networks by incorporating the dynamic influences of spatial environments, policy alterations, and system inter-dependencies.

In summary, while significant strides have been made in individual bridge maintenance and specific aspects of bridge network management, there is a notable deficiency in comprehensive approaches that address the complex, dynamic nature of real-world bridge

networks. Future research should focus on developing integrated models that consider these multifaceted influences to enhance the effectiveness and sustainability of bridge network management.

3

Methodology

3.1. General Framework

The primary goal of this thesis is to illustrate the significance of modelling dynamic changes in the environment and evaluate the impact of such changes on objectives. This impact is studied by examining the objective wise cost and the overall effectiveness of an inspection and maintenance policy for the network. The research is structured into two primary sections: modelling the transport network environment and creating models for reinforcement learning. Figure 3.1 provides the comprehensive structure of the research process.

The research was designed in a simulated transportation environment located in the United States. The environment is developed using the Partially Observable Markov Decision Process (POMDP) framework and includes a traffic model to account for the changes in the network. Lastly, a multi-attribute utility model combines and compares various objectives into a single reward function. The second part of the research creates the reinforcement learning agent and examines its ability to develop policies based on the environment. For this thesis, two algorithms, namely Double Deep Q-Learning and Deep Centralized Multi-agent Actor Critic, are considered. Numerous scenarios were designed and tuned to assess the effectiveness of the modelling and algorithm on policy creation.

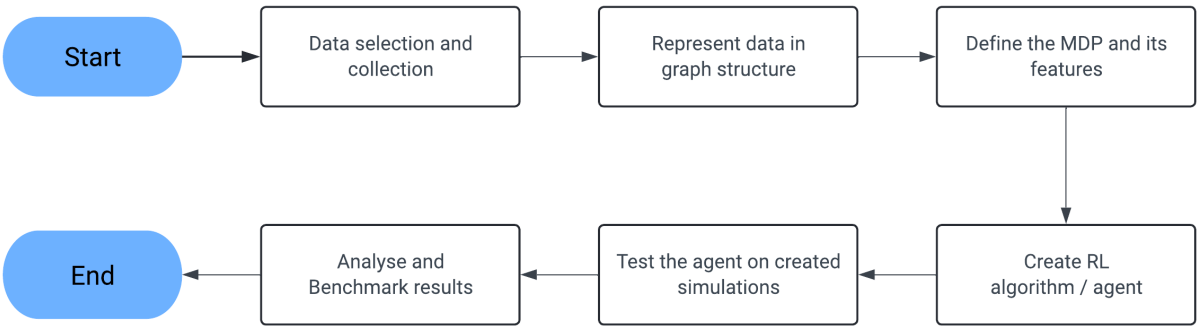


Figure 3.1: Flowchart depicting the entire process for this thesis starting from data collection to bench marking the results

The outcomes of this research are expected to provide insights into the benefits of dynamic environment modelling and inform the development of efficient inspection and maintenance policies for transportation networks.

3.2. Environment Modelling

Environment modelling is a critical aspect of reinforcement learning, where an agent interacts with a simulated environment to learn optimal decision-making strategies. This involves creating a digital representation of the real world or a specific scenario, allowing the agent to explore and experiment without real-world consequences.

The transport network used in this study is derived from a paper in the same field of research by Medury and Madanat (2013), with some abstractions to make it suitable for this research question. These abstractions include the creation of supplementary links and the substitution of few pavement elements with bridge elements. To efficiently represent the dynamics and context, a hierarchical decomposition of the network typology is done, organising it into links and segments, as seen in the flowchart in 3.2. These changes ensure the model reflects real-world scenarios of a multi-asset problem.

This hierarchical structure is then used to define the system as a graph with nodes, edges and edge attributes. Based on the typology, features for functional and structural aspects are defined to formalise the problem as a Markov Decision Process (MDP). Once the transitions in the environment are defined, the reward function is created based on the objectives. In

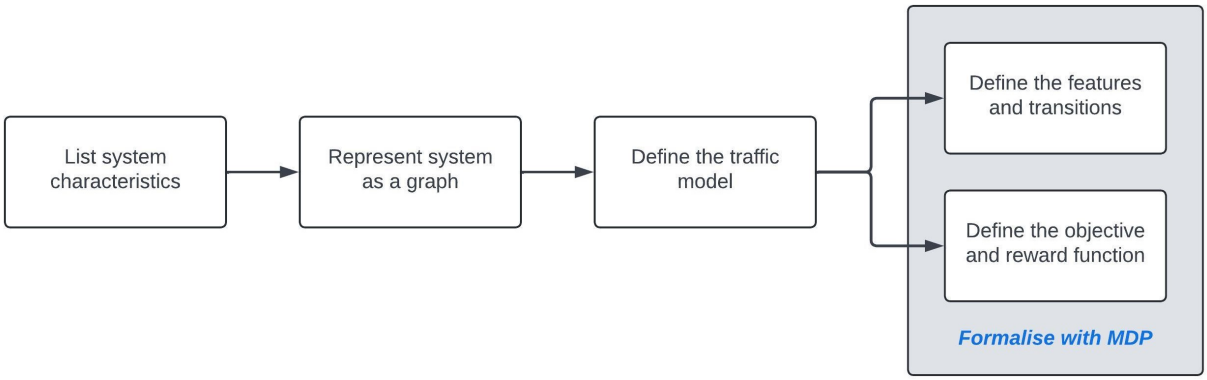


Figure 3.2: Flowchart showing the approach used to model the transport environment as a Markov Design Process

this research, there are four competing objectives that address the various stakeholders involved in the transport network's use and management. This suggests that a policy aimed at maintaining a certain part of the network might impact users by causing delays. The environment should effectively present these interconnections and dynamics between various entities.

The detailed aspects of the modelling approach are discussed further in the following chapters.

3.3. Reinforcement Learning

The next aspect of the research is to define a reinforcement learning model. There are many approaches to reinforcement learning, but the complexity of the problem makes certain approaches better than others. Two different algorithms, namely, Double Deep Q-Network and Deep Centralized Multi-agent Actor Critic, are discussed in the next subsections.

3.3.1. Double Deep Q-Network (DDQN)

Before we can understand DDQN, we must first understand the basics of Deep Q learning. In Deep Q-learning (DQN), the target value is calculated directly using a greedy algorithm, similar to Q-learning. Although this approach can lead to quick convergence of Q values towards the optimisation target, it often results in overestimated target values (Zhang et al.,

2022). This overestimation occurs because Q-learning uses the max operator, which selects and evaluates an action using the same values, leading to overly optimistic value estimates. The Double Deep Q-Network (DDQN) algorithm addresses this issue. Introduced by van Hasselt et al. (2015), DDQN extends the DQN algorithm originally published in 2010 by the same author.

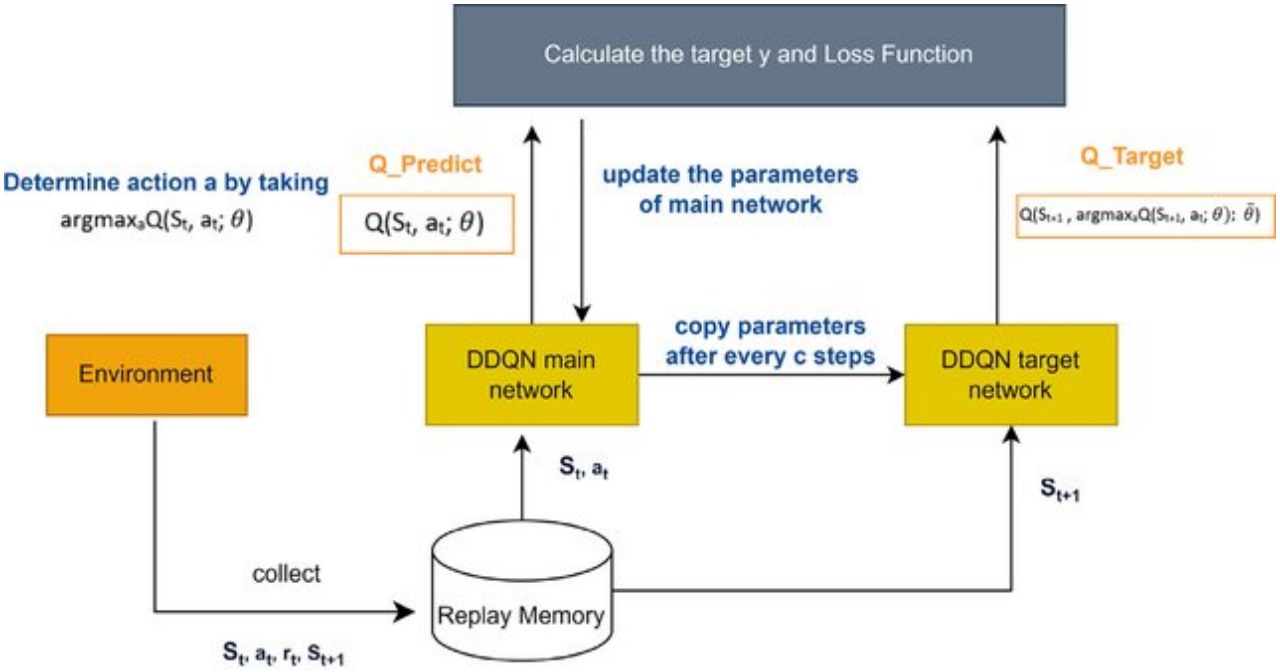


Figure 3.3: Double Deep Q-Network (DDQN) architecture defining both the networks (Firdous et al., 2023)

There are three key differences between the DDQN algorithm and the traditional Q-learning. In lieu of employing a conventional look-up table, DDQN employs a deep neural network to determine the optimal action estimation $Q(s, a, \theta) \approx Q'(s, a)$. Additionally, an experience replay mechanism is implemented. The data from exploration is saved as experience. The samples are then randomly selected from the stored memory data units to train the neural network parameters. Finally, the DDQN algorithm utilises two separate networks to decouple the action selection from the target value calculation, thereby mitigating the overestimation problem. The networks have the same structure and asynchronous parameters as the networks. The parameter θ of the current network Q is utilised to select the action that corresponds to the maximum Q value, whereas the parameter θ^- of the target network Q^- is utilised to evaluate the Q value of the optimal action. This approach facilitates the reduction of bias by guaranteeing that the policy remains based on values derived from

the current network weights. The loss function of the DDQN algorithm is formulated as follows:

$$L = \mathbb{E}\left[\left(r + \gamma \underset{a'}{max} Q(s', a') - Q(s, a)\right)^2\right] \tag{3.1}$$

This equation represents the expected squared difference between the predicted Q value and the target Q value, where r is the reward, γ is the discount factor, s' is the next state, and a' is the action that maximises the Q value for the next state. Good control performances are achieved when dealing with random disturbances. However, the action set of the DDQN algorithm is typically constrained, which will lead to a limited action set. The pseudo form of DDQN algorithm is shown in the Appendix in Algorithm 1.

However, when the solution spaces grows exponentially due to the environments complexity, DDQN can struggle to handle such a large space (Zhang et al., 2022). This often leads to long training times and sub optimal policies. This also impacts the convergence and scalability of the problem. In such cases a algorithm which can leverage their architecture to naturally suit such problems is needed. One such algorithm in literature is Deep Centralized Multi-agent Actor Critic (DCMAC) (Andriotis and Papakonstantinou, 2019) that will be discussed in the following section.

3.3.2. Deep Centralized Multi-agent Actor Critic (DCMAC)

The DCMAC method is a novel Deep Reinforcement Learning technique approach introduced by Andriotis and Papakonstantinou (2019) for inspection and maintenance planning of systems. which has evolved from actor critic methods which uses two neural networks. This splits the evaluations from the actions. The Actor critic method is formulated as a multi agent problem which conditionally assumes that the actions taken on a system are independent from each other which reduces the size of the action space considerably. For example consider a problem with 5 components and 4 actions, the action space in such a case is 4^5 which means 1024 combinations if it is formulated as a single agent problem. If the same problem is formulated in DCMAC which is a multi agent formulation the action space can be $4 \times 5 = 20$ actions, 4 actions for 5 components each. Therefore the policy generated

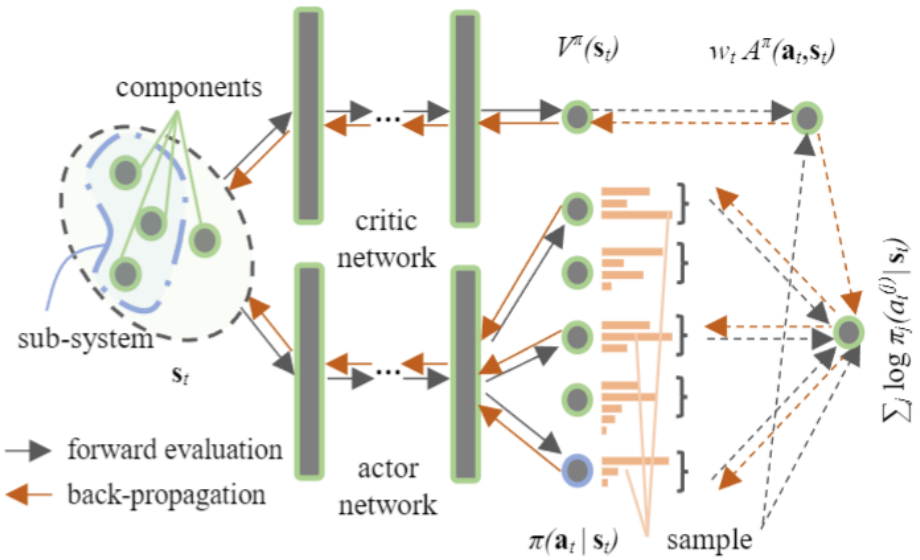


Figure 3.4: Deep Centralized Multi-agent Actor Critic (DCMAC) architecture defining the actor and critic network (Andriotis and Papakonstantinou, 2019)

that guides the actions is presented as $\pi = \prod_{i=1}^{n} \pi_i$ where $\pi : S \rightarrow A_i$ is the policy for all element n in the system significantly reducing the action space and enabling the algorithm to work on large scale problems.

In DCMAC, deep neural networks are utilised to generate improvement actions and approximate the critic value using actor and critic network parameter vectors, as θ_π and θ_v. To compute the policy function an off policy gradient estimator is used to generate a behaviour policy using importance sampling. DCMAC approximates the advantage function i.e. $A^\pi(s_t, a_t) = Q^\pi(s_t, a_t) - V^\pi(s_t)$ following temporal difference:

$$A(s_t, a_t \mid \theta^V) = c(s_t, a_t) + \gamma V^\pi(s_{t+1} \mid \theta^V) - V^\pi(s_t \mid \theta^V) \tag{3.2}$$

The critic network is updated through the mean square error by:

$$L_V(\theta^V) = \mathbb{E}_{s_t \sim \omega, a_t \sim \mu}[w_t(c(s_t, a_t) + \gamma V^\pi(s_{t+1} \mid \theta^V) - V^\pi(s_t \mid \theta^V))^2] \tag{3.3}$$

and its respective gradient, computed by back-propagating the weighted advantage function

through the critic network. DCMAC uses hidden layers of the actor network to extend into distinct output nodes using a centralised system information. Each segment is assigned one output node. Each segment than generates a probability distribution over the action options for that segment. The critic network provides a single approximate value of the total costs in the environment over the planning horizon which also includes implementation impacts. To train the actor and critic neural networks, the DCMAC method is executed over multiple episodes. Each episode simulates the condition states, identifies the improvement actions for the entire planning horizon, and stores the realised states, selected actions, and associated total costs in a replay buffer. This replay buffer is then used for training the networks through backpropagation. The pseudo code presented in the paper is referred in Appendix Algorithm 2.

For both the algorithms, an existing implementation for DDQN and DCMAC was used with minor changes in parameters and settings from IMPRL GitHub managed by Prateek Bhustali (Bhustali, 2023). The changes primarily focused on aligning the environment with the algorithm in terms of data structure and notations. Apart from this, the model parameters for each of the algorithms were adjusted to improve the convergence and performance of the model.

3.4. Experiments Setup

The previous sections dealt with the creation of the model and the reinforcement agent. To analyse and demonstrate the impact of the dynamic influences included in the environment, multiple scenarios are set up. The experiments begin with a simplified scenario and gradually incorporate more complexities and variables with each run. Three main scenario configurations are created:

1. **Scenario 1:** This scenario involves multiple individual components that do not form a system. A single maintenance-related objective is selected, with the goal of minimising the cost of this maintenance objective.
2. **Scenario 2:** This scenario involves a part of the final network, where the components are interconnected, and the policy for one component affects the others. Two competing objectives are selected, with the goal of minimising the costs associated with both

objectives.

3. **Scenario 3:** This scenario involves the entire simulated network, where the components are interconnected, and the policy for one component influences the other components. There are four objectives that need to be minimised in this case based on the decision makers preference.

The results from these experiments are analysed to evaluate the effectiveness of the model, reward function and the reinforcement agent in handling varying levels of complexity and interdependence within the simulated environment.

4

Environment Modelling

4.1. Overview

The performance and quality of the inspection and maintenance schedule depend on the nature and state of the environment model. In order to develop an effective inspection and maintenance plan it is essential to understand the transport environment as a whole with its interdependencies, dynamic changes, and overall effects the deteriorating elements on various factors. It is essential to have clear knowledge about the functioning of the environment along with the factors that cause deterioration.

This chapter details the development and construction of the transport environment that will be used in the study. The chapter begins with the structuring of the transport environment followed by discussing about the characteristics, topology, and conditions of the elements in the environment. This will be followed by the formulation of the environment as a Partially Observable Markov Decision Process. We finally end the chapter by describing the Multi-Attribute Utility theory to construct the reward function as discussed in the POMDP.

4.2. Transportation Network

The aim of this project is to illustrate the importance of considering dynamic changes in the form of change in traffic dynamics and agencies attitude towards risk and see its impact on the government, users, and the environment. This is done by creating an effective inspection and maintenance policy for efficient transport network management as a whole. To evaluate the impact of multiple attributes on the inspection and maintenance plan, we will consider a hypothetical example. The system discussed here is a multi-lane highway system in an urban area. It is assumed that the network is located in the United States, which is why all the data and values used further in this chapter are region-specific. While this doesn't give a true picture of an existing network, it works well as a framework for experimenting and illustrating the impacts of various factors on the management of a transportation network. Lastly, the horizon of the project is for 20 years.

4.2.1. Graph Representation

A transport network consists of various points of interest like junctions, stops, intersections to cities or areas, that are connected via routes or paths for efficient movement of people and goods. Therefore, the network's structure can range from very simple to extremely complex depending on the interpretation and representation of the system. The system can be mapped out using various mathematical tools but one of the most intuitive approaches for such problem typologies is graph theory (Żochowska and Soczówka, 2018).

Within graph theory there are multiple ways to create graphs like using physical coordinates capturing the connection between nodes or using linear distance measure between nodes being the most common methods. In this study the first method is used where the entire network is represented as a graph $G = (V, E)$ with nodes V being places of interest in a state and edges E being the routes that connect them. The adjacency matrix of dimensions $V \times V$ for the network is presented in Table 4.1 where each value denotes the absence (if $A_{i,j} = 0$) or existence (if $A_{i,j} > 0$) of a link between two nodes.

Nodes	0	1	2	3	4
0	0	1	0	0	0
1	0	0	1	0	1
2	0	0	0	1	1
3	0	0	0	0	1
4	0	0	0	0	0

Table 4.1: Adjacency matrix for the network

4.2.2. Problem Formulation

For this case study the network is adapted from a stylised pavement network used by Medury and Madanat (2013) seen in Figure 4.1. In this paper the authors used this network to create a pavement management system using approximate dynamic programming. The network represents a realistic representation of transport networks with a combination of series and parallel system through 11 individual segments and 10 nodes. This network was also further expanded by Zhou et al. (2022) to include bridges in segment 4 and 9 to include impacts of traffic. As this research involves dealing with network performance attributes like congestion and service levels the extended network was taken as a base.

Figure 4.1: Stylised pavement network presented by Medury and Madanat (2013)

To reflect the the diverse characteristics of the sections within a edge E_i, it is further divided into segments S such that $(S_1, S_2, S_n) \in E_i \; \forall \; E \subset G$. The segment can be either a bridge or a pavement. These segments are divided either based on length, location of a local intersect, change in topology of the environment or the typology of the segment. The system of representation aids in representing the network close to real life systems taking into consideration diverse topologies. The network keeping in mind these factors is presented in Figure 4.2. The network consists of 5 nodes connected by multiple segments representing either a bridge or a pavement. There are 12 segments where 8 are pavements and 4 are bridges. It is a combination system including both series and parallel segments which

encourages examination of vehicle routing and congestion due to reduced capacity when a segment or edge is under repair and maintenance.

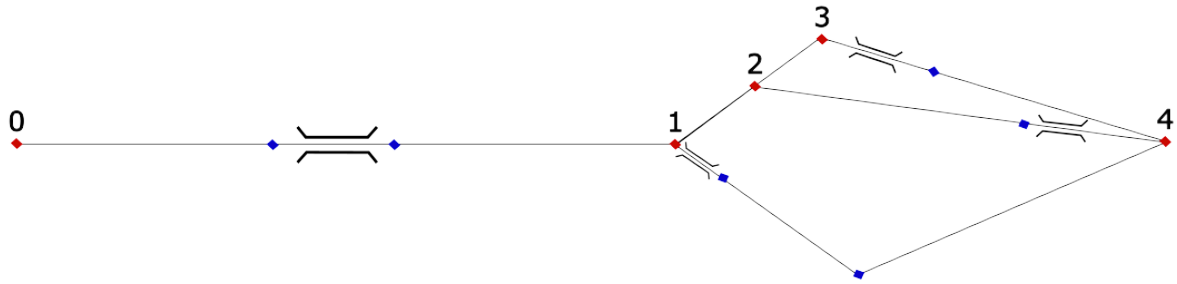


Figure 4.2: The transport network used for the experiments in Scenario 3

We further describe the id, length, number of lanes and capacity for each segment in Table 4.2. Width for the lanes is constant at 3.5m each based on modern construction standards. The capacity is calculated based on Federal Highway Administration (2018) capacity computation methods where using HCM method (HERS method) is recommended for capacity and LOS calculations.

Link	Segment	Type	Length (km)	No. of lanes	Capacity (vehicles/day)
0	0	Pavement	6.2	2	1872.49
	1	Bridge	1.14	2	2031.81
	2	Pavement	8.1	2	2067.62
1	3	Pavement	6.7	2	2477.049
2	4	Pavement	3.92	2	2095.84
3	5	Bridge	2.54	2	1781.81
	6	Pavement	4.51	2	1735.65
4	7	Pavement	8.8	2	1798.39
	8	Bridge	2.6	2	1917.61
5	9	Bridge	1.85	2	1756.78
	10	Pavement	7.5	2	1889.72
	11	Pavement	9.2	2	2034.6

Table 4.2: Attributes for every link of the network

Real-world systems are not always certain, hence a POMDP problem was created to address the inherent uncertainty. This model can capture the stochastic nature of the problem. In the following sections different components of the POMDP are explained.

4.3. Pavement Modelling

To simulate the deterioration of a pavement segment, two metrics are used, namely the Critical Condition Index (CCI) and International Roughness Index (IRI), to capture the comprehensive deterioration pattern. For every segment in this case, it is assumed that the segments deteriorate independently.

4.3.1. Critical Condition Index

The Critical Condition Index (CCI) is a numerical index that spans from 0 to 100 and is utilised to assess the severity of pavement distress. It was initially developed by the US Army Corps of Engineers based on the *PAVER* methodology. It is calculated as the lower of two ratings: the load distress rating (LDR) and the non-load distress rating (NDR). LDR is used to measure pavement distress, such as fatigue, rutting, and cracking, which is mainly caused by excessive vehicle loads. NDR considers the weathering of pavement material and construction deficiencies, such as lateral and longitudinal cracking, bleeding, or separation (Virginia Department of Transportation, 2018). The CCI provides a comprehensive assessment of the condition of the pavement and is used to guide maintenance and rehabilitation decisions (Bryce et al., 2012).

CCI State	CCI values	Description
$s = 6$	100 - 90	Excellent
$s = 5$	89 - 80	Very good
$s = 4$	79 - 61	Good
$s = 3$	60 - 50	Fair
$s = 2$	49 - 37	Poor
$s = 1$	< 37	Very poor

Table 4.3: Classification of all the possible states based on Critical Condition Index (CCI) values (Virginia Department of Transportation, 2018)

Table 4.3 summaries the grouping of CCI values based on the condition. These conditions range from excellent to very poor. A pavement section with a CCI value below 60 is categorised as 'deficient'.

4.3.2. International Roughness Index

The pavement surface deterioration is measured using the International Roughness Index (IRI) in m/km. IRI is a way of figuring out how uneven the pavement is m/km. (Gillespie et al., 1986). It is a commonly used metric for measuring surface roughness, which helps in determining the ride quality for the users. A higher IRI value indicates a surface that is uneven and rough, resulting in poor ride quality, according to Virginia Department of Transportation (2018).

IRI State	IRI values	Description
$s = 5$	< 0.95	Very good
$s = 4$	0.95 - 1.56	Good
$s = 3$	1.57 - 2.19	Fair
$s = 2$	2.20 - 3.14	Mediocre
$s = 1$	> 3.15	Poor

Table 4.4: Classification of all the possible states based on International Roughness Index (IRI) (m/km) values

The categories and ranges seen in Table 4.4 are derived from Saifullah et al. (2022), which correspond to the ride quality descriptors presented by Federal Highway Association (FHWA) in Federal Highway Administration (2002). This categorisation can be seen in the work of Faddoul et al. (2013) and United States Department of Transportation (2000). An average IRI of 2.20 or greater is deemed 'deficient' in terms of ride quality.

Since we use two features to model the pavements current condition, the belief of the agent for the component is defined as a single array of length (11,) that combines both the IRI belief followed by the CCI belief. This singular array representation streamlines the belief update process, provides an integrated approach to capturing the overall dynamics, and provides a concise approach to defining the pavements condition.

4.4. Bridge Modelling

The definition of a bridge is determined by its geometry, which comprises several components. Specifically, three distinct components, namely the deck, superstructure, and substructure, are considered when devising inspection and maintenance projects (Federal Highway Administration, 2023b). This thesis solely addresses the modelling and

degradation of bridge decks. As stated in the manual Rossow (2003), it is evident that well-maintained decks hold significant importance for the bridge system due to their proximity to traffic and their ability to safeguard the rest of the structure. This makes it susceptible to numerous weather-related deterioration's, such as rain, freezing and thawing cycles, as well as mechanical wear caused by traffic (Saifullah et al., 2024).

4.4.1. Deck Condition Rating

For rating the bridge deck's condition, the FHWA condition rating system is used, which ranges from 9 being a new or undamaged state to 0 being a failure. This rating system has been widely adopted by various departments of transportation across the US, including the Penn DOT (Pennsylvania Department of Transportation and Bureau of Design, 2022). For this, work the state classification is based on a study by Manafpour et al. (2018) also used in Saifullah et al. (2024) to characterise states until 9 individually and the states after 4 are considered Thus, we have 7 states in the system (6 intact and 1 failure). The table 4.5 shows different states and a general description of how the bridge is in each state.

Deck State	Condition	Description
$s = 9$	Excellent	NA
$s = 8$	Very good	No problems noted
$s = 7$	Good	Some minor problems noted
$s = 6$	Satisfactory	All the primary structural elements are sound; however, they may exhibit minor section loss, cracking, spalling, or scour
$s = 5$	Fair	Sectional loss, deterioration, spalling, or scour
$s = 4, ...$	Poor - Critical	Secondary structural components have been severely affected by loss of section, deterioration, spalling, or scour. Concrete may have local failures and shear cracks
$s = failed$	Failed	Out of service

Table 4.5: Classification of all the possible states based on Condition ratings for bridge component

The information pertaining to the initial state of the decks is provided in Table 4.6, which comprises of deck rating derived from the condition rating index and the deck age, which refers to the age of construction or repair, whichever occurs later. The deck's belief over all the state is modelled as a single array of length (7,) for each component.

Bridge ID	Deck Rating	Deck Age
1	6	6
5	5	9
8	6	12
9	7	2

Table 4.6: Deck details for the assumed initial deck rating and deck age

The information presented in the above sections helps define the state space of the environment. It consists of the belief of the agent over each component. It should be noted that, to maintain a uniform structure for the reinforcement learning model, the beliefs of the deck component are padded with zeros. Time is also included for the model to keep track of its position and other factors w.r.t to the entire horizon (Boyan and Littman, 2000).

4.5. Action Space

The primary goal of inspection and maintenance in optimizing transport infrastructure is to find the most appropriate and best combination of inspection and maintenance actions to ensure that the infrastructure remains in a sufficiently safe state. Each category and type of action is associated with a cost. Therefore, it is imperative to make decisions based on multiple criteria, such as the necessity of actions, the available budget, and other relevant factors.

This research examines the possibility for multiple inspection and upkeep procedures to be carried out simultaneously for each infrastructure component. For inspection, there are three available actions, and for maintenance, there are four available actions. It is assumed that any action taken on the pavement will improve both the CCI and the IRI together. The description for each action based on the segment type is described in Table 4.7. The action choices are as follows:

1. **Inspection actions:**
    - **No inspection (a_0^i):** No inspection is performed. The state of the network component remains ambiguous. The observation error is ∞.
    - **Routine Inspection (a_1^i):** A low fidelity inspection is performed. It is used to

evaluate the overall condition of the network element. The beliefs over the state become more certain through this inspection, but the observation error is higher than in-depth inspection.

- **In-depth Inspection (a_2^i):** An in-depth inspection is performed. This is a more comprehensive examination of the network component. This inspection helps to strengthen beliefs about the state.

2. **Maintenance actions:**

- **Do Nothing (a_0^m):** No action is performed. Based on the deterioration model, the condition of the network element continues to decline.
- **Minor Repair (a_1^m):** The network element is scheduled for minor repair and maintenance work. The state of the infrastructure improves slightly because of the transition model.
- **Major Repair (a_2^m):** The state of a network element is partially restored. This includes major modification work that will improve the original performance or capacity of the infrastructure slightly.
- **Reconstruction (a_3^m):** The network element is fully replaced. The element has been restored to its original state with absolute certainty.

Actions	Description (Pavement)	Description (Deck)
Do nothing	NA	NA
Minor repair	Moderate patching (<10%), surface treatment, partial depth patching, thin Asphalt Concrete (AC) overlay	Moderate cracks filling and patching area <10% of the deck area, minor replacement of reinforcement
Major repair	Heavy patching (<20% of the pavement area), full depth patching, structural overlay	Fixing Spalls/delamination with deck area <25%, major replacement of reinforcement
Reconstruction	Replacement of the entire pavement section	Replacement of the entire deck

Table 4.7: Description of all the tasks performed in the maintenance actions for pavements and bridges

As previously mentioned in the preceding sections, it can be inferred that the transport network segments undergo individual degradation, thereby allowing for the allocation of

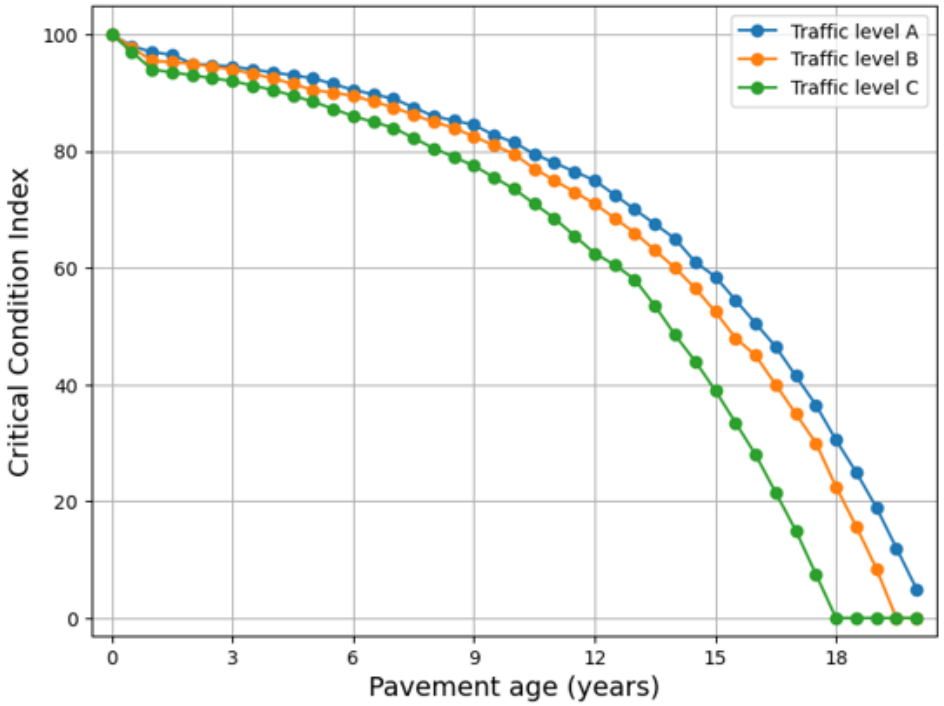


Figure 4.3: Mean Critical Condition Index (CCI) for different levels of traffic over the time horizon (Saifullah et al., 2024)

actions for each time-step to each segment. Since there are 3 inspections and 4 maintenance actions that occur simultaneously, the resulting action space is of size 12.

4.6. Transition Probabilities

Transition probabilities is a probabilistic model that describes the likelihood of a component being in state S and moving to state S' given an action A is taken by the agent. Each feature will be given a distinct transition probability matrix. Since the problem is a hypothetical network, we consider various models from various studies in this thesis. The transition model for IRI and state action transitions is based on Saifullah et al. (2022), which is a stationary model. The CCI state-action transition probabilities were retrieved from Saifullah et al. (2024).

The non-stationary gamma process proposed in Saifullah et al. (2022) was utilised for modelling the CCI features. The paper uses a modified version to accommodate various traffic levels (A to E), where A indicates heavy traffic and E indicates light traffic. The

data obtained from the Virginia Department of Transportation Pavement Management System (PMS) was utilised to model the mean CCI for different levels of traffic, as illustrated in Figure 4.3. It is assumed that the other structural parameters are constant (Yan et al., 2023). This thesis only considers traffic levels A, C, and E as high, medium, and low traffic scenarios.

The curves from each Traffic level are fitted in a non-stationary gamma process to show the deterioration over multiple episodes across the entire horizon. The generated sequences were then converted into transition probabilities for each traffic level. In this case, Prof. Charalampos Andriotis provided the Matlab code and transition probabilities for different traffic levels for different traffic levels. The transition probabilities for the pavements are presented in the Appendix A.1.1. The data and process for time varying CCI transition model can be found in Saifullah et al. (2024).

The deck transition model is defined using a similar process. This transition model is based on a deck study on the bridges of New Jersey (New Jersey Department of Transportation and Federal Highway Administration, 2015). A small subset of the data is used for interstate highway is used for this thesis. As the number of data points is limited to 6 deterioration instances, the model, and results seen in Figure 4.4 are experimental at best. Due to the lack of multiple sources and data, further research is necessary to confirm the credibility of this model. The gamma process was fitted to the dataset values through a series of trials and experiments involving the a and b values. Based on this, transition probabilities were calculated (Lou et al., 2016). The probabilities for the bridge are presented in the Appendix A.2.1.

4.6.1. Impact of Actions on the Modelling Metrics

Due to the varying environmental conditions, this research focuses on non-stationary transitions between time-steps. It is imperative that we assign each segment of the network with an age parameter, which will be formally referred to as the effective age of the component. The effective age identifies the rate of degradation of the component. As the effective age increases, the rate of decline also increases. Each maintenance action has different impacts on the age of the component, in this case, the age in relation to CCI for pavements and

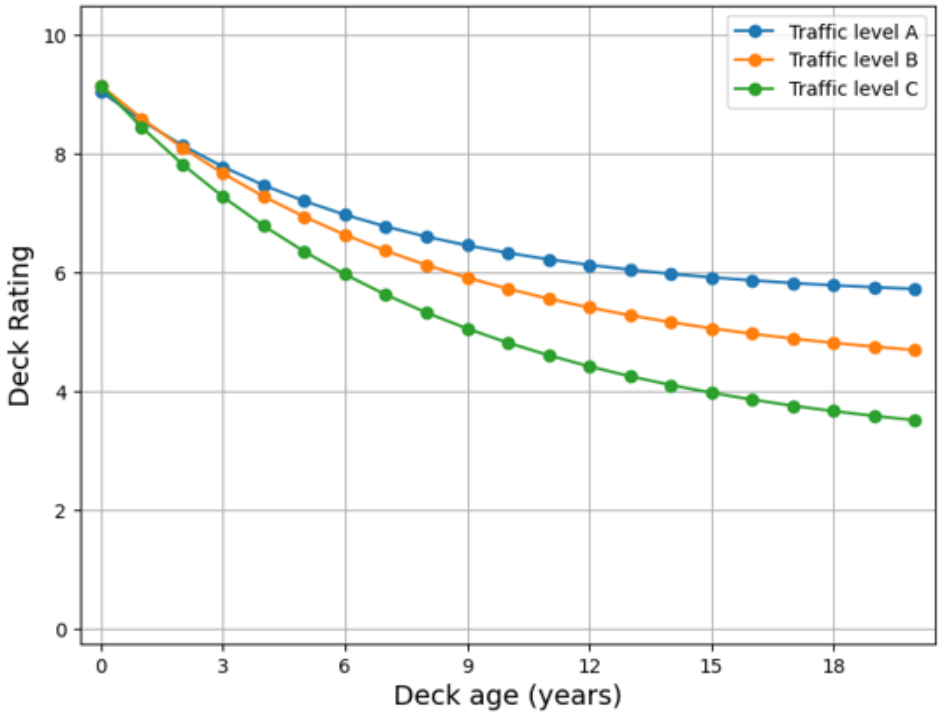


Figure 4.4: Mean Deck rating for different levels of traffic over the time horizon

the deck age for bridges. The component age and condition state update model has been retrieved from Saifullah et al. (2024). The relationship between age, condition, and action taken is explained in Table 4.8.

Actions	Segment effective age	Segment condition state
Do nothing	Increase by 1	Component continues to deteriorate based on the rate determined by the age.
Minor repair	Increase by 1	Change in state dependent on the state action transition model, but the rate of deterioration doesn't change.
Major repair	Reduced by 5 ($T_{age} = max(0, T_{age} - 5)$)	Change in state dependent on state action transition model and rate of deterioration also changes.
Reconstruction	Reset to 0	Reset to intact state.

Table 4.8: Impact of various actions on segment age and condition state

4.7. Observation Probabilities

As the agent is required to explore an environment where beliefs regarding states alter over time, asset managers may utilise tools such as inspection to aid in analysing the true condition of the segment. This reduces the degree of uncertainty of the belief. Based on the inspection action, there are three types of observations that can be obtained from the environment. Each action gives an observation with an error where no inspection means 'infinity error,' which reduces if the actions are routine or in-depth inspection. The model doesn't use the observation in the case of infinite error, and only the deterioration model affects the belief. The observation model for different inspection types is considered from (Saifullah et al., 2024). A more in-depth understanding of the segment's true state is created by the observations gathered based on the likelihood, progress, and current state of the elements. All the observation probabilities for both the pavements and bridges are presented in the Appendix A.1.2 and Appendix A.2.2.

4.8. Traffic Model

Traffic conditions on transport networks are becoming increasingly dynamic and complex as urbanisation continues to grow. Traditional models frequently presume that traffic conditions remain constant over time, however, this assumption is far from the truth. Traffic volumes are rising annually, and the composition of vehicles on the road is changing significantly. For instance, the number of goods transport vehicles and trucks is increasing, altering the overall traffic patterns. These modifications may have significant implications on the long-term efficacy and management of transportation networks.

It is essential to understand these evolving traffic conditions for accurate traffic modelling and network management. In this study, the transport network is shown as a directed graph, where traffic flows in designated directions between nodes. The specific number of trips between each node is detailed in Table 4.9.

To provide a more accurate representation of the network load and capacity factors, it is essential to incorporate the different types of vehicles using the network. This is achieved by converting all vehicle trips into a common volume range using Passenger Car Equivalent (PCE) values (Bhowmick and Mitra, 2017). First, the distribution of vehicle types (Federal

Nodes	0	1	2	3	4
0	0	3000	1000	0	1500
1	0	0	4500	2500	6000
2	0	0	0	0	9000
3	0	0	0	0	4000
4	0	0	0	0	0

Table 4.9: Matrix of trips between each node pair (vehicles/day)

Highway Administration, 2023c) in different trips is identified, then the volume for each trip is determined based on these distributions. This approach ensures a more precise estimation of the network's load and capacity. Table 4.10 lists the types of vehicles considered, along with their PCE values, distribution percentages, and projected growth rates (Federal Highway Administration, 2023a). The volume is determined by $\sum_{i=1}^{v} Trips_i \times PCE_i$ for the entire network.

Type of vehicle	PCE value	Distribution (in %)	Growth rate till 2049 (in %)
Motorcycle	0.50	0.28	0.50
Passenger Car	1.00	72.16	0.50
Bus	3.00	18.30	0.50
Truck	4.50	1.72	1.80
XL Truck	5.00	7.54	1.20

Table 4.10: Overview of the vehicles travelling in the network, their PCE values and growth rate till 2049

Yearly trip updates are calculated using the growth rate forecast for each vehicle type. According to the S&P Global pessimistic and optimistic economic outlooks, the 20-year forecast of annual growth in total Vehicle Miles Travelled (VMT) ranges from 0.5% to 0.9% (Federal Highway Administration, 2024). For this thesis the capacity for the segments is calculated as $c_i = c_j \times N \times f_w \times f_{HV} \times f_p$. Here, c_i is the capacity of the segment; c_j is the lane capacity under ideal conditions with design speed of j; N is the number of lanes; f_w is the lane width and clearance factor. Since all the segments in the network are constructed using modern standards and regulations the factor is set to 1 and f_p is driver population factor which is set between 0.85 to 1.0 for travellers on interstate and freeways (Lu et al., 1997). f_{HV} is heavy vehicle factor calculated using $f_{HV} = 1/(1 + P_T(E_T - 1)$ where P_T is Passenger car equivalent (PCE) of vehicle type and E_T is the Proportion of vehicle type.

Under ideal conditions, US typical link capacities are 2200 vehicles, but due to varying conditions, they can range from 1700 to 2200 vehicles.

4.8.1. Traffic Assignment Problem

The traffic assignment problem is a transportation routing optimisation problem. The approach aims to identify the optimal routes for all vehicles within the network, taking into consideration network constraints such as capacity, congestion, and speed. It is a key tool for managing transportation networks. In this study, the traffic assignment problem is utilised for various purposes, among them being:

1. Examine the disruption resulting from maintenance actions within the network, specifically in terms of diminished capacity of certain components and links.
2. Consider the effects of maintenance actions or increased traffic on travel time and speed.
3. Study the detour patterns resulting from maintenance actions for route disruptions and augmented travel time.

Since all of these studies will be conducted at annual intervals with the time-steps and horizon of the MDP, it is assumed that traffic will remain constant from one node to another for the year's span. The duration and capacity of the maintenance actions can be observed in Table 4.11. Based on this data, we can ascertain the base travel time on the network, as well as the travel times for planned maintenance actions. If multiple significant repairs and reconstruction initiatives are planned concurrently, it will result in an increase in travel time across the network as a result of the ripple effect, which can lead to excessive delays, detours, and congestion. This has a significant effect on the network's functionality and management. This does not negate the significance of undertaking maintenance measures for an efficient network. Hence, the traffic assignment model aids the agency in comprehending the impact of maintenance measures and devising a plan accordingly.

There are various traffic assignment models, including all-or-nothing assignment (AON), incremental assignment, capacity restraint assignment, user equilibrium assignment (UE), stochastic user equilibrium assignment (SUE), system optimum assignment (SO), etc. In

Type of action	Duration:Pavement (Days/lane-km)	Duration:Deck (Days)	Capacity (in %)
Inspection	0	0	100
Do nothing	0	0	100
Minor repair	3	25	95
Major repair	5	120	85
Reconstruction	25	365	45

Table 4.11: Duration of maintenance actions and the reduction in capacity as a result

this thesis, the User Equilibrium assignment model is employed owing to its relatively straightforward implementation and computational efficacy. Additionally, it provides an accurate estimate of traffic flow in numerous scenarios. The user equilibrium model is based on the Wardrop equilibrium principle. According to Wardrop and Whitehead (1952), in a transportation network, the travel time on all utilised routes between an origin and a destination is equal, and no driver can enhance their travel time by unilaterally altering routes. This suggests that if a driver decides to change their path, it will result in or influence the decision of other drivers. The mathematical formulation for the model is as follows:

$$\begin{aligned} Minimize\ Z &= \sum_{a} \int_{0}^{x_\alpha} t_\alpha(x_\alpha) dx \\ s.t. \sum_{k} f_k^{rs} &= q_{rs} \forall\ r, s \\ x_\alpha &= \sum_{r} \sum_{s} \sum_{k} \delta_{\alpha,k}^{rs} f_k^{rs} : \forall\ \alpha \\ f_k^{rs} &\geq 0 : \forall\ r, s, k \\ x_\alpha &\geq 0 : \forall\ \alpha \in A \end{aligned} \tag{4.1}$$

The Bureau of Public Roads (BPR) function will be the cost function that will be minimised (Bureau of Public Roads, U.S. Department of Commerce, 1964). The BPR function is used to predict the network's travel time using free flow speed, capacity, and volume. These factors also have an impact on the speed and duration of travel during maintenance operations and congestion. The BPR function is given in Equation 4.2.

$$tt_{x_\alpha} = fft_\alpha \times [1 + a(\frac{x_\alpha}{capacity_i})^b] \tag{4.2}$$

where, tt_{x_α} is the travel time given x_α is the current traffic; fft_α is the free flow speed given the traffic flow; $capacity_i$ is the segments capacity and a and b are model coefficients with values 0.15 and 4.0. Based on this calculated travel time, the model iteratively assigns traffic volume to different edges using travel time as weight to find the shortest path until model convergence.

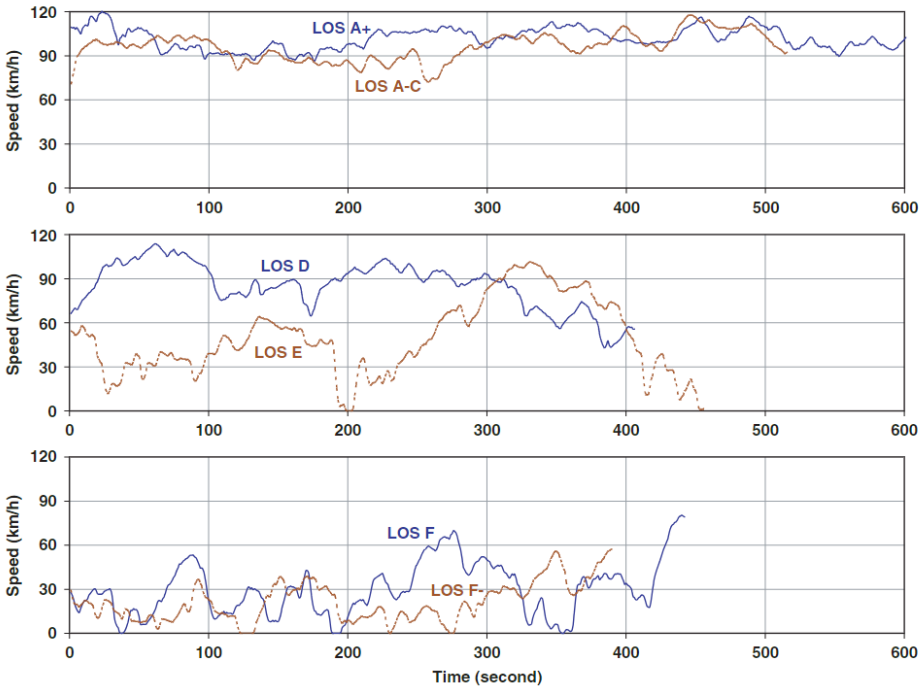


Figure 4.5: Acceleration and deceleration vehicle velocity trajectory for different Levels of service (Margiotta and Washburn, 2017)

Based on this calculated travel time, the model iteratively assigns traffic volume to different edges, using travel time as a weight, until the model converges. We extend this model to also calculate the maximum service flow based on travel time, volume, and capacity. This enables the model to calculate the average speed across the links, and therefore the density and Level of Service (LOS) of the links and the network. Table 4.12 gives a brief overview of the different levels of services at 80kmph base free flow speed (Margiotta and Washburn, 2017). The Los is a measurement that can be logically linked to emissions. There are various LOS values, denoted by the letters A through F. A typical vehicle velocity trajectory will

show different characteristics for each LOS. Under LOS A, vehicles typically travel near the free-flow speed of the highway, exhibiting minimal acceleration or deceleration disturbances, as depicted in Figure 4.5. As the LOS conditions get worse (i.e., LOS C, D, E, and F), vehicles travel at lower average speeds with more acceleration and deceleration events (Barth and Boriboonsomsin, 2008). Here, the maximum flow refers to the ADT, or the hourly traffic volume. When there are maintenance actions, the maximum service flow is multiplied by the max flow ratio, calculated as ($max\_service\_flow = max\_flow \times capacity\_ratio$). This is then used to classify the link into a level of service.

Level of Service	Density in cars/km/lane	Max Service Flow in cars/km/lane	Speed in kmph
LOS A	< 7	< 600	75 - 80
LOS B	7 - 12	600 - 1000	64 - 75
LOS C	12 -16	1000 - 1400	54 - 64
LOS D	16 - 22	1400 - 1670	40 - 54
LOS E	22 - 28	1670 - 2000	32 - 40
LOS F	> 28	> 2000	< 32

Table 4.12: Overview of Network Level of Service at 80kmph (Margiotta and Washburn, 2017)

Each level of service is associated with an average speed window, which is then used to back in the BPR function loop to calculate the correct travel time given the change in level of service and network speed (California County Association of Governments, 2005). These functions and values are then used in various reward metrics to calculate the detour or congestion in the network.

Thus, the outputs generated from the traffic model are:

- Base travel time on the network: The travel time given the maximum free flow speed and the density of vehicles on the network.
- New travel time based on the link capacities: The travel time of the link is determined by the actual free flow speed and density, based on the link's capacity.
- Flow of vehicles on each link: Determine the volume and distribution of vehicles on each link.
- Speed range of the vehicles on each link: The average speed range of the vehicles,

considering the level of service, density, and capacity of the link.

- Level of Service of the network: The network's performance as perceived by users is determined by travel duration, speed, and network's density.

These outputs are crucial for effective traffic management and planning. They allow for the evaluation of different scenarios and the optimisation of maintenance actions. This model provides a robust framework for understanding and improving the performance of transport networks by incorporating dynamic traffic conditions, vehicle type distributions, and detailed capacity factors. This model provides insight into strategic decisions, ensuring that transport networks remain efficient, safe, and capable of meeting future demands.

4.9. Evaluation Metrics Criteria

The transport network is characterised by four different types of metrics. The metrics encompass diverse categories or directions that are impacted by modifications in the network. These also reflect the different interests or objectives that people, organisations, and governments might have for a network. These metrics are described as follows:

- **Agency metric:** The agency metrics assess the economic aspects associated with the maintenance and management of the network. This metric considers the perspectives and relevant expenses incurred by maintenance agencies and government departments of transportation.
- **User metric:** The user metric evaluates the costs or expenses incurred by the users of the network as a result of inadequate planning by the management or inadequate network conditions.
- **Environment metric:** The environmental metric attempts to quantify the carbon emissions that are being released directly and indirectly due to management of the network.
- **Safety metric:** The safety metric measures the performance of the components and the entire network to ensure that it is at or above acceptable standards.

Table 4.13 provides a succinct overview of the various categories that comprise each metric

and the various types of data that are necessary to establish these metrics. This helps to establish a foundation for a detailed output that considers the multiple aspects of maintaining and managing a transport network.

Metric	Metric Category	Data Type
Agency	Maintenance cost	Action cost, Element dimensions, Failure probability, Reconstruction cost
	Mobilising cost	Maintenance cost, Factor of direct cost
	Overweight cost	Distribution of overweight vehicles, IRI state, Cost
User	Delay cost	Maintenance duration, Increased travel time, Wage per vehicle user
	Vehicle Operating cost	Vehicle distribution, IRI state, Vehicle operating cost, Segment length
Environment	Emissions from actions	Action taken, Element dimensions, Carbon emissions from actions
	Emissions due to congestion	Link volume, Capacity, Free flow speed, Congested speed, Carbon emission at speed range
	Emissions due to detour	Carbon emission per vehicle type, Old and new route, Maintenance duration
Safety	Component safety	Beliefs over state
	Network safety	Segment reliability, Failure probability

Table 4.13: Type of data needed for calculating different aspects of the metric

A discount rate r of 0.02 has been used when assessing the monetary value of some metrics. This discount rate is used to account for the effects of inflation, interest rates, and opportunity costs in the overall environment to recognise changes in the value of money over time. In the following subsection, future expenses are converted into their present value, thereby enabling a more precise and comparable assessment of the total expenses throughout the life cycle (Lei et al., 2023). Each of the metrics is explained in detail, including their construction and considerations.

4.9.1. Agency Cost Metric

The agency metric evaluates the economic implications of the maintenance and management of a transport network, reflecting the financial implications for maintenance agencies and government departments responsible for transport infrastructure. The agency metric is divided into three main categories: maintenance cost, mobilising cost, and overweight cost as presented in Equation 4.3.

$$C_{agency}(t) = C_{agency}^{direct}(t) + C_{agency}^{indirect}(t) + C_{agency}^{mobilising}(t) + C_{agency}^{overweight}(t) \tag{4.3}$$

Where, $C_{agency}(t)$ is the total cost borne by the maintenance agency in time-step t; $C_{agency}^{direct}(t)$ and $C_{agency}^{mobilising}(t)$ is the immediate cost for taking the actions; $C_{agency}^{indirect}(t)$ is the accrued cost due to the current action causing probable failure and $C_{agency}^{overweight}(t)$ is the additional damage caused to the facilities due to vehicle overloads.

A. Direct Cost from Inspection & Maintenance Action

Direct cost refers to the cost associated with performing inspection and maintenance activities. This includes the direct monetary cost for labour, machinery, material and other relevant costs, as well as the direct monetary cost for other relevant costs. These costs are dependent on the type of action, the type of component, and the geometry of the component. The average monetary cost associated with each action is described in Table 4.14. The values for different costs has been retrieved from Lei et al. (2023), Saifullah et al. (2024), Chowdhury (2016). The cost of minor repair and major repair are 15% and 45% of the reconstruction cost which was obtained from Federal Highway Administration (2019).

Actions	Pavement cost ($\$/m^2$)	Deck cost ($\$/m^2$)
No inspection	0	0
Routine inspection	0.08	0.25
In-depth inspection	0.16	4
Do nothing	0	0
Minor repair	20	400
Major repair	75	1200
Reconstruction	350	2650

Table 4.14: Overview of the impact of every action on different rewards

This helps in formulating the direct cost equation which is as follows:

$$C_{agency}^{direct}(t) = \frac{1}{(1+r)^t} \sum_{i=1}^{n} (C^{m,a} + C^{i,a}) \times W_i \times L_i \tag{4.4}$$

where $C_{agency}^{direct}(t)$ is the direct cost in time-step t; the monetary cost of action on a specific component i is determined by the cost per sq m based on the length L_i and width W_i of the component.

B. Indirect Cost from Failure Probabilities

The indirect costs are associated with the potential untimely reconstruction cost due to the probable failure based on the action taken in time-step t. In this case, it is assumed that the untimely cost of reconstruction (C^{rec}) is the 1.5 times the actual cost. $P[X]_i$ for bridges is defined in Table 4.15 for each condition state (Saifullah et al., 2024). The probability distribution for failure for pavement is assumed to be $[0.001, 0.001, 0.005, 0.005, 0.01, 1.0]$ based on the CCI states.

Condition state	Deck Failure Probability ($s_{t+1} = failed$)
$s_t = 9$	0.001
$s_t = 8$	0.001
$s_t = 7$	0.005
$s_t = 6$	0.005
$s_t = 5$	0.005
$s_t = 4, ...$	0.01
$s_t = failed$	1.0

Table 4.15: Deck failure probability given the current deck state

Based on these failure probabilities the indirect cost equation is formulated as follows:

$$C_{agency}^{indirect}(t) = \frac{1}{(1+r)^t} \sum_{i=1}^{n} P[X]_i \times (C^{rec} \times k^{factor}) \times W_i \times L_i \tag{4.5}$$

Where, $C_{agency}^{indirect}(t)$ is the indirect cost in time-step t; $P[X]_i$ is the failure probability of component i in that time-step; $C^{rec} \times k^{factor}$ is the current failure reconstruction (1.5 here). The cost is calculated per sq.m. based on the length L_i and width W_i of the component.

C. Mobilisation Cost for Maintenance Action

Mobilisation cost in the context of infrastructure management refers to the cost of preparing the facilities, crew, and supplies to undertake the maintenance action. The cost of mobilising these factors depends on the project type, location, and other factors. The values in Table 4.16 gives a brief overview of the % estimates (k^{mob}) for every project.

Type of component	% for mobilisation
Pavement	4 - 5%
Bridges	6 - 7%

Table 4.16: % of infrastructure cost expected as mobilising cost

These costs were retrieved from the Missouri Department of Transportation (2019) and while they are specific for each project in this thesis, we assume that the cost is applied on the average value of the direct maintenance cost. It should be noted that the cost is applied only once if an action is taken within the designated time-step, in order to encourage the grouping of actions within that time-step.

$$C_{agency}^{mobilising}(t) = \frac{1}{(1+r)^t}\Big(\sum_{i=1}^{n}(C^{m,a}+C^{i,a})\times W_i \times L_i\Big) \times \frac{k^{mob}}{total\_actions} \tag{4.6}$$

Where, $C_{agency}^{mobilising}(t)$ is the mobilisation cost in time-step t; and k^{mob} is the factor of mobilisation cost.

D. Overweight Cost

In numerous Departments of Transport across the US, freight prioritisation is considered as an important agency goal, partly due to increased demand over the past several years. These federal agencies are responsible for issuing permits for oversized and overloaded vehicles due to the safety impact of such vehicles on the network and to consider the potential damage that is caused due to overloaded vehicles. This metric delineates the supplementary expenses incurred by the agency resulting from overload damage.

As the volume of trucks varies from network to network, and furthermore, a limited number of sections are equipped with Weigh-In-Motion sensors to monitor overweight vehicles.

Type of vehicle	Pavement cost ($/km)	Bridge cost ($/km)
Trucks	1.032	0.0135
XL Trucks	1.2	0.0468

Table 4.17: Overweight cost for vehicle type: Truck and XL Truck

Due to the site-specific nature of the data, empirical data from a South Carolina network is assumed to be the basis. It defines the Average Annual Daily Truck Traffic (AADTT), the % of overloaded vehicles and damage caused in $\$/km$ (Lou et al., 2016).

$$C_{agency}^{overweight}(t) = \frac{1}{(1+r)^t} \sum_{i=1}^{l} \sum_{j=1}^{v} AADTT_{i,j} \times C^{overload} \times k_i^{iri\_factor} \times L_i \tag{4.7}$$

Where, $C_{agency}^{overweight}(t)$ is the additional cost due to overloaded vehicles in time-step t; and $AADTT_{i,j}$ is the Truck traffic on segment i; $C^{overload}$ is the overload cost based on Table 4.17. Lastly, k^{iri_factor} takes into account the IRI feature in calculating the cost.

4.9.2. User Cost Metric

The User Cost Metric is a comprehensive measure used to evaluate the economic impact of network conditions on its users. It is a vital indicator used to assess the financial burden on users. It is defined as the sum of costs caused by delays and increased vehicle operating expenses, formulated as:

$$C_{user}(t) = C_{user}^{delay}(t) + C_{user}^{operating}(t) \tag{4.8}$$

where $C_{user}(t)$ is the approximate cost incurred by the users of the network in time-step t, $C_{user}^{delay}(t)$ is economic or productivity loss due to maintenance actions and $C_{user}^{operating}(t)$ is the direct impact of poor network conditions leading to higher vehicle operating costs.

A. Delay Cost due to Maintenance Actions

Inefficiently planned actions can lead to serious delays throughout the network, leading to increased travel times. This is calculated based on the traffic model discussed in the preceding sections. This is directly represented as the productivity loss due to delay and has

been formulated in the equation below. The equation has been adopted from Saifullah et al. (2024). It is important to note that the hourly wage for trucks is $29.65 and the hourly wage for passenger cars is $21.89. These values are based on the 2019 data reported by Federal Highway Administration (2023d).

$$C_{user}^{delay}(t) = \frac{1}{(1+r)^t} \sum_{i=1}^{l} ATT_i \times d_i^{mnt} \times L_i \times ((W_{car} \times (1 - truck)) + (W_{truck} \times truck)) \quad (4.9)$$

where $C_{user}^{delay}(t)$ is the delay cost for the time-step t; ATT_i is the additional travel time on the link l; d_i^{mnt} is the maintenance duration and W_{truck} is the average hourly wage for trucks and W_{car} is the average hourly wage for passenger cars.

B. Vehicle Operating Cost based on Network Condition

Vehicle Operating Cost (VOC) is an important parameter in road user effects (RUE) for cost-benefit analysis of maintenance actions. The main components of VOC are fuel, oil, tyres and repairs and maintenance of the vehicle (Australian Transport Assessment and Planning, 2021). For this thesis the vehicle operating cost is a function of the International Roughness Index and was calculated using a program called RED MODEL Version 3.2. It is an Excel file that calculates the factor a_0, a_1, a_2 as shown in Table 4.18 for the VOC function based on the vehicle type and factors like terrain, elevations, etc. This is based on SSTAP method of cost estimation (Sub-Saharan Africa Transport Policy Program, 2006).

Type of vehicle	a_0	a_1	a_2
Motorcycle	0.163572	0.006988	0.000385
Car	0.116755	-0.000753	0.000381
Bus	0.16901	0.006576	0.000325
Truck	0.163572	0.006988	0.000385
XL Truck	0.572942	0.015662	0.001402

Table 4.18: IRI based VOC factor parameters

The VOC is calculated as $VOC = a_0 + a_1 \times IRI + a_2 \times IRI^2$, where IRI is the roughness index in in/mile. Thus, as the IRI value increases, the value of the Vehicle operating cost will also rise. The complete formulation of the cost is as follows:

$$C_{user}^{operating}(t) = \frac{1}{(1+r)^t} \sum_{i=1}^{l} \sum_{j=1}^{v} AADT_{i,j} \times VOC_j \times L_i \tag{4.10}$$

where $C_{user}^{operating}(t)$ is the excess operating cost for time-step t; $AADT_{i,j}$ is the traffic count of vehicle j on link i; VOC_j is the Operating Cost values calculated based on Table 4.18 and L_i is the length of the link.

4.9.3. Environmental Metric for CO_2 Emissions

Assessing the environmental impact of deteriorating transport networks requires evaluating various metrics, including energy usage, water pollution, soil degradation, and carbon emissions. Nonetheless, CO2 emissions are given prominence as the primary environmental indicator owing to their significant impact on global warming and social sustainability (Lei et al., 2023). This study identifies the carbon footprint generated by maintenance activities, detours, and congestion as the primary cause of environmental impact. The significance of the degree of deterioration in various components cannot be overstated; minor damage leads to CO2 emissions from materials and repair activities, whereas severe damage necessitates traffic detours and creates congestion, significantly elevating CO2 emissions.

$$C_{env}(t) = C_{env}^{action}(t) + C_{env}^{detour}(t) + C_{env}^{congestion}(t) \tag{4.11}$$

where $C_{env}(t)$ is the carbon emissions from the network in time-step t, $C_{env}^{action}(t)$ is the carbon emissions from maintenance actions; $C_{env}^{detour}(t)$ is the carbon emissions from detour caused by maintenance actions and $C_{env}^{congestion}(t)$ is the carbon emissions caused due to congestion.

A. Carbon Emissions caused by Maintenance Actions

Material consumption for maintenance actions is a significant contributor to carbon emissions. It also includes the carbon emissions from performing the activities. Table 4.19 gives the emissions per action. The emissions for reconstruction is retrieved from Xu and Guo (2022) while the emissions for minor and major repairs were abstracted from Gardiner (2022) and Wells (1995).

Type of action	Indirect emission factor	CO_2 emissions (g/m^2)
Inspection	0	0
Do nothing	0	0
Minor repair	0.15	6000
Major repair	0.45	15120
Reconstruction	1.0	33600

Table 4.19: CO_2 emissions for different inspection and maintenance action in gm/m_2

The carbon emissions released from the maintenance actions includes the direct emissions from actions in the given time-step as well as the probable emissions due to failure caused reconstruction. The formulation is as follows:

$$C_{env}^{action}(t) = \sum_{i=1}^{n} E_i^{action} \times W_i \times Ł_i + \frac{1}{(1+r)^t} \sum_{i=1}^{n} E_i^{action} \times W_i \times Ł_i \times P[X]_i \tag{4.12}$$

where $C_{env}^{action}(t)$ is the carbon emissions in time-step t; E_i^{action} is the carbon emissions from the action; $P[X]_i$ is the probability of failure of the component and $W_i, Ł_i$ are the dimensions of the component i.

B. Carbon Emissions caused by Detour due to Reduced capacity

One of the indirect carbon emissions sources resulting due to the maintenance actions are emissions due to detour. As maintenance activities are planned in certain segments the capacity of the segment and hear by the edge reduces significantly. Based on the max flow min cut theorem by Ford and Fulkerson, which states that *'the max flow of a graph is equal to the minimum cut'* the minimum cut sum is the capacity of the edge. Detours cause extra distance to be covered and emissions from operating which can be seen in Table 4.20. This data was abstracted from Lei et al. (2023).

Type of vehicle	CO_2 emissions (g/km)
Motorcycle	80.5
Passenger Car	161
Bus	644
Truck	604.9
XL Truck	1236.1

Table 4.20: CO_2 emissions per kilometre for vehicles considered in the network in g/km

Based on the traffic model assignment and carbon emissions model, the formulation for emissions from detour is as follows:

$$C_{env}^{detour}(t) = \sum_{i=1}^{tr} \sum_{j=1}^{v} ADT_{i,j}(t) \times E_j \times (D_i - O_i) \times d \tag{4.13}$$

where, $C_{env}^{detour}(t)$ is the carbon emissions from detour in time-step t; $ADT_{i,j}(t)$ is average trips between nodes for each vehicle type j; E_j is the emissions for each vehicle type; $(D_i - O_i)$ is the difference in distance between new and old route and d is the maintenance duration for segments in the old route.

C. Carbon Emissions caused by Congestion

Various studies have shown that longer travel times and reduced speeds are effects of congestion that increase carbon emissions. Whenever congestion brings the average vehicles peed below 70 kmph, there is a negative net impact on CO2 emissions. Vehicles spend more time on the road, which results in higher CO2 emissions (Barth and Boriboonsomsin, 2008). As more maintenance activities are planned keeping in mind that the traffic is rising it is necessary to adapt the detour paths to reduce the congestion on routes.

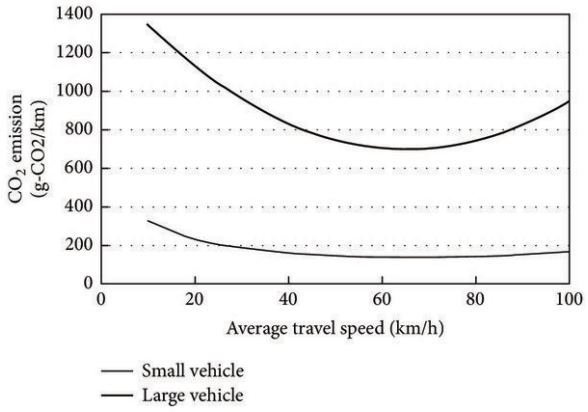


Figure 4.6: Relationship between carbon emissions and vehicle speed (Barth and Boriboonsomsin, 2008)

Based on the travel time, speed and maximum flow calculated in the traffic assignment model the level of service for the link is determined. Based on this the following equation is formulated:

$$C_{env}^{congestion}(t) = \sum_{i=1}^{l} \sum_{j=1}^{v} E_j^{ti} \times AADT_i \times L_i \tag{4.14}$$

where, $C_{env}^{congestion}(t)$ is the carbon emissions from congestion in time-step t; E_j^{ti} is the emission at the given speed for the type of vehicle; $AADT_i$ is the annual average daily traffic on link i and L_i is the length of the link.

4.9.4. Safety Metric for Component and Network Safety

In the case of a probabilistic environment, the safety metrics defines the component wise and overall network level structural condition. These metrics are crucial for understanding the current state of the entire network, allowing for proactive maintenance and risk management.

$$C_{safety}(t) = C_{safety}^{comp}(t) + C_{safety}^{network}(t) \tag{4.15}$$

where $C_{safety}(t)$ is the safety condition in time-step t; $C_{safety}^{comp}(t)$ is the immediate cost for taking the actions and $C_{safety}^{network}(t)$ is the accrued cost due to the current action causing probable failure.

A. Component Level Safety Metric

Structural condition levels are determined through a combination of visual inspections, assessment indicators, and control limits (Abdallah et al., 2022). This comprehensive approach accurately reflects the structure's current operational state and enables the prediction of future condition levels using the methods described in this research. At the component level, structural health is defined by the level of NBI structural condition that ranges from 0 to 9. Thus if the overall value is high the Safety metric indicates that the system is in a poor state. The safety metric for components is defined as follows:

$$C_{safety}^{comp}(t) = \sum_{i=1}^{n} \prod_{i=1}^{s} belief_{i,s}(t) \tag{4.16}$$

where, t is defines the time-step of the problem; C_{safety} is the safety level of all the segments in the network; and $belief_{i,s}(t)$ is the belief of being in a state s for component i.

B. Network Level Safety Metric

When defining the network level safety we essentially define the reliability. Reliability is the probability that a component or system will perform satisfactorily given the horizon. As explained in Suo et al. (2012) systems can be defined as series, parallel or combination system which is a two-state event, i.e. Operational or failed. As mentioned in previous sections we assume that the segments are independent of each other. We calculate the failure probability of the entire system by:

$$C_{safety}^{network}(t) = 1 - \begin{cases} \prod_{i=1}^{s}(1 - P[X]_i) & , \quad \text{if } s \text{ in series system} \\ (1 - P[X]_i)^n & , \quad \text{else} \end{cases} \tag{4.17}$$

where $P[X]$ is the probability of failure of an edge (segments in edges are a series system) for time-step t.

4.10. Multi-attribute Utility Model

A transport network impacts the economy, society and environment in many ways and each of them is not in the same unit or system of measurement. In such cases, when we have to deal with multiple objectives on different scales, we can use a multi-attribute utility model. The multi-attribute utility model used in this thesis was first suggested by Dong et al. (2015) to quantify a multi-criteria optimisation problem for highway bridges. As a complex real life problem it is not possible to quantify everything in terms of a single unit, in such case a utility model is used.

4.10.1. Decision-makers Risk Attitude

Risk attitude parameter (γ) is used to quantify the risk attitude of a policymaker in the context of maintenance and inspection decisions. Optimal maintenance plans for the life cycle are significantly influenced by the decision makers' time preference and perception of risks, which manifest themselves as delay and probability discounting. Specifically, delay

discounting typically postpones maintenance actions, whereas probability discounting may slightly hasten them (Cheng et al., 2020). The value can be $\gamma > 0$ for risk-averse behaviour and $\gamma < 0$ for risk-accepting behaviour.

- $\gamma > 0$ exhibits a risk-averse attitude. Risk-averse policies prefer to avoid risk and tend to make decisions that minimise potential losses. A risk-averse policymaker might delay maintenance actions until they are more certain that it is necessary to avoid unnecessary costs.
- $\gamma < 0$ exhibits risk-accepting policies are more willing to take risks and may act proactively in uncertain situations. In the context of maintenance, a risk-accepting policymaker might decide to conduct maintenance earlier than strictly necessary. This could be to avoid potential future failures even if it means higher immediate costs or maintenance actions that may turn out to be unnecessary.

4.10.2. Utility functions

Utility is defined as a measure of value (or desirability) to the decision maker. Utility theory provides a framework that can measure, combine, and compare utility values consistently (Ang and Tang, 1984). The following equations calculate the utility for each metric. The max values are the accepted or maximum available limits. This utility function normalises the all the metric values between 0 and 1.

$$u_{agency}(t) = \frac{1}{1-\exp(-\gamma)}\left[1-\exp\left(-\gamma\frac{C^{max}_{agency}-C_{agency}(t)}{C^{max}_{agency}-C^{min}_{agency}}\right)\right] \tag{4.18}$$

$$u_{user}(t) = \frac{1}{1-\exp(-\gamma)}\left[1-\exp\left(-\gamma\frac{C^{max}_{user}-C_{user}(t)}{C^{max}_{user}-C^{min}_{user}}\right)\right] \tag{4.19}$$

$$u_{env}(t) = \frac{1}{1-\exp(-\gamma)}\left[1-\exp\left(-\gamma\frac{C^{max}_{env}-C_{env}(t)}{C^{max}_{env}-C^{min}_{env}}\right)\right] \tag{4.20}$$

$$u_{safety}(t) = \frac{1}{1-\exp(-\gamma)}\left[1-\exp\left(-\gamma\frac{C^{max}_{safety}-C_{safety}(t)}{C^{max}_{safety}-C^{min}_{safety}}\right)\right] \tag{4.21}$$

where u_{agency}, u_{user}, u_{env} and u_{safety} are utility models; C^{max}_{metric} is the max allowed or expected value of the metric and C^{min}_{metric} is the minimum value which is usually set to 0.

The maximum cost is the amount the decision maker is willing to allocate or bear. Based on this context, the maximum values for agency can be the budget assigned, the maximum values for carbon emissions can be in line with the Climate agreements, and the maximum user values can be dependent on economic growth values. These values need to be clearly defined and fine-tuned based on several social, economic, and climate models to reflect the position of all the stakeholders correctly. It is essential that correct max values are picked as it directly impacts the utility of the metric. If the max value is too large, the utility values will be very closely packed and close to 0. This may be a good starting point, which can be updated with subsequent runs. On the contrary, if the maximum value is too low, the utility will become negative, resulting in a significant impact on the ultimate function.

4.11. Reward function

Based on the utility functions in 4.10.2 the reward function is defined using weights predefined for each metric. The main aim of using an utility based model in the reward function is to improve stability and convergence as noted by Lei et al. (2023) where use of metric values in different units when used directly can cause instability in the model and difficulties in convergence. The aim is to maximise this functions output across the entire horizon.

$$Reward(t) = w_{agency}u_{agency}(t) + w_{user}u_{user}(t) + w_{env}u_{env}(t) + w_{safety}u_{safety}(t) \tag{4.22}$$

where w_{agency}, w_{user}, w_{env} and w_{safety} represents the weight for each metric; and u_{agency}, u_{user}, u_{env} and u_{safety} are utility models for each metric. Each utility function u transforms the raw metric values into a standardised utility value, allowing for consistent and comparable contributions to the overall reward. By appropriately selecting the weights w, the model balances the importance of each metric according to the desired outcomes, ensuring a comprehensive and holistic optimisation strategy.

5

Results and Evaluation

5.1. Scenario Setup

This chapter demonstrates and discusses all scenarios relevant to this research. As mentioned in the previous chapters, the purpose of these simulations/runs is to demonstrate the impact of dynamic influences on objectives and policy. There are three primary cases, each of which is built up by adding constraints and features to the previous case.

The following sections describe the overview and setup parameters for each scenario. Based on the above setup parameters, a few baseline runs were done, and the result is presented. The baselines used for this research are failure-replacement and time-based maintenance (TBM).

- **'Do Nothing':** No maintenance actions are taken, allowing the segments to deteriorate naturally.
- **'Failure Replace':** Components are only replaced upon failure, with no preventive maintenance.
- **Time-based maintenance:** Actions like minor repairs are performed every five years and inspecting every two years.

These policies provide a range of maintenance strategies from minimal to proactive interventions. Subsequently, a few experimental runs are conducted for aspects like changing cost of action, inclusion of budget, change in preferences, etc., followed by an initial evaluation of the outcomes and policies generated for each scenario.

5.2. Scenario 1

A simplified example is constructed for the first scenario. The objective of this case is to understand the interaction between components, maintenance actions, and the impact of actions on objectives and reward function. Instead of considering a system of components, five independent pavement segments are considered. Each segment has a length of 6 km, with two lanes of 3.5 m each. The set of actions available to the agent in the scenario are 'Do nothing', 'Minor repair', 'Major repair', 'Reconstruction', 'Inspection'. The inspection action in this instance is an in-depth inspection, and the agent is limited to taking only one action per time step. The objective of the algorithm is to minimise the agency cost, i.e. the overall economic cost for up keeping the components.

In the first time step, the state of the components is always intact. As the environment is modelled as a POMDP, the initial belief of the agent may be either certain or ambiguous. The deterioration model used for this scenario is stationary as the main goal is to study the impact of actions on cost and beliefs. Prior to setting up the various runs in the scenario, the transitions based on deterioration and inspection model were carefully studied to ascertain their impact on the agent's state and beliefs. During the initial runs, it was noticed that the deterioration model was not aggressive enough, which resulted in the component remaining in the same state for the entire time horizon as seen in Figure 5.1.

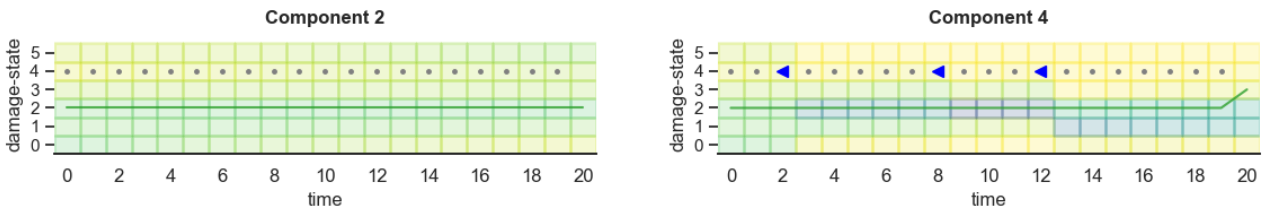


Figure 5.1: Policy generated with a non-aggressive deterioration model leading to only 'Do nothing' or 'Inspection' actions

To overcome this issue, an aggressive deterioration model was employed, as depicted in the

matrix below. Furthermore, the number of years were increased by simulating the years it took for the belief to move from the intact to the failed state. The time horizon was extended to 40 years in this case. The initial run also revealed that given the inexpensive nature of the inspection action, the agent continued to suggest only inspection action for all time steps when other actions were not taken. An example of such a policy can be seen in Figure C.1 in Appendix C.

$$P(s_t \mid s_{t+1}, 'Do\ nothing') = \begin{bmatrix} 0.7075 & 0.2904 & 0.0021 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.7760 & 0.2240 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.8936 & 0.1059 & 0.0005 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.8188 & 0.1807 & 0.0005 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.8344 & 0.1656 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{bmatrix}$$

In the subsequent subsections, two experimental runs using this scenario are presented to evaluate the performance and decision-making of the agent under different conditions. The experiments will focus on the efficacy of the actions, the impact of a hard boundary constraint, and the overall impact on the objectives and reward function.

5.2.1. Baseline for the Scenario

Before defining the experiments, it is important to establish some baseline checks. These aid in defining the benchmarks that will be utilised to evaluate the policies obtained from the various runs. Below are some benchmarks derived from testing the model with baseline policies like the 'Do Nothing', 'Failure replace' and 'Time-Based Maintenance' policy. When simulated for 10,000 runs, the mean, and standard deviations for failure-replacement and TBM policy are calculated.

The metrics include total cost, standard error, direct cost, and indirect cost, chosen to capture both the economic and operational impacts of each policy. From Table 5.1, we observe that the 'Failure replace' policy incurs higher total costs due to high cost of reconstructing failed components, while the TBM policy balances costs more effectively by preventing failures through regular maintenance.

Characteristic	Failure replace policy	TBM policy
Total cost (mean)	1,044,346,506	215,505,941.5
Std Error	15,250	654,008.2
Direct cost (mean)	73,500,000	102,900,000
Indirect cost (mean)	969,964,506	64,667,941.5

Table 5.1: Baseline values for Scenario 1 for failure-replacement and time-based maintenance policy considering there is no hard constraint in the environment

5.2.2. Model Parameters

For scenario 1 Double Deep Q-Network (DDQN) algorithm was used for determining the policy. To identify the best possible combination for the hyperparameters, several runs were done and values were updated to see the expected behaviour. A batch size of 64 was selected as it strikes a balance between computational efficiency and training stability. A discount factor of 0.97 indicates that future rewards will be slightly discounted, but still play a significant role in the value estimation. Lastly, the target network reset is set to a lower value to enable the algorithm to pursue feasible targets. The model parameters are as follows:

Parameter	Value
Algorithm	DDQN
Memory capacity	8_000
Batch size	64
Learning rate	0.001
Discount factor	0.97
Optimiser	'Adam'
Target network reset	30
Exploration strategy	Epsilon greedy
Episodes	15_000

Table 5.2: Model hyperparameters for experiment 1 listing the parameters setup for DDQN

5.2.3. Experiment Run 1

The initial experiment conducted on the scenario using the previously specified parameters. This run was conducted to determine the type of policy the algorithm will develop when there are no constraints, and to determine how much cost is attributed to each cost component in the cost model. The initial belief for this run was set to be certain, assuming

complete knowledge of the intact state. The algorithm began with a random policy and gradually learned an optimal policy over 15,000 episodes. One of the realisations of the policy is presented in Figure 5.2.

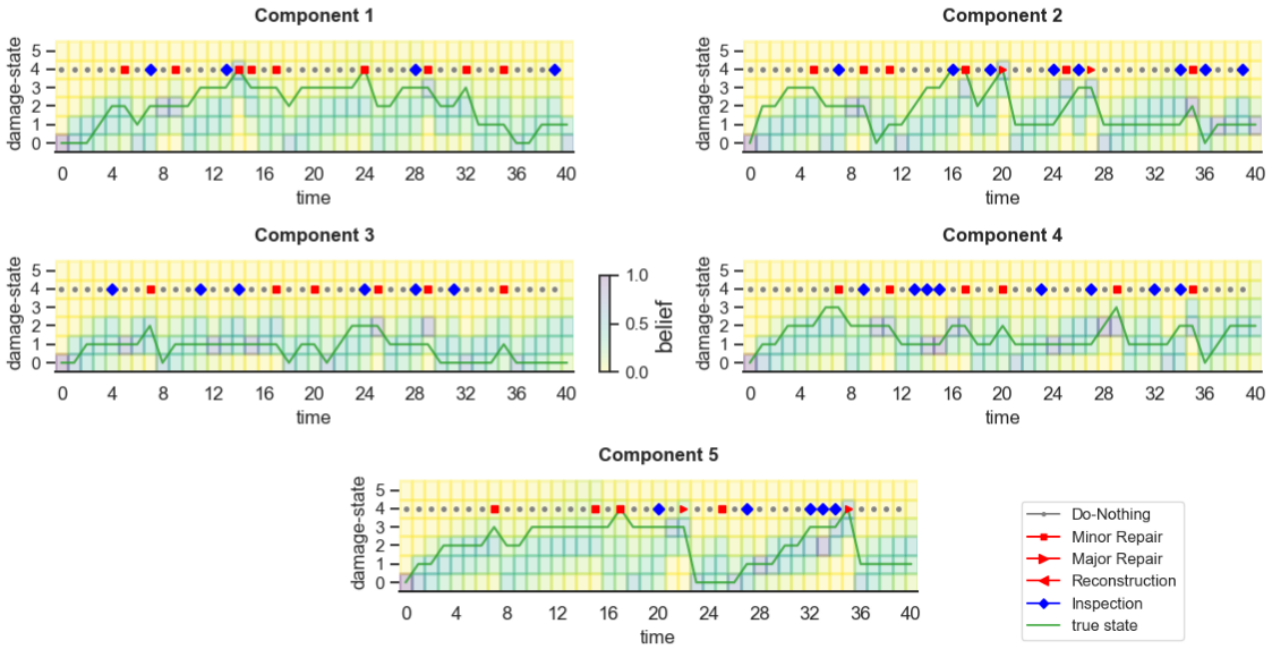


Figure 5.2: Policy realisation for scenario 1 with no constraints and only one objective to minimise i.e. the total agency cost

Initially, the policy favoured frequent inspections to gather information about the pavement segments' conditions. Over time, with continuous explorations, the agent started recommending more 'Minor repairs' to prevent high future costs. The frequency of 'Major repair' actions remained low, indicating that the algorithm found it cost-effective to maintain segments through less intensive repairs. This can also be attributed to the high rate of deterioration. 'Reconstruction' was never mentioned in any of the policies, possibly due to the high expense of replacement. The total agency cost of this realised policy was 134,545,156, which is significantly lower than the baseline runs for both failure replace and time-based maintenance. Compared to the time-based maintenance policy, the economic spending for repairs was much lower for DDQN, with an approximately 25% reduction in direct and mobilisation cost. The algorithm also avoids conditions where the damage state is higher than 4 (Poor for Critical Condition Index). The interaction and impact of a change in one cost on the other factors can be seen in the Figure 5.3. The direct and mobilisation

costs decreased at the same rate, mostly due to the dependence of mobilisation costs on the direct cost. The reduction in expenditure on maintenance activities leads to a rapid rise in indirect expenses. From this run, it can be deciphered that:

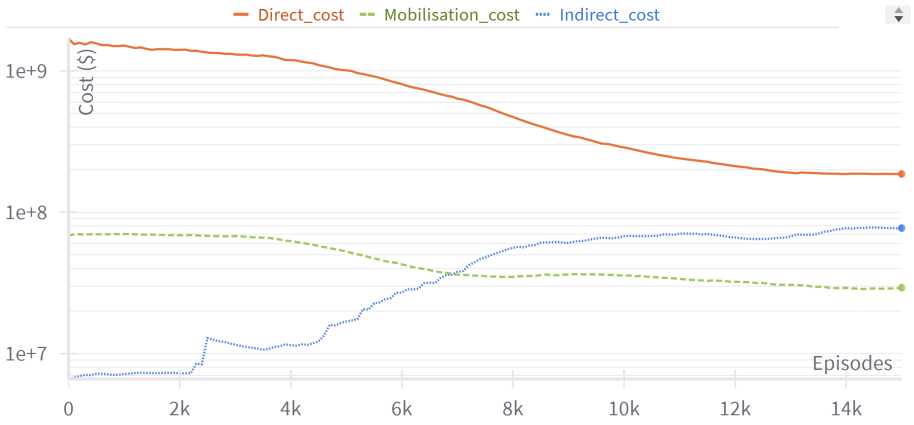


Figure 5.3: Interaction between various costs

- Do nothing actions have low immediate cost but have higher long-term deterioration cost.
- Minor repair have moderate cost with a significant impact on extending the segment's life and when compared to Major repair it becomes a more cost effective alternative.
- Reconstruction action has the highest cost and is mostly reserved for segments in near-failure states.

5.2.4. Experiment Run 2

In this experiment, the impact of budget on the policies is studied, and the previous experiment is expanded upon. Every Department of Transportation or maintenance agency has a designated amount of budget allocated for repair and maintenance initiatives. It would be interesting to look at the impact of budget allocation on different costs and policies generated.

Using the results from the first run, we analyse the mean direct and mobilisation cost values. These values represent the maximum average cost that the agent necessitates to

avert component failures. We use this value as the maximum budget for this run as a hard constraint. One approach to managing this is to truncate the episode, but this can lead to issues such as limited exploration and biased return estimates. Instead, for this run, a budget of $ 120,000,000 was established. Any unused funds from this budget are carried forward to the next budget value. If the cost of actions exceeds the budget, the action is adjusted to "do nothing" resulting in the component remaining in the same state or worsening based on the deterioration model. This increases the risk associated with failure and reconstruction in the form of indirect costs.

The realisation of the policy with budget constraint is presented in Figure 5.4 with 70% of the budget allocated.

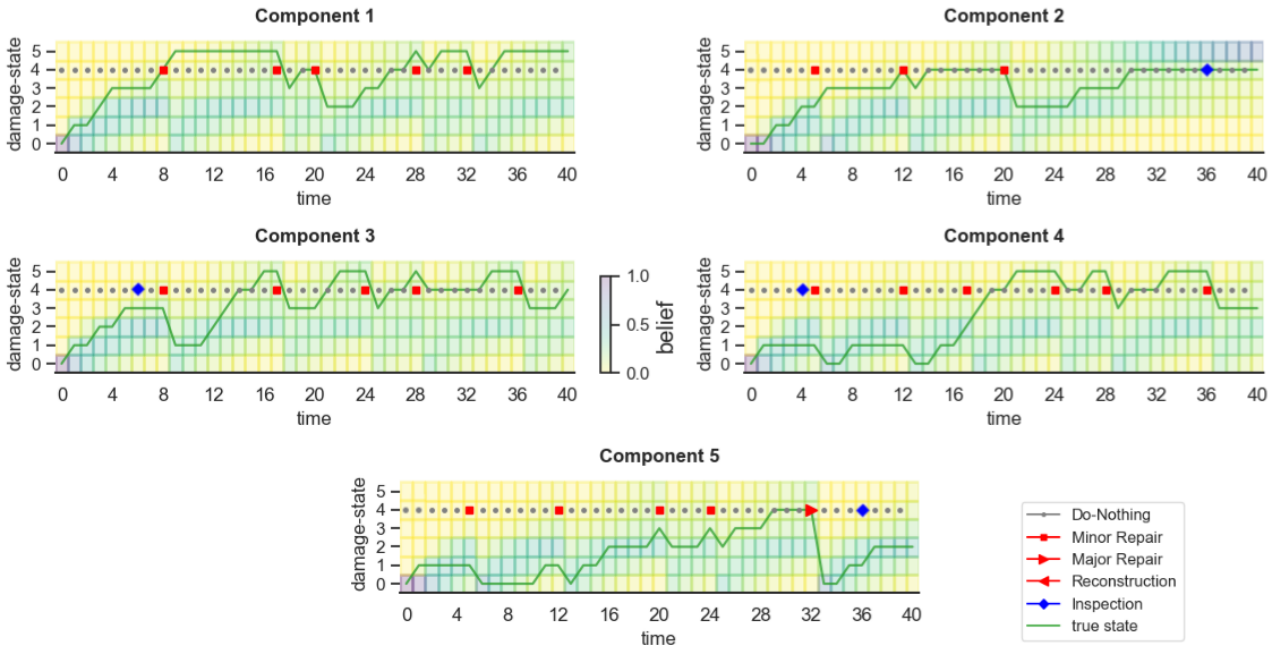


Figure 5.4: Policy realisation for scenario 1 with budget constraints (70% budget allocated) and only one objective to minimise i.e. the total agency cost

As mentioned earlier, there are various instances when the budget is decreased, and the decision makers would like to see the impact of the reduced budget on different costs. Based on these figures, it is evident that there is a noticeable decrease and increased sparsity of maintenance actions. In the experiment with 70% budget allocation, the system often reaches the failure state. The agent does face issues with maintaining the system towards the end of the time horizon, but this might be likely due to lack of a terminal cost. Interestingly, there

are a few inspection and 'major repair' actions in the policies generated. The mean of all the costs in different budget explorations can be seen in Table 5.3.

Characteristics	100% Budget	90% Budget	80% Budget	70% Budget
Total cost	126,203,376	156,892,590	204,622,034	320,299,872
% change in cost	-	24%	62%	153%
Direct cost	81,628,397	71,061,080	64,575,964	56,057,064
% change in cost	-	-13.2%	-20.8%	-31.3%
Mobilisation cost	14,227,012	12,817,600	11,231,473	8,897,055
% change in cost	-	-10%	-21%	-37.5%
Indirect cost	30,347,967	73,013,910	128,814,597	255,345,753
% change in cost	-	140%	324%	741%

Table 5.3: Exploration of the impact of various budget allocations on different costs impacting the agencies

The table shows the percentage and absolute increase and decrease in various costs as the allocated budget is reduced. It can be noted that when the allocated budget reaches 80% the indirect risk increases three fold and then continues to rise steeply at 70% allocation that could be catastrophic for the network.

5.2.5. Initial Reflection

During the run, it was clear that the type of policy generated by the algorithm is very sensitive to environment values, such as costs or factors. It was also observed that the agent would persist in executing inspection actions based on the specified inspection action cost of $0.20, as it was comparatively inexpensive to execute. Furthermore, it avoided major repairs and reconstruction for the entire time horizon. This raises questions regarding the nature of the repair and inspection cost values and recommends the necessity of additional investigation or addition of penalty measures to prevent the agent from engaging in unnecessary actions.

In certain runs, it was noteworthy that the cost function views the network as a system rather than individual components, resulting in a reduction of the total cost. This phenomenon can lead the agent to overlook a component at times. It is possible to improve the agent's quality by adding additional penalties that require all components to perform at a certain state.

From the second experiment with varying budgets a number of interesting studies were

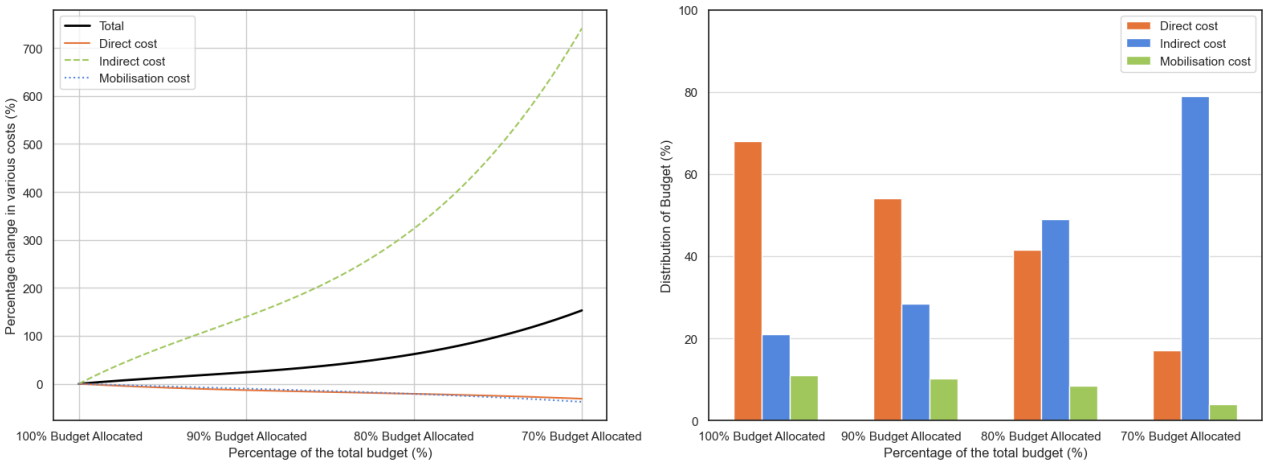


Figure 5.5: (a) Change in direct, indirect and mobilising cost w.r.t. change in budget allocated for the maintenance ; (b) Distribution of each cost in the total agency cost at different budget levels

done. In Figure 5.5 we see the interaction between the direct cost and indirect cost. The relationship between them is inversely proportional, but the magnitude differs. The second illustration illustrates how the reduction in expenditure leads to an increase in indirect expenses.

5.3. Scenario 2

The second scenario builds upon the learnings and insights gained from the first scenario. It aims to deepen the understanding of the interactions between components within a network and how the condition of one component can affect others and the overall system objective. In this expanded scenario, the network includes nine components, comprising both pavements and bridge segments. Unlike the previous scenario where components were isolated, here the segments are interconnected, forming edges that connect nodes or places of interest, introducing dependencies among them. Figure 5.6 and Table 5.4 provide detailed information regarding the layout and characteristics of the network. The red markings in the figure denotes the nodes and the blue markings are the start and end points for the segments.

The set of available actions for the agent includes 'Do nothing', 'Minor repair', 'Major repair', 'Reconstruction', 'No Inspection', 'Routine Inspection', and 'In-depth Inspection'. The agent can take two actions per time step: one maintenance action and one inspection

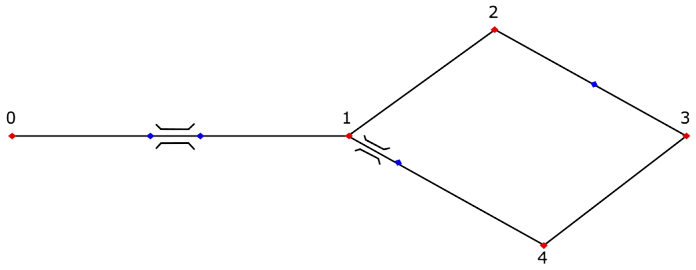


Figure 5.6: Proposed transport network for scenario 2 which is part of the larger network studied on this thesis

Link	Segment	Type	Length (km)	No. of lanes	Capacity (vehicles/day)
0	0	Pavement	6.2	2	1872.49
	1	Bridge	1.14	2	2031.81
	2	Pavement	8.1	2	2067.62
1	3	Pavement	6.7	2	2477.049
2	4	Pavement	8.86	2	2095.84
	5	Bridge	2.54	2	1781.81
3	6	Pavement	4.05	2	1998.39
4	7	Pavement	4.8	2	1735.65
	8	Pavement	3.0	2	1798.39

Table 5.4: Attributes for every link of the network for scenario 2

action. Therefore, the total possible action combinations are 8. Similar to the first scenario, all components start in an intact state at the first time step. The initial belief of the agent about the state of the components is certain in all the cases. The deterioration and inspection models applied in this scenario are derived from the transition models presented in Chapter 4. The primary goal of this simulation is to minimise both the agency cost and the user cost over an episode duration of 20 time steps. To assess the impact of maintenance actions on users, a traffic model is employed, with trip details outlined in Table 5.5.

Nodes	0	1	2	3	4
0	0	3000	1000	1500	1500
1	0	0	4500	2500	6000
2	0	0	0	0	9000
3	0	0	0	0	4000
4	0	0	0	0	0

Table 5.5: Matrix of trips between each node pair (vehicles/day) for scenario 2

The subsequent subsections present two experimental runs using this scenario to evaluate the performance and decision-making of the agent under different conditions. These experiments will focus on the impact on competing objectives and the reward function generated by the algorithm. Furthermore, statistical observations derived from changes in the cost distribution resulting from budget changes will be analysed to provide more comprehensive insights.

5.3.1. Baseline for the Scenario

Two benchmarks were created in this case where both agency and user cost determined the total cost of the policy. When simulated for 10,000 runs, the mean, and standard deviations for failure-replacement and TBM policy are calculated.

Characteristic	Failure replace policy	TBM policy
Total cost (mean)	23,138,868,439.3	2,335,123,860.3
Std Error	50,036,880.3	10,459,011.8
Agency cost (mean)	541,790,932.3	376,983,765.8
Direct cost (mean)	106,531,704.9	71,320,594.7
Indirect cost (mean)	337,441,211.3	218,290,468.6
User cost (mean)	22,597,077,506.9	1,958,140,094.4
Delay cost (mean)	21,630,438,596.6	1,074,951,888.7
VOP cost (mean)	966,638,910.3	883,188,205.6

Table 5.6: Baseline values for Scenario 2 for failure-replacement and time-based maintenance policy considering both agency and user costs

The metrics include total cost, standard error, agency cost, and user cost, which are chosen to capture both the economic and user impacts of each policy. From Table 5.6, we can clearly see the impact of repair and maintenance on both the costs. The largest impact is evident in the delay cost. As the number of vehicles on the network continues to increase, and the condition of the network continues to deteriorate, the travel time for users will also continue to grow.

5.3.2. Model Parameters

For scenario 2 Deep Centralised Multi-agent Actor Critic (DCMAC) algorithm was used for determining the policy. The experiments use a large memory capacity due to the vast state

space in the problem. We use standard batch size and optimiser for these runs. A slightly higher gamma compared to typical single-agent setup is used to reflect the need for agents to consider the long-term consequences of their joint actions. Lastly a low learning rate with help in determining a stable learning policy. The model parameters are as follows:

Parameter	Value
Algorithm	DCMAC
Memory capacity	25_000
Batch size	64
Discount factor	0.98
Actor Learning rate	0.0001
Critic Learning rate	0.005
Optimiser	'Adam'
Exploration strategy	Epsilon greedy
Episodes	21_000

Table 5.7: Model Parameters for experiment 2 listing the parameters setup for DCMAC

5.3.3. Experiment Run 1

The initial test scenario focused on evaluating the cost of maintenance under the condition of a single objective. This run mirrors earlier testing phases, as the presence of only one objective causes the infrastructure segments to function independently rather than as a cohesive system. The primary aim of this test was to establish a baseline for maintenance expenses, which can be compared against results from future scenarios that incorporate additional parameters.

The lack of budget constraints in this initial run allowed for an unrestricted evaluation of maintenance activities, ensuring that the true cost of maintenance could be accurately measured under ideal conditions. The test provides a clear understanding of the maintenance expenses required to achieve optimal performance for each segment when treated independently, without imposing financial limitations. Results from this run will serve as a reference point for future tests where multiple objectives and parameters will be introduced. This comparative analysis will help understand the impact of additional complexity on maintenance expenses and system performance. This initial single-objective, no-budget constraint scenario serves as a solid foundation for subsequent analyses, offering more

in-depth insights into the dynamism of infrastructure upkeep under various scenarios and objectives. The cost distribution histogram from this run can be seen in Figure 5.7.

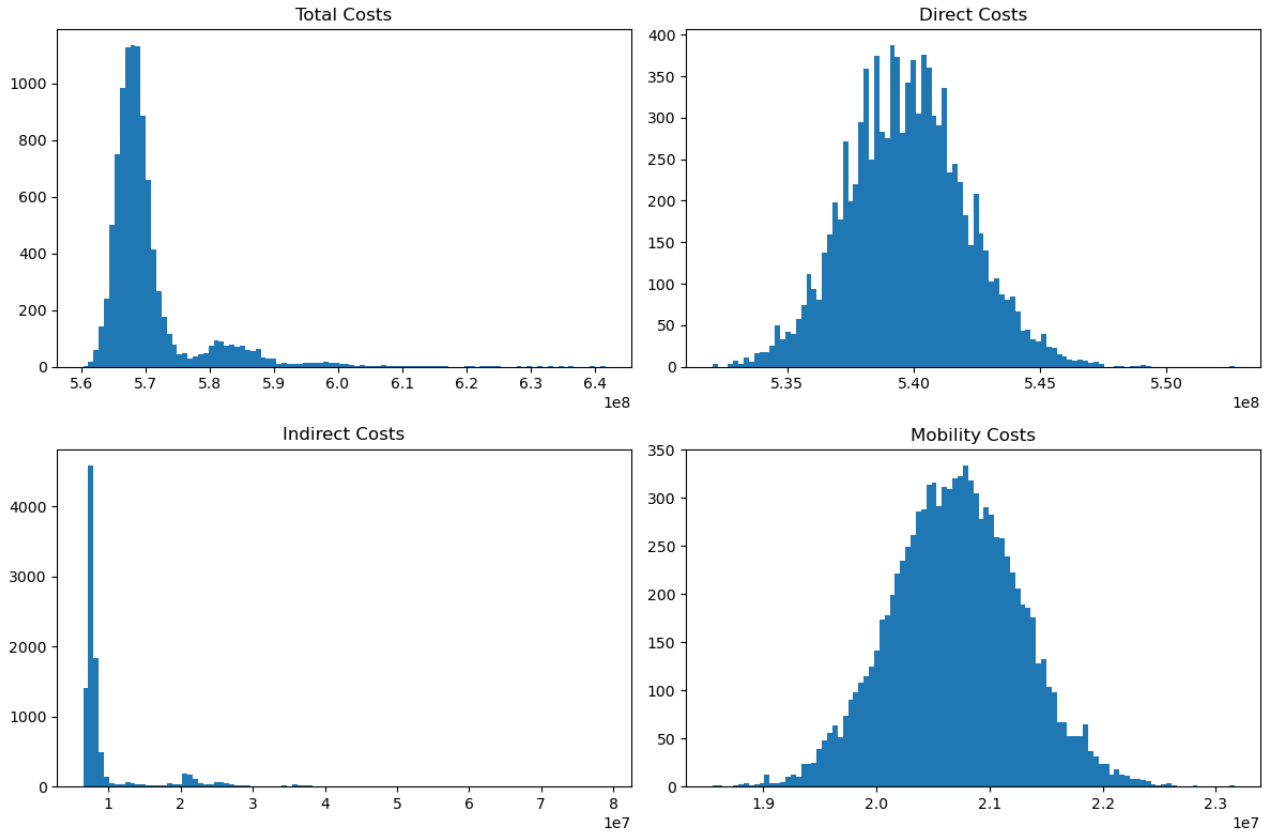


Figure 5.7: Cost histogram for scenario 2 with no constraints and only one objective to minimise i.e. the total agency cost

The figure above shows the reward/ cost distribution histogram of all the factors considered in the agency cost. The direct cost graph exhibits characteristics of a normal distribution and is symmetric. The cost is centred around 5.4 x 1e8 and shows that the policy is normally predictable. Since the mobilisation cost is highly dependent on the direct cost, a similar shape is observed. It is centred at 2.07 x 1e7. Since deterioration is a stochastic process that depends on the transition matrix, the indirect cost is skewed left. It is centred at 1 x 1e7. This effect is carried on to the total cost, with also exhibits similar traits. We extend on these observations in the next experimental run.

5.3.4. Experiment Run 2

The second experiment builds on the first run. In this run more parameters and objectives are added to see the impact on the policy and the maintenance cost. The objective added to this run is the user cost that takes into consideration the delay cost and vehicle operating cost. The objective competes with the agency cost as when the roads are in poor condition or under repair or maintenance the user cost will rise significantly. In such situations when time based maintenance policies are not adequate we need a more balanced policy that can try to minimise both the costs. The policy realised from this run can be seen in Figure 5.8.

Several observations were made regarding the policies generated by the algorithm. The agent would prefer to maintain a favourable condition of the Critical condition index and base their actions on it. In some cases, this may result in the International roughness index being in a bad state, but no action is assigned. The figure in 5.8 depicts some policies, for the entire set, refer Appendix C.2. As there are no budget constraints, the agent has successfully maintained all the components in adequate states. There are numerous minor and major repairs in the actions along with various inspections. However, in none of the components, 'reconstruction' was used, even when the component is in the last state, as seen in Component 1 IRI. This may be attributed to the high cost of the action. It was also noted that the agent would either prefer 'No inspection' or 'In-depth inspection' in most of the cases, completely avoiding 'Routine inspection'. It is probable that the cost is highly comparable, and an 'in-depth inspection' provides more precise belief updates. However, it should also be noted that in multiple components, the agent would take inspection action every year, which might not be feasible or representative of the real-world policies.

Next, based on the values retrieved in Table 5.8 we try to analyse the impact of adding user cost or two competing costs can impact the cost and other factors. We notice a significant reduction in the total cost, with almost 50% less cost given the use of the policy created. From the values presented, it is possible to infer that the agent now makes less and sparser action decisions based on the time horizon. This reduces the cost of actions and delays caused by a high number of actions. When Time-based maintenance was used, there was an inspection or maintenance action every few years, which is reflected in the user cost for the baseline. The results of this experiment suggest that resource planning is efficient, given the

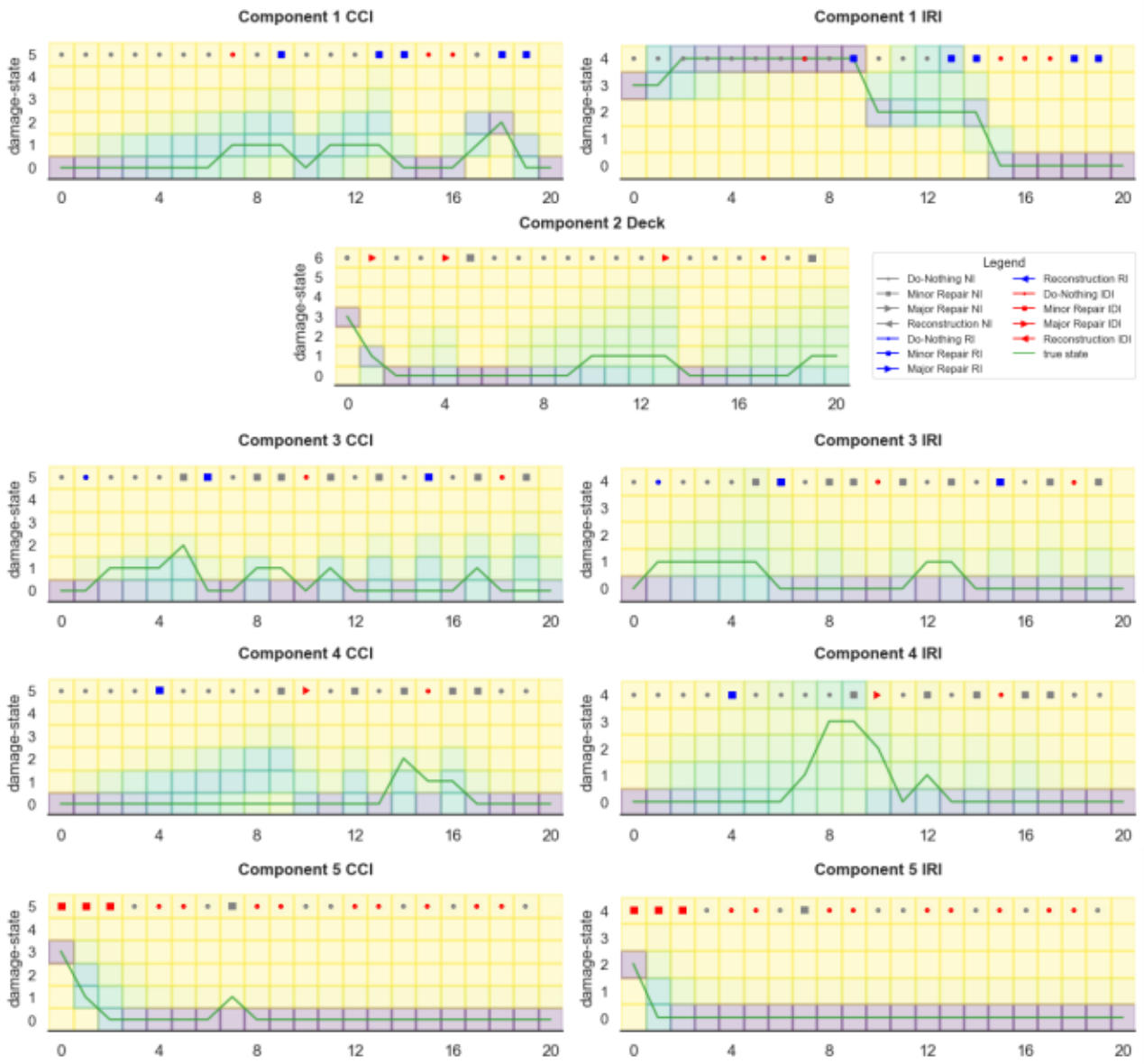


Figure 5.8: Policy realisation for scenario 2 with with two objective to minimise i.e. the total agency cost and user cost

low user cost. This is also evident in the policy graph, where we notice a lack of information, constant inspections, and the avoidance of multiple components being repaired in the same time-step.

Characteristic	TBM Baseline	Run 2 results
Total cost (mean)	2,798,868,439.3	1,353,109,339.78
Std Error	50,036,880.3	1,763,425.73
Agency cost (mean)	201,790,932.3	32,066,284.13
Direct cost (mean)	66,531,704.9	9,290,717.94
Indirect cost (mean)	37,441,211.3	14,733,114.15
User cost (mean)	2,597,077,506.9	61,921,638.50
Delay cost (mean)	1,630,438,596.6	15,451,341.86
VOP cost (mean)	966,638,910.3	46,470,296.65

Table 5.8: Comparison of various costs derived in this run with the baseline and the 1st run

5.4. Scenario 3

In the third scenario, we consider the entire network as described in the Chapter 4. This final scenario aims to demonstrate the impact of dynamic features, such as changing traffic characteristics and decision makers' preferences, on the algorithm's policy. This scenario expands on the problem presented earlier, encompassing the entire transport network, which includes 12 components within 6 links, consisting of both pavements and bridge segments. The pavement components have two features, IRI and CCI, while the bridge has one feature: deck rating.

The available actions for each segment are ['Do Nothing', 'Minor Repair', 'Major Repair', 'Replace'] x ['No Inspection', 'Routine Inspection', 'In-depth Inspection']. The agent can perform two actions per time step: one maintenance action and one inspection action. Therefore, the total possible action combinations are 12. While certain actions may be redundant and unlikely to be performed together, they are retained in the action set for this thesis. Not all components in the network start in an intact state, but the agent's belief about the state is certain for this scenario. The deterioration and inspection models applied are derived from the transition models presented in Chapter 4.

A traffic model is included in this experiment to capture the cyclical nature of traffic changes over time. The primary goal of this simulation is to minimise all costs and objectives, namely agency costs, user costs, environmental emissions, and failure levels. Since the model includes objectives with different units and ranges, the costs are processed using a multi-attribute utility model. The risk factor for this scenario is set to -1. To form the

cost function, all metrics are weighted to reflect a preference for minimising one objective over the others. The maximum value for each metric used is: $85,000,000 for agency costs, $180,000,000 for user costs, 200,000 tonnes for environment metric and 58 for safety metric. These values are arbitrary in this case.

The following subsections present one experimental runs using this scenario to evaluate the performance and decision-making of the agent under different conditions. These experiments will focus on the impact of competing objectives and changes in decision makers' preferences on the policies generated by the algorithm. Furthermore, statistical observations derived from changes in cost distribution resulting from preference changes will be analysed to provide more comprehensive insights.

5.4.1. Model Parameters

For scenario 3 Deep Centralised Multi-agent Actor Critic (DCMAC) algorithm was used for determining the policy. In this scenario the number of episodes is increased considerably to achieve convergence and optimal policy. The learning rates and discount factor is similar to the previous scenario. The memory size is also increased as the environment now is very complex and this will help create robust policies. The model parameters are as follows:

Parameter	Value
Algorithm	DCMAC
Memory capacity	30_000
Batch size	64
Discount factor	0.99
Actor Learning rate	0.0001
Critic Learning rate	0.005
Optimiser	'Adam'
Exploration strategy	Epsilon greedy
Episodes	50_000

Table 5.9: Model Parameters for experiment 3 listing the parameters setup for DCMAC

5.4.2. Experiment Run 1

The first experiment of the in the scenario is focused on evaluating the models response to the different objectives and look at the performance of multi-attribute utility model's

reward function. In this experimental run the preference or weights for all the objectives are equal i.e. 0.25. The primary aim of this test is to create a baseline which can be compared against results from future scenarios that incorporate additional parameters. This initial run gives an unrestricted evaluation of the impact of having multiple parameters in the reward function.

The main objective of the experiment run is to understand the impact of different preferences on the policy outcomes. This is done by performing a sensitivity analysis on the preferences and the policies. Sensitivity analysis is a crucial step in understanding how different factors influence outcomes in a given model. In the previous cases, the weights for all metrics were set to 0.25, indicating equal importance for each metric. This experiment aims to analyse the impact of changing weights w_{agency}, w_{user}, w_{env} and w_{safety}, to understand how prioritising one objective over others affects the costs for other metrics and the overall policy.

	Agency Weight	User Weight	Environment Weight	Safety Weight
Run 0	0.25	0.25	0.25	0.25
Run 1	0.40	0.20	0.20	0.20
Run 2	0.20	0.40	0.20	0.20
Run 3	0.20	0.20	0.40	0.20
Run 4	0.20	0.20	0.20	0.40

Table 5.10: List of scenarios with different metric weights

To achieve this, we adjust the weights so that in each scenario, it is important to note that in each scenario only one weight was changed. The resulting policies and total costs are then compared to observe the effects of these priority changes. This analysis provides insights into how increasing the importance of one metric influences the scores of the other metrics. In all scenarios listed in Table 5.10, the metric of concern has double the weight of the other metrics in individual comparisons. All other parameters remain constant. The results from the experiment can be seen in Table 5.11. Snippets from the policy are included in Appendix C.4.

Based on the values presented in the aforementioned table, the higher total values suggest a good enhancement and overall superior performance when compared to the baseline values. The agency metric shows the most increase, recommending that the cost allocated

	Total	Agency Metric	User Metric	Environment Metric	Safety Metric
TBM Baseline	9.06236	10.75857	7.87317	6.54355	11.07413
Run 0	12.01	13.96	9.78	6.11	6.11

Table 5.11: Results when compared with the first experiment and baselines

for the maintenance and overall upkeep of the transport environment has been reduced. This indicates that more efficient planning, utilisation and management of resources and actions was done. It was appropriate to take actions considering the long-term condition. The user metric also increases. This could be due decrease in the number of repair and maintenance actions reducing the delays that might be caused due to them. The expenditure on the network is enough to keep the network in satisfactory state and not increase travel burden on the users.

The decrease in the environment metric indicates that Run 0 resulted in more carbon emissions compared to the baseline but the variation is very slight. Efficient management practices and better overall conditions of the transport network likely contributed to this constant value in emissions, reflecting a commitment to sustainability and reduced environmental impact. Lastly, the lower safety metric for Run 0 shows that the network and its components were in poor condition and safety standards compared to the baseline. This suggests poor performance in keeping safety and structural integrity. This however falls in place as the expenditure on maintenance has decreased leading to lower safety levels in the network. The analysis suggests that the latest run (Run 0) has shown incremental improvements in overall performance, with noticeable gains in agency metric efficiency, and user metrics.

Additionally, the runs with varying weights were not successful due to limitations in computational power and time constraints. The difficulty of the problem and the fluctuating nature of the weights resulted in each change transforming the problem into a different POMDP. Because of this, the model could not be used in different situations, which led to poor policy performance. The fundamental issue at hand is that altering the weights alters the dynamics of the environment, rendering the previously trained policy inadequate. This is due to the fact that the policy is highly sensitive to the precise configuration of the weights,

and a modification of these parameters leads to a new environment that necessitates a complete retraining. A static policy that is based on one set of weights cannot be applied to other weight configurations, which leads to poor performance.

One way to solve this problem is to add weights to the POMDP state space. By including weights in the state representation, the agent would be able to explore how different weight configurations impact the environment and learn policies that can adapt to these variations. The agent sees weight fluctuations as a part of the environment it needs to comprehend and adapt to, rather than being fixed and known beforehand. However, introducing weights into the state space is not without its complications. Adding new factors to the state space increases its size, which can make the problem more complicated. This makes it harder to create and teach good policies because the state space becomes bigger and more complicated. The agent is required to effectively navigate this expanded state space, which necessitates meticulous consideration of how to represent and manage these supplementary factors. In future research, concepts such as hierarchical learning approaches could be explored, where the agent first learns a general policy and then refines it based on specific weight configurations. This approach can be incorporated with meta-learning to enhance adeptness.

5.4.3. Reflections

The analysis of Scenario 2 reveals significant insights into network management and maintenance policy development. Introducing nine interconnected components highlights the complexity arising from dependencies among segments, where the condition of one segment can impact others and the overall system performance as seen in Table 5.8. This underscores the necessity for a nuanced understanding of these interactions for effective maintenance planning. The agent's capacity to perform dual actions—maintenance and inspection—each time step allows for responsive, tailored strategies that balance operational efficiency and user experience. The experimental setup, considering both agency and user costs, emphasises the importance of holistic evaluation. Results indicate that failure-replacement policies lead to significantly higher costs compared to time-based maintenance, particularly in user costs, illustrating the critical role of proactive strategies.

The DCMAC algorithm effectively manages the vast state space and complex decision-

making environment, with parameters like memory capacity and learning rates proving crucial for its performance. The agent's strategic decision-making, such as avoiding high-cost 'reconstruction' actions, reflects a sophisticated balance of immediate costs and long-term benefits. The second experimental run, incorporating user costs, demonstrates the effectiveness of multi-objective optimisation, achieving a notable reduction in total costs. This balance between agency and user costs ensures sustainable network management, avoiding the overburdening of one aspect at the expense of another.

6

Discussion and Conclusion

6.1. Discussion

In this research, a transport network environment is modelled and followed by the development of a reinforcement learning model to identify an optimal inspection and maintenance policy, taking into account diverse objectives and various dynamic factors in the environment. This section focuses on the results of the project. The first step involves examining the simulated setting, followed by examining the key outcomes from various scenarios with respect to the reinforcement learning experiments. We further explore different uses and examples within the built environment where this approach can be implemented.

6.1.1. Environment Model

The method of dividing a larger problem into smaller distinct systems, such as dividing a transport network into edges and nodes, greatly enhances the understanding and management of complex systems as described in subsection 4.2.1. This approach facilitates a deeper analysis and allows for the development of targeted, effective policies. One of the most significant advantages of segmentation is the capacity to model and analyse segments based on their distinct typologies and contexts. By addressing the distinct characteristics of

each segment, strategies and policies can be tailored to their distinct requirements, thereby augmenting the overall efficiency and effectiveness. This could be observed from the policies in all the scenarios where the actions were very targeted.

Using a graph-based approach to represent intricate systems yields substantial advantages in terms of simulation and clarity. A graph of nodes and links shows the structure and interdependencies of the transport network. Although this study primarily utilised graph features such as the shortest distance, future research can leverage the full potential of graph theory to analyse connectivity, flow, and other critical properties. Additionally, this graph-based system can simplify the identification of bottlenecks and critical paths within the network, further enhancing management and optimisation efforts. The performance of this graph-based approach, however, is contingent on the careful selection of appropriate nodes and links. It is crucial to define the right boundaries and ensure that the selected nodes represent key entities or points within the system, such as intersections, cities, or data centres. Accurate node selection ensures that all critical points are included, allowing for a comprehensive analysis. These nodes can be categorised based on various criteria, including importance, distance, and type, to facilitate a more detailed and nuanced understanding of the system.

In this study, each scenario introduced new variables and experimental conditions. The initial scenario, despite its simplicity, required multiple iterations of environmental and algorithmic adjustments to achieve optimal policies. This pattern persisted across subsequent scenarios. Working with an interconnected network, the segment structure proved invaluable for maintaining clarity and precision throughout the experiments. The hierarchical organisation of segments, coupled with the classification of different segment types, supported dynamic adjustments while preserving the robustness of the overall layout and operations. The experiments yielded several valuable insights, particularly in relation to budget constraints, traffic models, and various objectives. The analysis helped understand the relationships both within and between segments. Understanding how different cost components interact within a segment can identify key cost drivers and areas for potential cost reduction. Comparing costs across segments can reveal inefficiencies, disparities, and opportunities for cross-segment optimisations, thereby guiding more informed decision-

making.

Considering the results, it is evident that traffic models play a pivotal role in shaping effective policies for each segment. By comprehending traffic patterns, policymakers can devise measures that optimise flow, mitigate congestion, and enhance safety. Traffic models also help to predict future trends, which makes it easier to make policies. By evaluating the impact of different policies on traffic within and across segments, we can ensure that decisions are data-driven and effective. The segmentation of transportation networks into graph-based models provides a powerful tool for understanding and managing complex systems. This method makes transportation management more efficient and effective by allowing for specific analysis and planning.

6.1.2. Reinforcement Learning Experiments

The scenarios generated in the previous chapter along with the experiments done in each run gave various insights regarding the environment and model. Multi-agent systems or methods like the one used here i.e., DCMAC outperformed DDQN and offered several advantages. The scenarios that use DCMAC converge faster and produce a stable policy especially in complex environments with multiple actions and components. The primary benefits of multi-agent systems include enhanced problem-solving capabilities, scalability, and robustness. By using DCMAC we were able to break down the large transport network problem in smaller and manageable sub-problems. In this sub-problem each agent focuses on one component and tries to effectively maximise or minimise the cost as required. The distributed nature of these systems also contributes to robustness, as the failure of one agent does not necessarily compromise the overall system's performance. However, an extension to this is required than can deal with environments that are constantly changing and always need to be updated with more factors. The current approach of state augmentation can quickly explode the state space causing problems in convergence and policy stability.

Extending on reinforcement learning, the connect between the environment and the model should be constructed for stable and efficient learning. In this case using a multi-attribute model helps fine-tune the environment and enhance the decision-making process. This approach allows for a more nuanced evaluation of different actions based on multiple criteria,

rather than relying on a single reward signal. By considering various attributes, such as efficiency, safety, and cost, the model can make more informed and balanced decisions. This is particularly useful in environments where trade-offs between different factors must be carefully managed. Fine-tuning preferences using a multi-attribute model helps create a more holistic and adaptable RL system, capable of handling diverse and dynamic scenarios with greater efficacy.

As we see in Scenario 2, the policy has to make a trade off between the costs for users and agency. Due to high agency cost the number of actions taken were reduced in such a way that the network is within acceptable limits. The traffic model in this case in not constant meaning the number of vehicles in the system are steadily growing. The agent therefore, has to consider not only the aspects of network condition and costs but also the inevitable increase in vehicle volume that will lead to constant rise in some costs. These costs when paired with poor network conditions and unplanned repairs can cause high economic loss for both users and agencies.

In Scenario 3, we extend this notation of multiple objectives and look at the impact of having various factors that are not comparable in their original format. A multi-attribute utility model is created to consider and compare these costs to form a reward function. Using a multi-attribute model, we aid the DCMAC to converge faster due to simplification and scaling of all the factors.

6.1.3. Extension to the Built Environment

The modelling methods described in this thesis, which involve breaking down large problems into smaller, distinct systems, can significantly enhance the understanding and management of complex architecture and urban systems. By applying similar frameworks to various areas within architecture and the built environment, we can improve both efficiency and effectiveness.

One application is in the use of modular components within a building. Different elements such as walls, floors, roofs, and services can be represented as nodes, with their dependencies and connections depicted as edges. By managing the lifecycle of each component—covering installation, maintenance, repair, and replacement—through predictive analytics, we can

determine optimal maintenance or replacement times, thereby reducing costs and extending the longevity of components.

Another interesting application is in the management and operation of building systems, such as HVAC or fire systems. System components (like air handlers, ducts, and thermostats) can be modelled as nodes with airflow paths and control signals as edges. This allows for real-time performance monitoring, enabling energy efficiency optimisation and proactive maintenance scheduling. This proactive approach helps prevent costly breakdowns and maintains comfort levels. Additionally, the concept of personalised control systems can be incorporated, where individual user preferences and behaviour patterns tailor indoor environments. This enhances occupant comfort and reduces energy consumption by optimising control strategies based on user behaviour, effectively integrating individual systems with building systems to find the most optimal policy through constant data updates and user preferences.

This framework can also be applied to architectural design and site planning, particularly in construction management across design, procurement, construction, and inspection phases. Tasks can be represented as nodes, while workflows and dependencies are edges. Using a deep reinforcement framework alongside BIM and other data-driven tools can optimise project timelines, allocate resources efficiently, and manage costs by identifying critical paths and interdependencies within project phases. This approach isn't limited to planning—it can also be applied on-site through construction robots performing tasks like bricklaying, welding, and inspection. By optimising the allocation of robots and human workers to tasks, we can improve construction efficiency and enhance safety by understanding interactions between robots and human workers.

Extending this outlook to the urban scale, we can apply similar modelling methods to city infrastructure management. By modelling different infrastructure systems together—such as transport networks, electric grids, and underground pipelines—we can understand their interdependencies. This approach optimises resource allocation for infrastructure maintenance and service delivery by identifying critical components and usage patterns. It also enhances resilience planning by understanding interdependencies and potential failure points, enabling proactive risk management.

The modelling methods discussed in this thesis offer various tools for improving the management and efficiency of complex systems in architecture, construction, and urban planning.

6.2. Limitations

This study has several limitations that must be acknowledged. To begin with, a significant portion of the data and readings used are based on data from similar regions or empirical data rather than direct on-site measurements, potentially leading to inaccuracies. This lack of data and characteristics required for the project resulted in multiple simplifications. Furthermore, the analysis assumes that road segments deteriorate independently, overlooking common factors that may affect multiple segments, as well as shared conditions impacting repair works. This line of thought can also be extended to the inspection actions and their findings. The focus on the International Roughness Index (IRI) and the Crack Condition Index (CCI) as performance indicators limits the comprehensive understanding of road conditions; a combination of multiple features could offer a more complete assessment.

Secondly, the traffic model spans one year, assuming constant traffic volume, which may not reflect real-world fluctuations. Additionally, due to a lack of literature and socio-economic models, maximum values for the multi-attribute utility model were assumed from trial and error, influencing the model's reliability. The model also excludes important objectives such as social metrics for availability and connectivity, which could provide a more holistic evaluation. Furthermore, the transportation system was regarded as an autonomous entity; however, there exist numerous factors, systems, and models within the built environment that are interconnected and impacted by this system. Therefore, the results and policies are contained and experimental.

Finally, high computational time constraints led to excluding many experiments, limiting the scope and robustness of the findings. Addressing these limitations in future research could improve the accuracy and comprehensiveness of the study's outcomes.

6.3. Future Work

The project offers various directions for future research, which can be broadly categorised into the modelling aspect and the computational aspect.

On the modelling side, the work can be enhanced by incorporating social objectives such as equity and equality into the availability and maintenance of the infrastructure system. Expanding the reward model to include the economic costs of repair and maintenance, detailing actions taken, and varying costs based on the state of the components would provide a more nuanced approach. Moreover, the development of an efficient modelling framework that yields outcomes at diverse micro, meso-, and macroscopic scales, recognising that these scales may vary across the network, will facilitate collaborative learning among adjacent segments and facilitate dynamic updates of values. Establishing spatial and structural correlations between components can improve the model's accuracy, emphasising the need for precise inspections of all segments at all times. The expansion of the scope beyond operations and maintenance to include decommissioning and material recycling would further enhance the circularity of the system.

From a computational perspective, using deep reinforcement learning in fast-changing environments could benefit from the incorporation of prioritised sweeping updates to keep the environment and computations current. Exploring the integration of Graph Neural Networks (GNNs) in the agent, given the adoption of graph typologies, presents another promising direction.

Additionally, a comprehensive approach to network management, beyond just maintenance, could be developed. This would involve suggesting areas prone to overload, identifying critical areas, determining the need for new junctions and paths, and potentially integrating the transportation network with underground utilities for cross-network management. These enhancements would significantly advance the capabilities and applications of the project, contributing to more efficient and effective infrastructure management.

Bibliography

Abdallah, A. M., Atadero, R. A., and Ozbek, M. E. (2022). A state-of-the-art review of bridge inspection planning: Current situation and future needs. *J. Bridge Eng.*, 27(2). https://doi.org/10.1061/(asce)be.1943-5592.0001812.

Andriotis, C. P. and Papakonstantinou, K. G. (2019). Managing engineering systems with large state and action spaces through deep reinforcement learning. *Reliab. Eng. Syst. Saf.*, 191.

Ang, A.-S. and Tang, W. (1984). *Probability Concepts in Engineering Planning and Design: Decision, Risk, and Reliability*, volume II. Wiley.

Australian Transport Assessment and Planning (2021). M1 technical report. Technical report.

Barth, M. and Boriboonsomsin, K. (2008). Real-world carbon dioxide impacts of traffic congestion. 2058(1):163–171.

Bektas, B. and Albughdadi, A. (2021). Bridge decommissioning and its impact on bridge management. In *Bridge Maintenance, Safety, Management, Life-Cycle Sustainability and Innovations*. CRC Press, 1 edition.

Bhowmick, D. and Mitra, S. (2017). Status of signalized intersection safety-a case study of kolkata.

Bhustali, P. (2023). Imprl: Implementations of various rl algorithms. https://github.com/omniscientoctopus/imprl.

Boyan, J. A. and Littman, M. L. (2000). Exact solutions to time-dependent mdps. In *Neural Information Processing Systems*.

Bryce, J. M., Flintsch, G. W., Diefenderfer, B. K., and Katicha, S. W. (2012). A pavement structural capacity index for use in network-level evaluation of asphalt pavements. *Virginia Tech Works*. Available online at Virginia Tech's institutional repository.

Bureau of Public Roads, U.S. Department of Commerce (1964). *Traffic Assignment Manual: For Application with a Large, High Speed Computer*.

California County Association of Governments (2005). Appendix b: Traffic level of service calculation methods.

Calvert, G., Neves, L., Andrews, J., and Hamer, M. (2020). Multi-defect modelling of bridge deterioration using truncated inspection records. *Reliability Engineering System Safety*, 200:106962.

Cheng, M., Yang, D. Y., and Frangopol, D. M. (2020). Investigation of effects of time preference and risk perception on life-cycle management of civil infrastructure. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 6(1).

Chowdhury, T. (2016). Supporting document for the development and enhancement of the pavement maintenance decision matrices used in the needs-based analysis. Technical report, Virginia Transportation Research Council, Maintenance Division, Richmond, VA.

Chu, M., Liu, A., Lau, V. K. N., Jiang, C., and Yang, T. (2022). Deep reinforcement learning based end-to-end multiuser channel prediction and beamforming. *IEEE Transactions on Wireless Communications*, 21(12):10271–10285.

Demir, and Dicleli, M. (2023). Live load effects in hammer-head piers of continuous highway bridges and design equations based on numerical simulations verified by field tests. *Engineering Structures*, 279.

Dong, Y., Frangopol, D. M., and Sabatino, S. (2015). Optimizing bridge network retrofit planning based on cost-benefit evaluation and multi-attribute utility associated with sustainability. *Earthquake Spectra*.

Faddoul, R., Raphael, W., Soubra, A.-H., and Chateauneuf, A. (2013). Incorporating bayesian networks in markov decision processes. *Journal of Infrastructure Systems*, 19(4):415–424.

Federal Highway Administration (2002). System conditions. https://www.fhwa.dot.gov/policy/2002cpr/ch3b.cfm.

Federal Highway Administration (2018). Simplified highway capacity calculation method for the highway performance monitoring system. Online. https://www.fhwa.dot.gov/policyinformation/pubs/pl18003/hpms_cap.pdf.

Federal Highway Administration (2019). *Bridge Replacement Unit Costs*. Federal Highway Administration, Washington D.C.

Federal Highway Administration (2021). FHWA resources for the asset management practitioner. Accessed November 5, 2021.

Federal Highway Administration (2023a). 2023 fhwa forecasts of vehicle miles traveled (vmt). Accessed: 2024-06-25.

Federal Highway Administration (2023b). Estimating benefits for bridge protection improvements.

Federal Highway Administration (2023c). Table vm-4 - highway statistics 2021: Distribution of annual vehicle distance traveled - 2021. Accessed: 2024-06-25.

Federal Highway Administration (2023d). Work zone road user costs - concepts and applications: Chapter 2. work zone road user costs.

Federal Highway Administration (2024). Traffic volume trends - vmt forecast summary. Accessed: 2024-06-09.

Firdous, N., Mohi Ud Din, N., and Assad, A. (2023). An imbalanced classification approach for establishment of cause-effect relationship between heart-failure and pulmonary embolism using deep reinforcement learning. *Engineering Applications of Artificial Intelligence*.

Gardiner, G. (2022). Refurbishing bridges at half the time, cost versus replacement. https://www.compositesworld.com/articles/refurbishing-bridges-at-half-the-time-cost-versus-replacement.

Gillespie, T. D., Paterson, W. D., and Sayers, M. W. (1986). Guidelines for conducting and calibrating road roughness measurements. Technical report, The World Bank. World Bank Technical Paper Number 46.

Gudivada, V. N., Rao, D., and Raghavan, V. V. (2015). Big Data Driven Natural Language Processing Research and Applications. In Govindaraju, V., Raghavan, V. V., and Rao, C., editors, *Handbook of Statistics*, volume 33 of *Handbook of Statistics*, page 203–238. Elsevier.

Haydari, A. and Yılmaz, Y. (2022). Deep reinforcement learning for intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):11–32.

Joe, W. and Lau, H. C. (2020). Deep reinforcement learning approach to solve dynamic vehicle routing problem with stochastic customers. *Proceedings of the International Conference on Automated Planning and Scheduling*, 30(1):394–402.

Kerkhof, R. M. v. d., Lamper, L., and Fang, F. (2018). *De waarde van Smart Maintenance voor de Nederlandse Infrastructuur*. World Class Maintenance.

Krachtopoulos, K. (2023). Multi-objective deep reinforcement learning for predictive maintenance of road networks. Master's thesis, Technical University of Delft. `http://resolver.tudelft.nl/uuid:90714f43-5f34-4dce-b1e3-3623cdc8fde1`.

Lei, X., Dong, Y., and Frangopol, D. M. (2023). Sustainable life-cycle maintenance policymaking for network-level deteriorating bridges with a convolutional autoencoder–structured reinforcement learning agent. *Journal of Bridge Engineering*, 28(9):04023063. `https://doi.org/10.1061/JBENF2.BEENG-6159`.

Lou, P., Nassif, H., Su, D., and Truban, P. (2016). Effect of overweight trucks on bridge deck deterioration based on weigh-in-motion data. *Transportation Research Record*, 2592(1):86–97.

Lu, J. J., Huang, W., and Mierzejewski, E. A. (1997). Driver population factors in freeway capacity. Final Report WPI 0510759, University of South Florida, Center for Urban Transportation Research. Accession Number: 00737930.

Maljaars, J. (2020). Evaluation of traffic load models for fatigue verification of european road bridges. *Engineering Structures*, 225.

Manafpour, A., Guler, I., Radlińska, A., Rajabipour, F., and Warn, G. (2018). Stochastic analysis and time-based modeling of concrete bridge deck deterioration. *Journal of Bridge Engineering*, 23(9).

Margiotta, R. A. and Washburn, S. S. (2017). Simplified highway capacity calculation method for the highway performance monitoring system. Technical Report PL-18-003, Federal Highway Administration, Office of Policy and Governmental Affairs.

Medury, A. and Madanat, S. (2013). Incorporating network considerations into pavement management systems: A case for approximate dynamic programming. *Transportation Research Part C: Emerging Technologies*, 33:134–150. https://doi.org/10.1016/j.trc.2013.03.003.

Mendoza Lugo, M. A., Nogal, M., and Morales-Nápoles, O. (2024). Estimating bridge criticality due to extreme traffic loads in highway networks. *Engineering Structures*, 300:117172.

Missouri Department of Transportation (2019). Category:618 mobilization. Accessed on 2024-06-06.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

New Jersey Department of Transportation and Federal Highway Administration (2015). Impact of freight on highway infrastructure in new jersey. Technical Report FHWA-NJ-2016-004, New Jersey Department of Transportation.

Orcesi, A. D. and Frangopol, D. M. (2011). Probability-based multiple-criteria optimization of bridge maintenance using monitoring and expected error in the decision process. *Structural and Multidisciplinary Optimization*, 44(2):137–148.

Pennsylvania Department of Transportation and Bureau of Design (2022). *Bridge Management System 2 (BMS2) Coding Manual: Publication 100A*, 2022 edition.

Rossow, M. P. (2003). *FHWA Bridge Maintenance Manual—Decks*. Federal Highway Administration.

Saifullah, M., Andriotis, C., Papakonstantinou, K., and Stoffels, S. (2022). Deep reinforcement learning-based life-cycle management of deteriorating transportation systems. In *11th International Conference on Bridge Maintenance, Safety and Management (IABMAS)*. https://par.nsf.gov/biblio/10350545.

Saifullah, M., Papakonstantinou, K. G., Andriotis, C. P., and Stoffels, S. M. (2024). Multi-agent deep reinforcement learning with centralized training and decentralized execution for transportation infrastructure management. *arXiv preprint arXiv:2401.12455*. http://arxiv.org/abs/2401.12455.

Sjaarda, M., Meystre, T., Nussbaumer, A., and Hirt, M. A. (2020). A systematic approach to estimating traffic load effects on bridges using weigh-in-motion data. *Stahlbau*, 89(7):585–598.

Sub-Saharan Africa Transport Policy Program (2006). Red model version 3.2: Hdm-4 voc (version 3.2). https://www.ssatp.org/sites/ssatp/files/publications/HTML/Models/RED_3.2/RED%20-%20RED%20Model%20Version%203.2/RED%20-%20HDM-4%20VOC%20(version%203.2).xls.

Suo, B., Cheng, Y., Zeng, C., and Li, J. (2012). Calculation of failure probability of series and parallel systems for imprecise probability. *International Journal of Engineering and Manufacturing (IJEM)*, 2(2):91–98. https://www.mecs-press.org/ijem/ijem-v2-n2/IJEM-V2-N2-12.pdf.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. The MIT Press, 2nd edition.

United States Department of Transportation (2000). 1999 status of the nation's highways, bridges and transit: Conditions & performance.

van Hasselt, H., Guez, A., and Silver, D. (2015). Deep reinforcement learning with double q-learning. *arXiv preprint arXiv:1509.06461*. Presented at AAAI 2016.

Virginia Department of Transportation (2018). State of the pavement 2018. Online. https://www.vdot.virginia.gov/media/vdotvirginiagov/about/vdots-transportation-system/highways/state-of-the-pavement/State_of_the_Pavement_2018.pdf.

Wardrop, J. G. and Whitehead, J. I. (1952). Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers*, 1(5):767–768.

Wells, D. T. (1995). Technical assistance report: Maintenance, repair, and rehabilitation unit costs for pontis. Technical report, Virginia Transportation Research Council.

Wiering, M. (2001). Reinforcement learning in dynamic environments using instantiated information.

Xu, G. and Guo, F. (2022). Sustainability-oriented maintenance management of highway bridge networks based on q-learning. *Sustainable Cities and Society*, 89.

Yan, T., Marasteanu, M., Turos, M., Barman, M., Manikavasagan, V., and Chakraborty, M. (2023). Cost estimate of b vs. c grade asphalt binders. Technical Report MN 2023-19, Minnesota Department of Transportation.

Yang, D. Y. (2022). Deep Reinforcement Learning–Enabled Bridge Management Considering Asset and Network Risks. *Journal of Infrastructure Systems*, 28(3):04022023.

Zhang, X., Shi, X., Zhang, Z., Wang, Z., and Zhang, L. (2022). A ddqn path planning algorithm based on experience classification and multi steps for mobile robots. *Electronics*, 11(14).

Zhou, W., Miller-Hooks, E., Papakonstantinou, K. G., Stoffels, S., and McNeil, S. (2022). A reinforcement learning method for multiasset roadway improvement scheduling considering traffic impacts. *Journal of Infrastructure Systems*, 28(4). https://doi.org/10.1061/(ASCE)IS.1943-555X.0000702.

Żochowska, R. and Soczówka, P. (2018). Analysis of selected transportation network structures based on graph measures. *Scientific Journal of Silesian University of Technology. Series Transport*, 98:223–233. https://doi.org/10.20858/sjsutst.2018.98.21.

A

Environment Dynamics

A.1. Pavement dynamics

A.1.1. State-Action Transition Probability for Pavement features

Transition Probability for International Roughness Index (IRI)

Below, the transition probabilities for the actions retrieved by Saifullah et al. (2022) are presented:

$$P(s_t|s_{t+1}, 'Do\ nothing') = \begin{bmatrix} 0.840 & 0.121 & 0.039 & 0.0 & 0.0 \\ 0.0 & 0.788 & 0.142 & 0.070 & 0.0 \\ 0.0 & 0.0 & 0.708 & 0.192 & 0.100 \\ 0.0 & 0.0 & 0.0 & 0.578 & 0.422 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

$$P(s_t|s_{t+1}, 'Minor\ repair') = \begin{bmatrix} 0.970 & 0.030 & 0.0 & 0.0 & 0.0 \\ 0.850 & 0.120 & 0.030 & 0.0 & 0.0 \\ 0.450 & 0.400 & 0.120 & 0.030 & 0.0 \\ 0.0 & 0.450 & 0.400 & 0.120 & 0.030 \\ 0.0 & 0.0 & 0.450 & 0.400 & 0.150 \end{bmatrix}$$

$$P(s_t|s_{t+1}, 'Major\ repair') = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.950 & 0.050 & 0.0 & 0.0 & 0.0 \\ 0.800 & 0.200 & 0.0 & 0.0 & 0.0 \\ 0.700 & 0.250 & 0.050 & 0.0 & 0.0 \\ 0.450 & 0.350 & 0.200 & 0.0 & 0.0 \end{bmatrix}$$

$$P(s_t|s_{t+1}, 'Reconstruction') = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

Transition Probability for International Roughness Index (IRI)

A.1.2. Observation Probability for Pavement features

Observation Probabilities for International Roughness Index (IRI)

Inspection type	$p(o_t = j - 2 \mid s_t = j)$	$p(o_t = j - 1 \mid s_t = j)$	$p(o_t = j \mid s_t = j)$	$p(o_t = j + 1 \mid s_t = j)$	$p(o_t = j + 2 \mid s_t = j)$
No inspection	0.20	0.20	0.20	0.20	0.20
Routine inspection	0.0	0.20	0.60	0.20	0.0
In-depth inspection	0.0	0.05	0.90	0.05	0.0

Table A.1: Observation Probability $p(o_t|s_t)$ for different inspection actions if true IRI state is s_t

Observation Probabilities for Critical Condition Index (CCI)

Observation Probability $p(o_t|s_{t+1})$ if pavement inspection action is 'routine inspection' for true CCI state s_t:

$$P(o_t|s_{t+1}, 'Routine\_inspection') = \begin{bmatrix} 0.687 & 0.259 & 0.054 & 0.0 & 0.0 & 0.0 \\ 0.276 & 0.422 & 0.297 & 0.005 & 0.0 & 0.0 \\ 0.023 & 0.139 & 0.648 & 0.167 & 0.022 & 0.001 \\ 0.0 & 0.003 & 0.266 & 0.455 & 0.248 & 0.028 \\ 0.0 & 0.0 & 0.031 & 0.224 & 0.486 & 0.259 \\ 0.0 & 0.0 & 0.0 & 0.005 & 0.059 & 0.936 \end{bmatrix}$$

Observation Probability $p(o_t|s_{t+1})$ if pavement inspection action is 'in-depth inspection' for true CCI state s_t:

$$P(o_t|s_{t+1}, 'Indepth\_inspection') = \begin{bmatrix} 0.801 & 0.197 & 0.002 & 0.0 & 0.0 & 0.0 \\ 0.153 & 0.664 & 0.183 & 0.0 & 0.0 & 0.0 \\ 0.001 & 0.078 & 0.822 & 0.099 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.149 & 0.693 & 0.158 & 0.0 \\ 0.0 & 0.0 & 0.001 & 0.137 & 0.718 & 0.144 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.042 & 0.958 \end{bmatrix}$$

A.2. Bridge dynamics

A.2.1. State-Action Transition Probability for Deck features

$$P(s_t|s_{t+1}, 'Minor\ repair') = \begin{bmatrix} 0.970 & 0.030 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.850 & 0.120 & 0.030 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.400 & 0.450 & 0.120 & 0.030 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.400 & 0.450 & 0.120 & 0.030 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.400 & 0.450 & 0.120 & 0.030 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.400 & 0.450 & 0.150 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

$$P(s_t|s_{t+1},'Major\ repair') = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.950 & 0.050 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.800 & 0.200 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.600 & 0.300 & 0.100 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.400 & 0.400 & 0.200 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.300 & 0.400 & 0.300 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

$$P(s_t|s_{t+1},'Reconstruction') = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

A.2.2. Observation Probability for Deck features

Observation Probabilities for Deck states based on NBI condition ratings

Observation Probability $p(o_t|s_{t+1})$ if deck inspection action is 'routine inspection' for true Deck state s_t:

$$P(o_t|s_{t+1},'Routine\_inspection') = \begin{bmatrix} 0.800 & 0.150 & 0.050 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.150 & 0.650 & 0.150 & 0.050 & 0.0 & 0.0 & 0.0 \\ 0.050 & 0.150 & 0.600 & 0.150 & 0.050 & 0.0 & 0.0 \\ 0.0 & 0.050 & 0.150 & 0.600 & 0.150 & 0.050 & 0.0 \\ 0.0 & 0.0 & 0.050 & 0.150 & 0.650 & 0.150 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.050 & 0.150 & 0.800 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

Observation Probability $p(o_t|s_{t+1})$ if deck inspection action is 'in-depth inspection' for true

Deck state s_t:

$$P(o_t|s_{t+1}, 'Indepth\_inspection') = \begin{bmatrix} 0.900 & 0.100 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.100 & 0.800 & 0.100 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.100 & 0.800 & 0.100 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.100 & 0.800 & 0.100 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.100 & 0.800 & 0.100 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.100 & 0.900 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

B

Algorithms used in the experiments

B.1. Pseudo-code for algorithms

B.1.1. Double Q-Learning (DDQN)

Algorithm 1 Double Q-Learning (DDQN)

Initialise primary network Q_θ, target network $Q_{\theta'}$, replay buffer D, $\tau << 1$
for each iteration **do**
  for each environment step **do**
    Observe state s_t and select $a_t \sim \pi(a_t, s_t)$
    Execute a_t and observe next state s_{t+1} and reward $r_t = R(s_t, a_t)$
    Store (s_t, a_t, r_t, s_{t+1}) in replay buffer D
  end for
  for each update step **do**
    sample $e_t = (s_t, a_t, r_t, s_{t+1}) \sim D$
    Compute target Q value:
      $Q^*(s_t, a_t) \approx r_t + \gamma\ Q_\theta(s_{t+1}, argmax_{a'} Q_{\theta'}(s_{t+1}, a'))$
    Perform gradient descent step on $(Q^*(s_t, a_t) - Q_\theta(s_t, a_t))^2$
    Update target network parameters:
      $\theta' \leftarrow \tau * \theta + (1 - \tau) * \theta'$
  end for
end for

B.1.2. Deep Centralized Multi-agent Actor Critic (DCMAC)

Algorithm 2 Deep Centralized Multi-agent Actor Critic (DCMAC)

Initialise replay buffer
Initialise actor and critic network weights θ^{π}, θ^{V}
for episode = 1, M **do**
  for t = 1, T **do**
    Select action a_t at random according to exploration noise
    Otherwise select $a_t \sim \mu_t = \pi\left(\cdot \mid \hat{b}_t, \theta^{\pi}\right)$
    Collect reward $r\left(\hat{b}_t, a_t\right)$ sampling $\hat{b}_t$
    Observe $o_{t+1}^{(l)} \sim p\left(o_{t+1}^{(l)} \mid b_t^{(l)}, a_t\right)$ for $l = 1, 2, ..., m$
    Compute beliefs $b_{t+1}^{(l)}$ for $l = 1, 2, ..., m$

$$b^{(l)}\left(s_{t+1}^{(l)}\right) = \frac{p\left(o_{t+1}^{(l)} | s_{t+1}^{(l)}, a_t\right)}{p\left(o_{t+1}^{(l)} | b_t^{(l)}, a_t\right)} \sum_{s_t^{(l)} \in S^{(l)}} p\left(s_{t+1}^{(l)} \mid s_t^{(l)}, a_t\right) b^{(l)}\left(s_t^{(l)}\right)$$

    Store experience $\left(\hat{b}_t, a_t, \mu_t, r\left(\hat{b}_t, a_t\right), \hat{b}_{t+1}\right)$ to replay buffer
    Sample batch of $\left(\hat{b}_i, a_i, \mu_i, r\left(\hat{b}_i, a_i\right), \hat{b}_{i+1}\right)$ from replay buffer
    If $\hat{b}_{i+1}$ is terminal state $A_i = r\left(\hat{b}_i, a_i\right) - V\left(\hat{b}_i \mid \theta^{V}\right)$
    Otherwise $A_i = r\left(\hat{b}_i, a_i\right) + \gamma V\left(\hat{b}_{i+1} \mid \theta^{V}\right) - V\left(\hat{b}_i \mid \theta^{V}\right)$
    Update actor parameters θ^{π} according to gradient:

$$g_{\theta^{\pi}} \simeq \sum_i w_i \left(\sum_{j=1}^{n} \nabla_{\theta^{\pi}} log \pi_j(a_i^{(j)} \mid \hat{b}_i, \theta^{\pi})\right) A_i$$

    Update actor parameters θ^{V} according to gradient:

$$g_{\theta^{V}} \simeq \sum_i w_i \nabla_{\theta^{V}} V^{\pi}\left(\hat{b}_t \mid \theta^{V}\right) A_i$$

  end for
end for

C

Additional Experiment Runs

C.1. Scenario 1 Runs

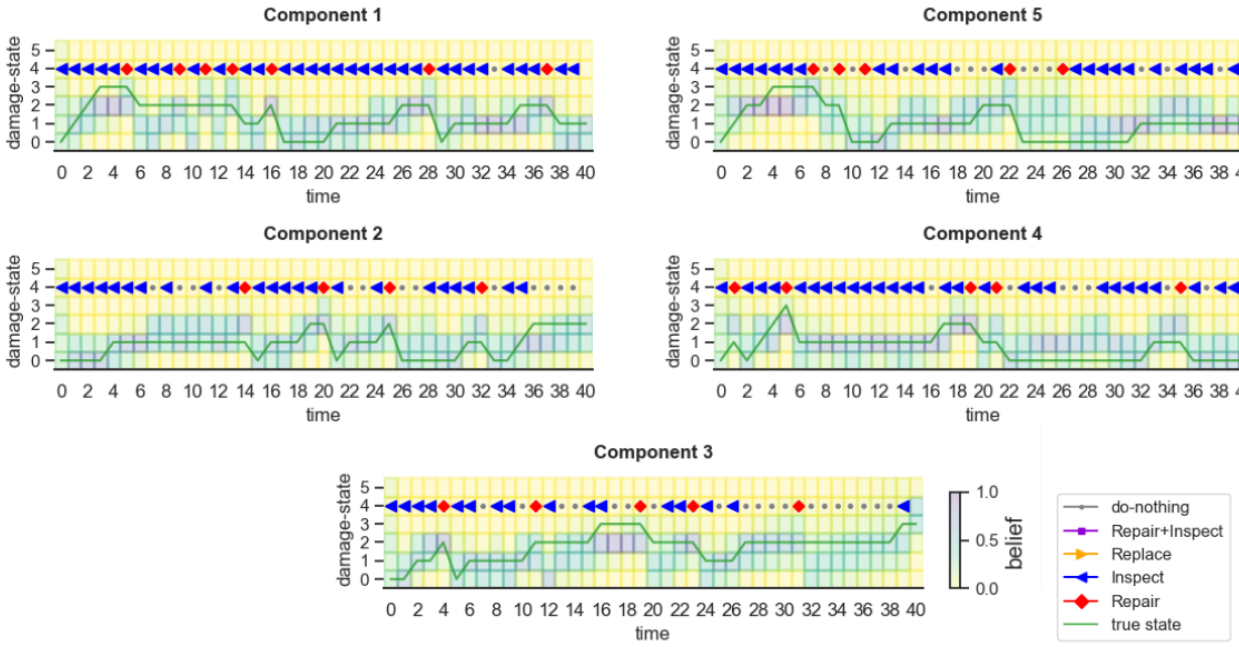


Figure C.1: Run in scenario 1 when inspection action was suggested at every time-step when repair or maintenance.

C.2. Scenario 2 Runs

C.2.1. Entire Policy for Run 2 in Scenario 2

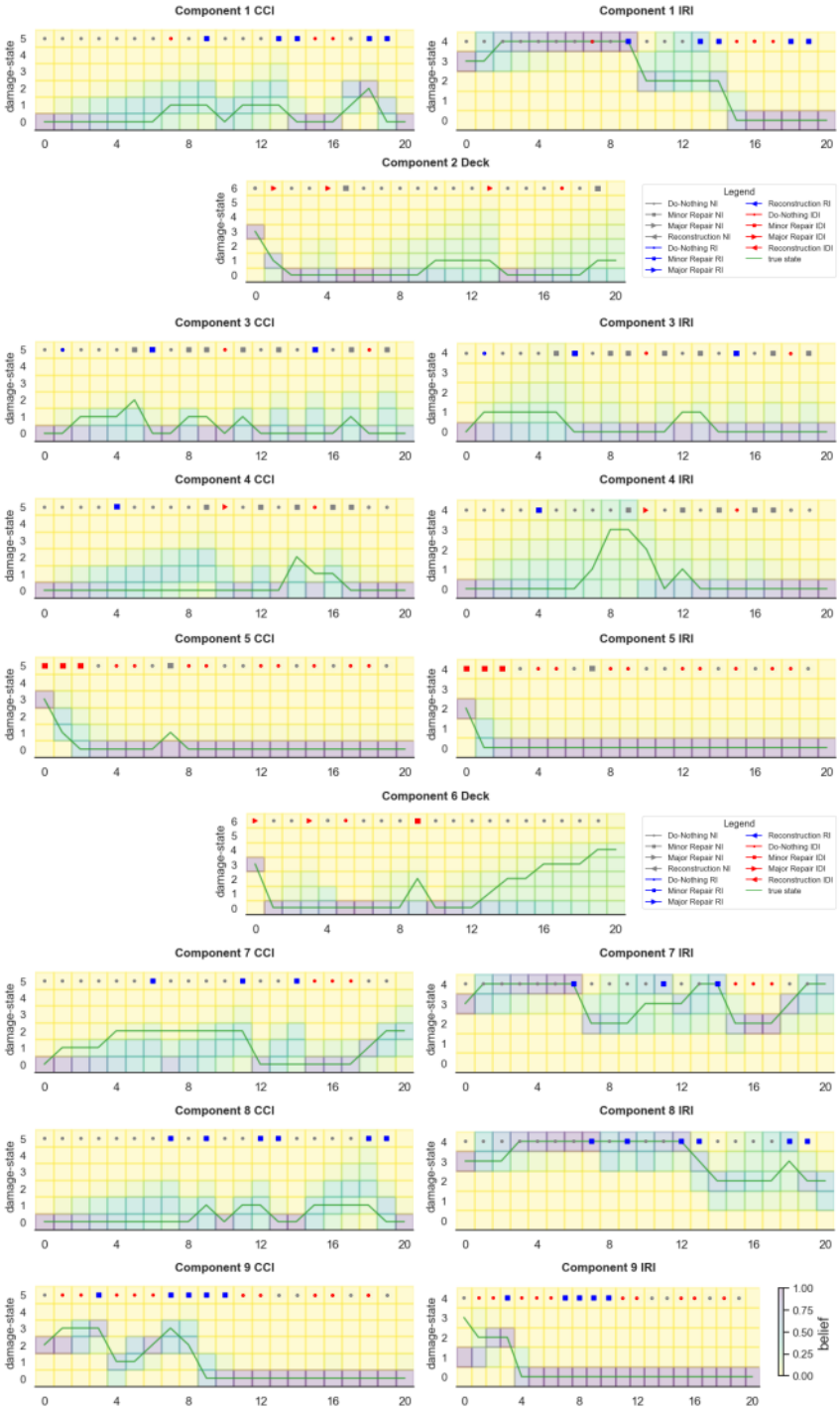


Figure C.2: Run in scenario 2 for all the components when minimising 2 objectives agency and user.

C.3. Scenario 3 Runs

C.3.1. Policy for Run 0 in Scenario 3

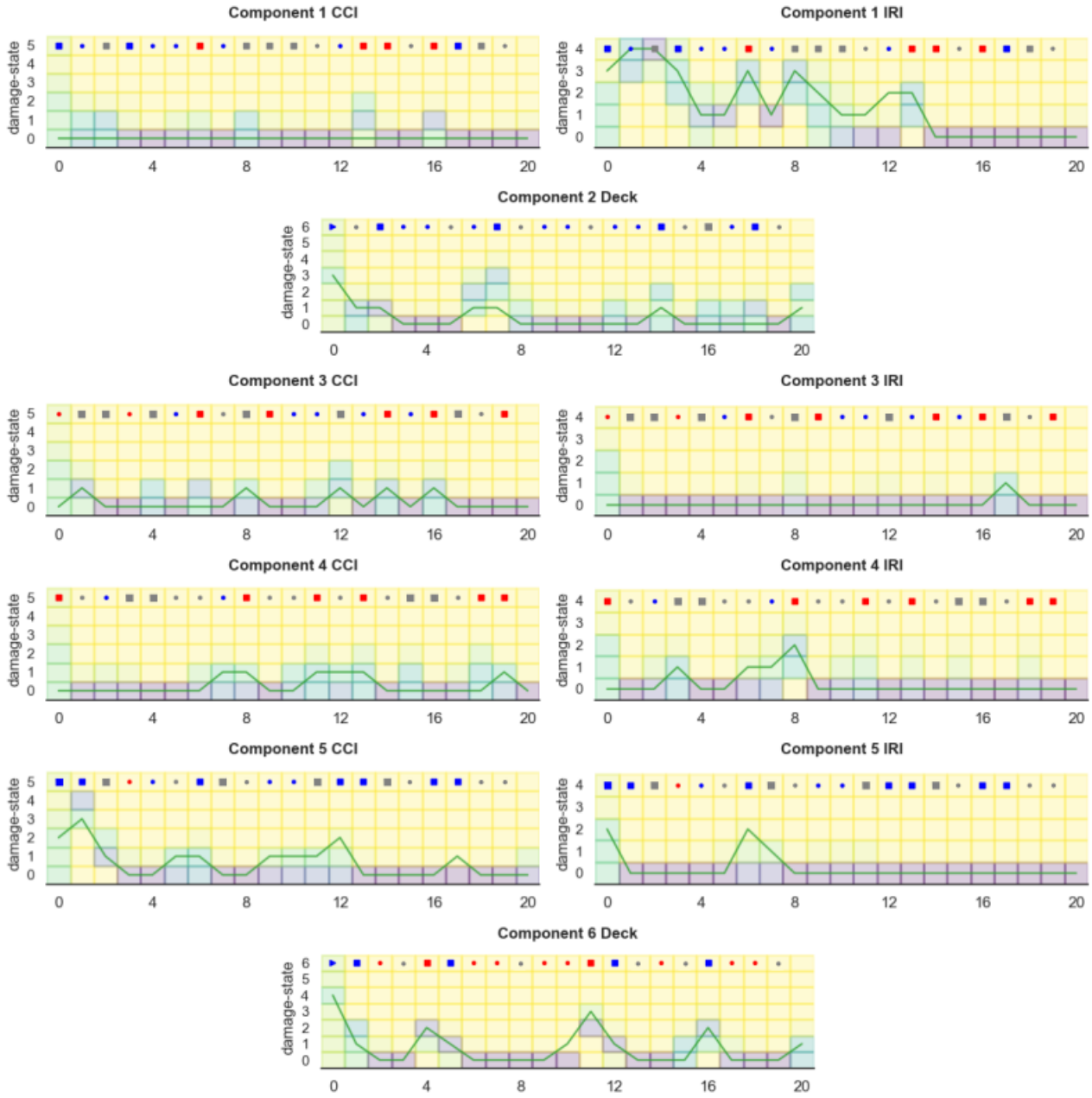


Figure C.3: Run in scenario 3 for first 6 components with all the objectives

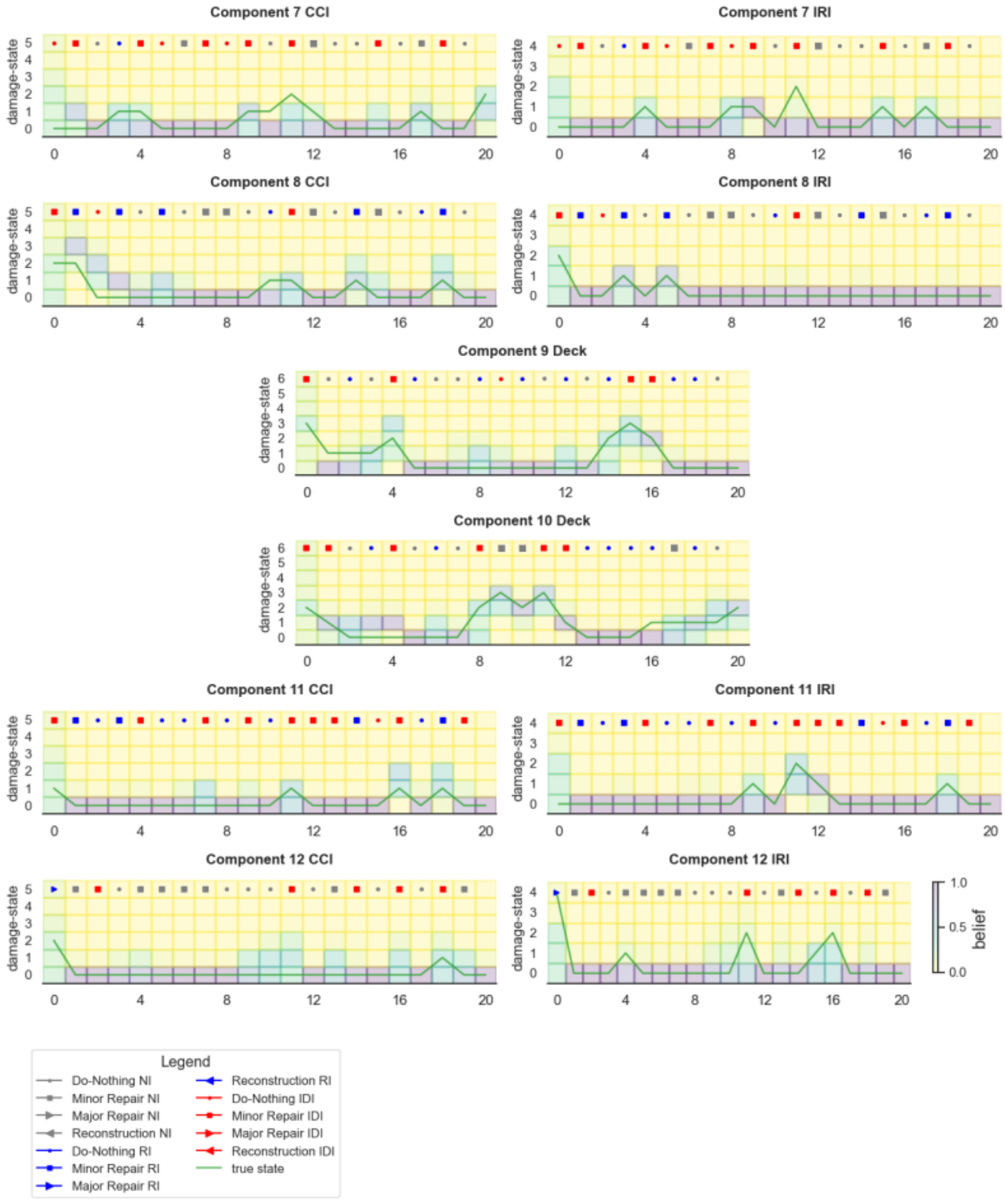


Figure C.4: Run in scenario 3 for next 6 components with all the objectives